REPORT NO. UMTA-MA-06-0025-80-4

INCREASED RAIL TRANSIT VEHICLE CRASHWORTHINESS IN HEAD-ON COLLISIONS

Volume IV - IITRAIN Users' Manual

Edward E. Hahn

IIT RESEARCH INSTITUTE
10 West 35th Street
Chicago IL 60616

JUNE 1980
FINAL REPORT

Prepared by

U.S. DEPARTMENT OF TRANSPORTATION
URBAN MASS TRANSPORTATION ADMINISTRATION
Office of Technology Development and Deployment
Washington DC 20590

NOTICE

This document is disseminated under the sponsorship of the Department of Transportation in the interest of information exchange. The United States Government assumes no liability for its contents or use thereof.

NOTICE

The United States Government does not endorse products or manufacturers. Trade or manufacturers' names appear herein solely because they are considered essential to the object of this report.

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
UMTA-MA-06-0025-80-4		

4. Title and Subtitle	5. Report Date
INCREASED RAIL TRANSIT VEHICLE CRASHWORTHINESS IN HEAD-ON COLLISIONS - Volume IV-IITRAIN Users' Manual	June 1980
	6. Performing Organization Code
7. Author(s) Edward E. Hahn	8. Performing Organization Report No. DOT-TSC-UMTA-80-17, IV
9. Performing Organization Name and Address IIT Research Institute * 10 West 35th Street Chicago IL 60616	10. Work Unit No. (TRAIS) UM904/R0734 11. Contract or Grant No. DOT-TSC-1052-4
12. Sponsoring Agency Name and Address U.S. Department of Transportation Urban Mass Transportation Administration Office of Technology Development and Deployment Washington DC 20590	13. Type of Report and Period Covered Final Report June 1975 to June 1978 14. Sponsoring Agency Code

15. Supplementary Notes

*Under Contract to: U.S. Department of Transportation
Research and Special Programs Administration
Transportation Systems Center
Cambridge MA 02142

16. Abstract

A computer code for the simulated crash of two railcar consists is described. The code is capable of simulating the mechanics of head-on impact of two consists on straight level track. The simulation is limited to two dimensions, namely a vertical plane containing the length of the track. The user can model the individual car components or cars in a complex or as simple a manner as is warranted by the simulation results desired. Although specifically developed for transit cars, the code can also be used to simulate freight trains or intercity passenger trains.

17. Key Words	18. Distribution Statement
computer crash simulation, head-on collisions, railway car crashworthiness, train crashes, transportation safety, mass transit, urban rail car, rapid transit car, crashworthiness, commuter rail car.	DOCUMENT IS AVAILABLE TO THE PUBLIC THROUGH THE NATIONAL TECHNICAL INFORMATION SERVICE, SPRINGFIELD, VIRGINIA 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	232	

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
UMTA-MA-06-0025-80-4	PB 80-205735	

4. Title and Subtitle	5. Report Date
INCREASED RAIL TRANSIT VEHICLE CRASHWORTHINESS IN HEAD-ON COLLISIONS Volume IV: IITRAIN USERS' MANUAL	June 1980
	6. Performing Organization Code
7. Author(s) Edward E. Hahn	8. Performing Organization Report No. DOT-TSC-UMTA-80-17,IV
9. Performing Organization Name and Address IIT Research Institute* 10 West 35th Street Chicago, Illinois 60616	10. Work Unit No. (TRAIS) MA-06-0025(UM904/R0734)
	11. Contract or Grant No. DOT-TSC-1052-4
12. Sponsoring Agency Name and Address U.S. Department of Transportation Urban Mass Transportation Administration 400 Seventh Street, S.W. Washington, DC 20590	13. Type of Report and Period Covered Final Report June 1975 - June 1978 Volume IV of IV
	14. Sponsoring Agency Code UTD-30

15. Supplementary Notes

*under contract to: U.S. Department of Transportation
Research and Special Programs Administration
Transportation Systems Center
Cambridge, Massachusetts 02142

16. Abstract

As systems manager for the Urban Mass Transportation Administration (UMTA) Rail System Supporting Technology Program, the Transportation Systems Center (TSC) is conducting research and development efforts directed toward the introduction of improved technology in urban rail system applications. As part of this program, TSC is conducting analytical and experimental studies toward improved safety in urban rail systems. A specific goal in this area of safety is to reduce the number of injuries that may result from the collision of two trains.

In this report, Volume IV, a computer code for the simulated crash of two railcar consists is described. The code is capable of simulating the mechanics of head-on impact of two consists on straight level track. The simulation is limited to two dimensions, namely a vertical plane containing the length of the track. The user can model the individual car components or cars in a complex or as simple a manner as is warranted by the simulation results desired. Although specifically developed for transit cars, the code can also be used to simulate freight trains or intercity passenger trains.

This final report is comprised of three other volumes which are: Volume I: Initial Impact (UMTA-MA-06-0025-80-1); Volume II: Primary Collision (UMTA-MA-06-0025-80-2); and Volume III: Guidelines for Evaluation and Development of New Railcar Designs (UMTA-MA-06-0025-80-4).

17. Key Words	18. Distribution Statement
Collisions; Commuter Rail Cars; Crashworthiness; Computer Crash Simulation; Head-On Collisions; Impacts; Railcar Crashworthiness; Rapid Transit Cars; Train Crashes; Transportation Safety Manuals;	Available to the public through the National Technical Information Service, Springfield, Virginia 22161.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	233	A11

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

PREFACE

As systems manager for the Urban Mass Transportation Administration (UMTA) Rail System Supporting Technology Program, the Transportation Systems Center (TSC) is conducting research and development efforts directed toward the introduction of improved technology in urban rail system applications. As part of this program, TSC is conducting analytical and experimental studies toward improved safety in urban rail systems. A specific goal in this area of safety is to reduce the number of injuries that may result from the collision of two trains.

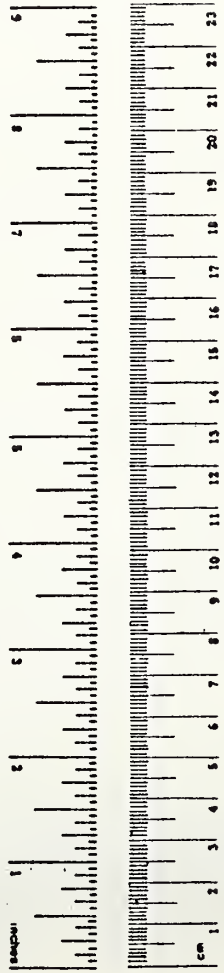
On 30 June 1975, TSC contracted with IIT Research Institute (IITRI) to perform this study to develop engineering methods and data pertaining to improved technology in urban rail systems which will lead to increased rail transit vehicle crashworthiness and passenger injury minimization. This final report is submitted in four volumes. Part 1 describes the results of Task 1 which is concerned with the initial impact of two transit cars. The results of Task 2 which is concerned with the primary collision of two impacting transit car consists are described in Part 2. Part 3 describes the results of Tasks 3 and 4 of this study which are concerned with prediction of passenger injury and guidelines for evaluation of railcar designs. The final volume is a manual containing a description of the organization and use of the IITRAIN computer code which was developed as a tool to help meet the goals of this contract.

Major IITRI contributors to the work covered in this report include Edward E. Hahn, Arne H. Wiedermann, Anatole Longinow, Robert W. Bruce and Steven C. Walgrave. The author takes this opportunity to acknowledge the contributions to this report made by Dr. A. Robert Raab, Mr. Samuel Polcari, Dr. Ming Chen, Mr. George Neat and Mr. Ronald Madigan of the U.S. Department of Transportation, TSC, Cambridge, Massachusetts.

METRIC CONVERSION FACTORS

Approximate Conversions to Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
	LENGTH			
in	inches	2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
	AREA			
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
	MASS (weight)			
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t
	VOLUME			
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.96	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
	TEMPERATURE (exact)			
	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C



Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
	LENGTH			
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
	AREA			
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
	MASS (weight)			
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	
	VOLUME			
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	35	cubic feet	ft^3
m^2	cubic meters	1.3	cubic yards	yd^2
	TEMPERATURE (exact)			
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F

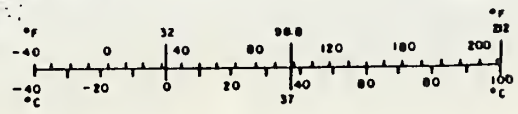


TABLE OF CONTENTS

	Page
1. INTRODUCTION	1
1.1 Program Objectives	1
1.2 Background	4
2. IITRAIN COMPUTER CODE	6
2.1 Program Organization	6
2.2 Capabilities and Limitations	6
2.3 IITRAIN Connecting Elements	8
3. IITRAIN INPUT AND OUTPUT FORMAT	32
4. SAMPLE PROBLEM	55
5. IITRAIN CODE LISTING	111
5.1 IITRI Subprograms	111
5.2 COMMON Storage Blocks	113
5.3 Program Listings	117
6. REFERENCES	225

LIST OF ILLUSTRATIONS

1. IITRAIN Control Diagram	7
2. Sample Connection Among Masses	9
3. Element Type 1	11
4. Element Type 2	12
5. Element Type 3	13
6. Element Type 4	14
7. Element Type 5	15
8. Element Type 6	16
9. Element Type 7	17
10. Element Type 8	18
11. Element Type 9	19
12. Element Type 10	20
13. Element Type 11	21
14. Element Type 12	22
15. Element Type 13	23
16. Element Type 14	24
17. Element Type 15	25

LIST OF ILLUSTRATIONS (Concl)

	Page
18. Element Type 16	26
19. Element Type 18	27
20. Element Type 19	28
21. Element Type 20	29
22. Element Type 21	30
23. Element Type 22	31
24. Sample Problem Consist Models	56

LIST OF TABLES

1. Program Modules	7
2. IITRAIN Elements	10
3. IITRAIN Input Format	33
4. Mass Data	57
5. Connection Point Data	59
6. Physical Properties of Elements	69
7. IITRAIN Deck Data	76
8. Sample Problem Computer Output	87
9. IITRAIN Programs	111
10. TIME COMMON Block	113
11. STAT COMMON Block	113
12. MASS COMMON Block	114
13. ELEM COMMON Block	114
14. CONN COMMON Block	114
15. GRAV COMMON Block	115
16. WRITE COMMON Block	115
17. FORC COMMON Block	115
18. SAVE COMMON Block	116
19. PAIR COMMON Block	116

1. INTRODUCTION

The collision of two consists of transit cars can be broken into three separate, but interdependent, phenomena: initial impact, primary collision, and secondary collision. Initial impact is concerned with the mechanics of the initial impact of the leading cars of two consists. The interaction of all of the cars and car components of two impacting consists comprise the primary collision. Secondary collisions include the interaction of passengers with the car components, passengers with passengers and passengers with other loose objects. This final report, submitted in four volumes, describes the results of the IIT Research Institute (IITRI) program which is concerned with the collision of transit car consists on straight level track. Part 1 of the final report is concerned with the initial impact of the leading cars of two consists. The results of the study of the primary collision of two impacting consists are given in Part 2, and Part 3 is concerned with secondary collisions including the prediction of passenger injury and guidelines for evaluation of new railcar designs. The final volume is a manual containing a description of the organization and use of the IITRAIN computer code which was developed as a tool to help meet the goals of this contract.

1.1 Program Objectives

The program objectives, as taken from the contract, are restated here.

Item 1a: Formulate an analytical model in two dimensions, longitudinal and vertical, of the leading cars of two impacting consists in sufficient detail to examine the mechanics of head-on initial impact on straight track. This model will include the distribution of mass in the cars as well as the nonlinear force-deformation relationships existing among major structural subassemblages. Consideration will be given to the shapes and configurations of the impacting surfaces and to the forces generated by

the impact. The model shall be capable of establishing the critical parameters which govern whether the cars crush, displace vertically and override, or crush with subsequent override.

Item 1b: Utilize the above analytical model of initial impact to assess impact controlling devices currently in service, such as anticlimbers, couplers and draft gears of various designs. This assessment shall uncover the critical parameters of such devices which govern whether the cars crush, displace vertically and override or crush with subsequent override. The contractor shall develop recommendations concerning future directions of effort in design of impact controlling devices which would be particularly pertinent to crashworthiness goals.

Item 1c: Develop an experimental test plan for the evaluation of the strength and effectiveness of future designs for impact controlling devices. These tests are to assure that the forces generated during impact do not produce structural failure of the impact controlling device or vertical misalignment and override of the car body. The test plan is to be sufficiently detailed so that all equipment, fixtures, instrumentation and procedures are completely described.

Item 2a: Develop an analytical model in two dimensions, longitudinal and vertical, of the primary collision of two impacting consists of urban railcars of similar and different configurations. This model will include the formulation of the leading cars developed in Part 1 of this program, as well as the distributions of mass and nonlinear force-deformation relationships existing among major structural subassemblages. This model shall be capable of determining the extent of crushing and/or override suffered by the individual cars in the consists, as well as the time histories of displacement, velocity, and acceleration in both the longitudinal and vertical directions.

Item 2b: Develop methods for generating the dynamic force-deformation relationships for structural subassemblages comprising the critical modules of railcars. These methods shall include finite-element analysis, scale modeling and full-scale testing procedures including specifications for required testing equipment and instrumentation. Utilize the finite-element analytical method to generate the nonlinear force-deformation relationships among major components of a typical urban railcar.

Item 3: Develop the analytical methodology of passenger injury due to secondary collision to include modes of injury due to longitudinal, vertical, and pitching motions of the vehicles after impact. This methodology shall be capable of considering the location of the passenger prior to impact, his orientation (seated, standing, facing forward, facing sideways, facing rearward), the configuration of interior features of the cars, passengers' density, and passenger restraint. This methodology shall also be capable of determining the severity of the injury sustained by the passenger.

Item 4: Utilize the results of Items 1 through 3 to develop guidelines for the evaluation of proposed railcar designs, and guidelines for the development of new railcars. These guidelines are to be developed in parametric form, so that individual parameters may be considered and the effects of specific values assigned or computed for these parameters may be assessed. These parameters are to include:

a - the number of cars in the consist
b - operational velocity ranges
c - dimensions and weights of each car
d - placement and dimensions of windows and doors
e - placement and weights of mechanical/electrical equipment
f - interior configurations of passenger compartment
g - carbody force-deformation relationships between major structural subassemblages
h - locations of carbody centers of gravity (c.g.)

1.2 Background

The task of developing an accurate computer simulation of a head-on railcar crash poses many difficult problems due to the complexity of the railcar interactions and the lack of information on the mechanisms causing crush, override, or crush with subsequent override. The effects of coupler motions, draft gear behavior, sill flexibility, truck dynamics, braking action, rail flexibility, c.g. locations, and initial conditions for the position of all components must be accounted for in any realistic simulation. Many of these factors which affect crash dynamics are highly nonlinear and may also be very sensitive to small changes in the initial conditions just prior to impact.

Some significant research in the area of railcar crash dynamics has been conducted during the past few years. Boeing-Vertol (Ref. 1) has conducted studies funded by DOT/TSC which attempted to identify significant parameters affecting crashworthiness of rail vehicles. Locomotives, freight cars, long distance passenger cars, and urban transit cars were all considered and it was concluded that, among these, crashworthiness of the urban transit car was the area which offered the greatest probability of reasonably immediate success.

Calspan (Ref. 2) also pursued research funded by DOT/TSC in the urban railcar area. This research covered three broad categories: crashworthiness of urban railcars; state of the art crash energy management devices for urban railcars; and parametric structural studies of urban railcars.

The RPI/AAR Railroad Tank Car and Safety Research and Test Project included a preliminary study of computer simulation of vertical motion during impact conducted by J. B. Raidt (Ref. 3). The objective of this study was "to investigate the existence of relative vertical motions between cars and to determine the conditions creating potential for coupler disengagement". The computer model was validated against a test case where a loaded hopper car impacted an empty hopper car, backed up by several loaded hopper cars, at 10 mph. The measured horizontal and vertical impact forces agreed reasonably well with the computer generated forces.

Washington University (Hohenemser, Diboll, Yin and Szabo) has also developed a computer crash model (Ref. 4). The basic assumptions for this model are the same as for the Raidt model. The motion is limited to the vertical plane, car bodies are assumed to be rigid with springs representing underframe elasticity; trucks are also rigid bodies, connected to the car body with vertical springs. The entire analysis is linear except for hysteresis losses in the draft gear, friction between lading and the car bottom, and lifting of the car body from the draft gear.

None of the described models have been successful in simulating head-on railcar crashes at any significant speed, particularly with respect to predictions of override. Some probable causes of the lack of accuracy of present models in this respect include assumptions of linearity, neglect of track elasticity, lack of control of initial conditions (i.e., draft gear positions), insufficient detail in the local interaction of couplers and other contacting appurtenances and accurate representation of the input parameters for the model.

To solve or circumvent the many difficulties which arise in the simulation of railcar crashes, IITRI chose a developmental approach to the computer model formulation and implementation. For the first stage of this development, simplified computer modules were written to simulate each of the subcomponents associated with the overall model. An executive program was also written to control all calculations. This modular form of computer analysis allows ease of modification of the analytical model. The use of simplified, but realistic, subcomponent computer modules enables the completion of a running computer program at an early stage in the project.

This computer program was exercised to study railcar crash dynamics and the computer results were analyzed critically to determine which modules needed to be modified to successfully simulate a head-on crash. Modifications were carried out and the resulting simulation further evaluated until a satisfactory simulation was obtained.

2. IITRAIN COMPUTER CODE

2.1 Program Organization

A lumped mass approach to the model formulation was selected but the procedure will allow finite elements to be used if required, resulting in a "hybrid" formulation. The main program modules and their corresponding functions are shown in Table 1. Figure 1 is a control diagram showing the manner in which the various modules interact.

2.2 Capabilities and Limitations

The IITRAIN computer code is designed to simulate a system of m masses, connected by n elements, subjected to applied external forces (and moments) specified as functions of time. Each of the individual mass degrees of freedom can be constrained or given specified initial conditions. At present, motions of the masses as functions of time cannot be specified. Limitations of the number of masses, m, and the number of elements, n, are only dependent on FORTRAN dimension statements and the storage capability of the computer being used.

The program is designed so that the user has his choice of integration procedures. However, at present, the only option available is simple Euler integration. Another option provided for in the program organization is the capability of integrating the equations of motion for different masses over different time intervals.

Both printed motion and force output are available. Displacements, velocities and/or accelerations (linear and rotational) of any number of points on any number of masses (limited by FORTRAN dimension statements and computer storage limitations only) can be called out. Also the internal forces and moments in any number of connecting elements can be specified as output. No graphical output is available but the capability of adding either printer-plotter or Calcomp graphs is built into the code.

TABLE 1. PROGRAM MODULES.

Name	Function
EXEC	Controls overall program
INPT	Controls input and echo print of input
INIT	Initializes program variables
INTG	Controls integration scheme
EULR	Euler integration subroutine
FINT	Controls internal force calculation
FEXT	Controls external force calculation
ACCL	Computes accelerations
OUTP	Controls Output
FNSH	Terminates the calculations and saves information required for further processing of output

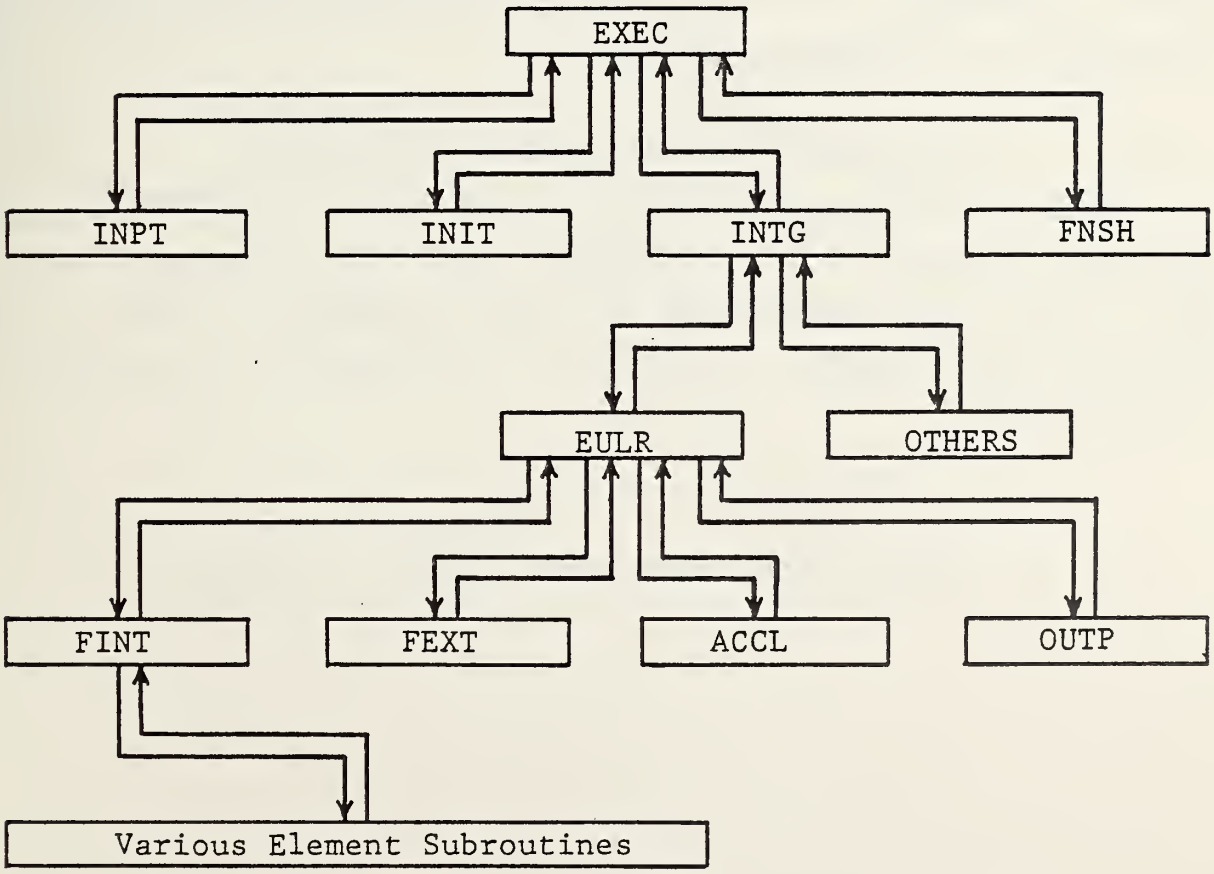


FIGURE 1. IITRAIN CONTROL DIAGRAM

Interaction among masses is provided by connecting elements. Commecting points on the masses are specified in a local coordinate system with origin at the mass c.g. A single connecting element can connect up to three masses at up to three connecting points per mass. Figure 2 is a schematic showing a possible system of three masses connected by a single element.

There are two general classifications of element types which must be considered; deformable elements such as linear springs or elastic-plastic beams and constraint elements such as pinned joints. The internal forces in deformable elements can be determined from the element properties and the state variables of the masses to which it is attached while the constraint element internal forces are determined from the kinematic relationships expressing the constraint imposed between masses.

There are 21 connecting elements available in the IITRAIN computer code. These elements are listed in Table 2. Element types 6 through 10 in Table 2 are constraint elements while the remaining are deformable elements.

2.3 IITRAIN Connecting Elements

Figures 3 through 23 show schematics of the 21 connecting elements available in the IITRAIN code. The required data to specify the mass connections of these elements as well as the physical data required to describe the elements are included in these figures.

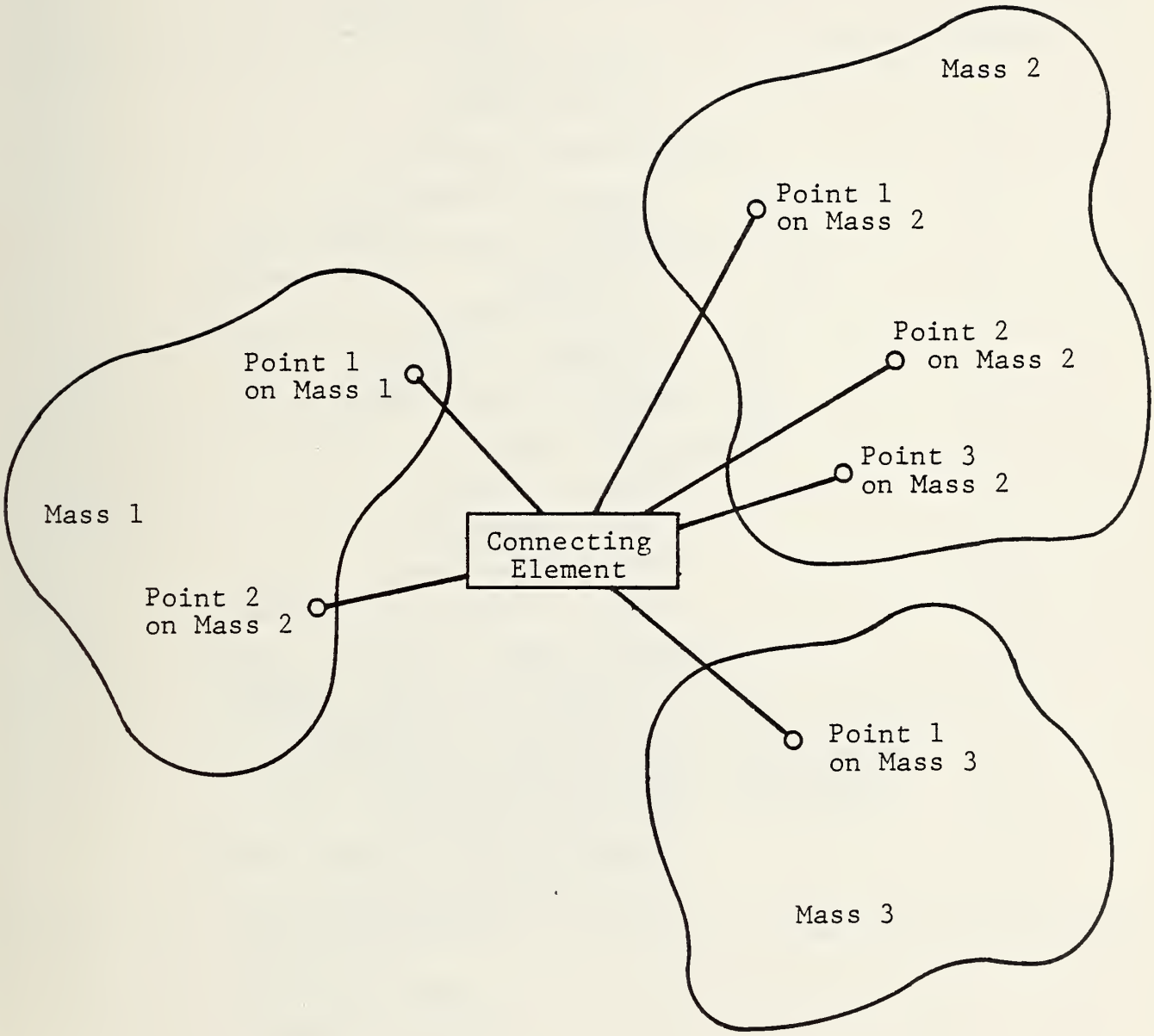
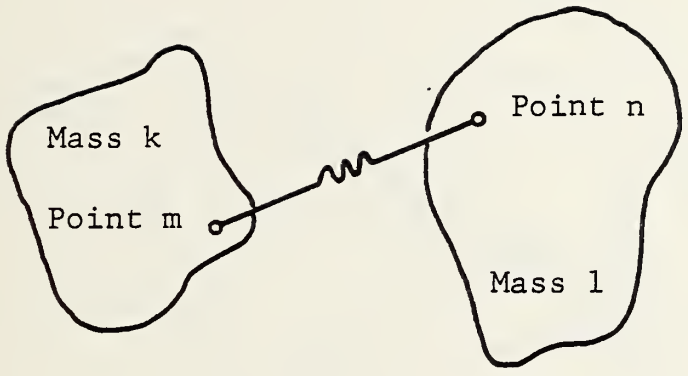


FIGURE 2. SAMPLE CONNECTION AMONG MASSES

TABLE 2.—IITRAIN ELEMENTS.

Type	Description
1	Linear spring
2	Linear dashpot
3	Torsional spring
4	Torsional dashpot
5	Elastic-plastic beam
6	Pin joint
7	Slider joint
8	Sliding pin joint
9	Double slider joint
10	Rigid joint
11	Type 1 coupling
12	Type 2 coupling
13	Type 3 draft gear
14	Type 3 coupler end element
15	Type 1 anticlimber
16	Nonlinear torsional spring
18	Wheel-rail interaction
19	Nonlinear spring
20	Nonlinear dashpot
21	Special linear spring
22	Tapered beam element



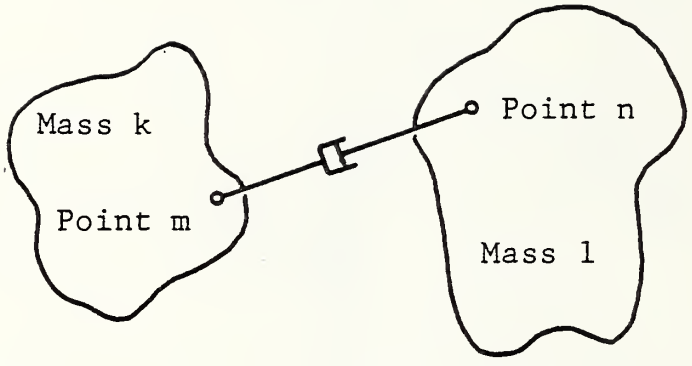
Element Type 1 Linear Spring

Connects point m on mass k to point n on mass 1
m specified by local x and y coordinates
n specified by local x and y coordinates

Element Physical Properties

spring constant
free length
fracture load
damping constant

FIGURE 3. ELEMENT TYPE 1



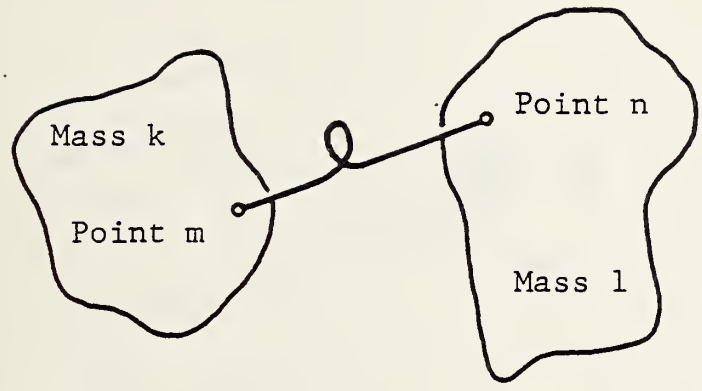
Element Type 2 Linear Dashpot

Connects point m on mass k to point n on mass 1
m specified by local x and y coordinates
n specified by local x and y coordinates

Element Physical Properties

damping constant

FIGURE 4. ELEMENT TYPE 2



Element Type 3 Torsional Spring

Connects point m on mass k to point n on mass l

m specified by local x and y coordinates

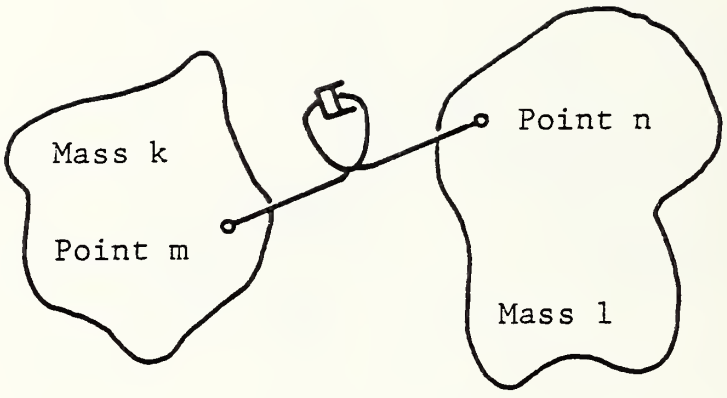
n specified by local x and y coordinates

Element Physical Properties

torsional spring constant

effective free angle length of spring modulus 360 degrees

FIGURE 5. ELEMENT TYPE 3

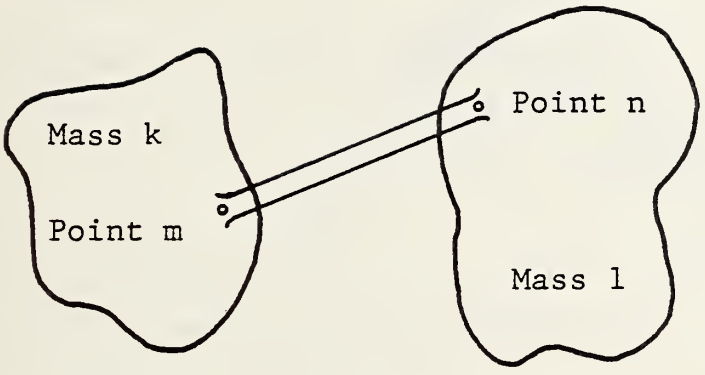


Element Type 4 Torsional Dashpot

Connects point m on mass k to point n on mass 1
m specified by local x and y coordinates
n specified by local x and y coordinates

Element Physical Properties
torsional dashpot constant

FIGURE 6. ELEMENT TYPE 4



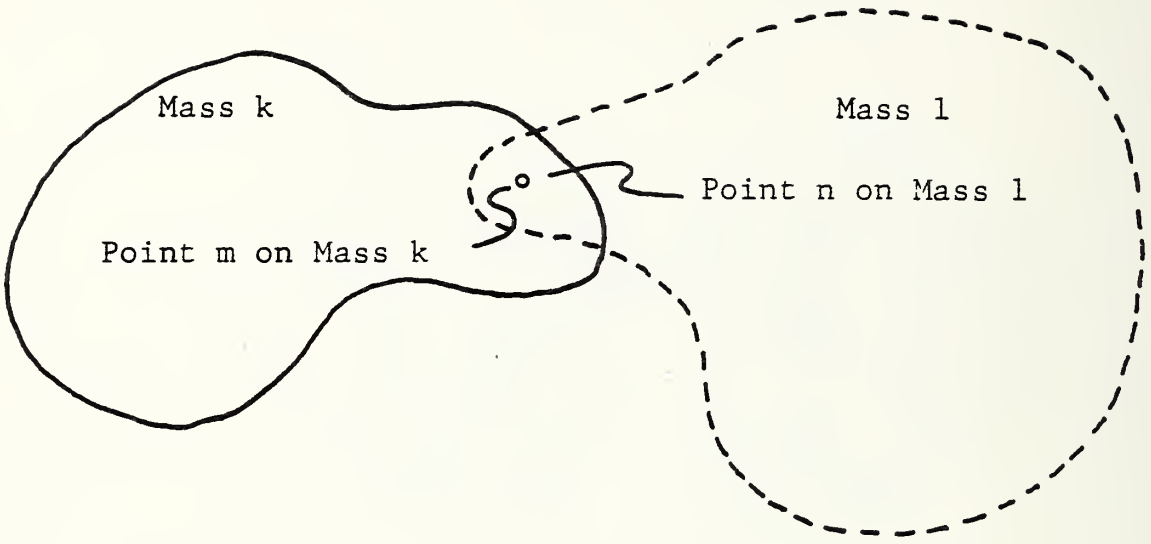
Element Type 5 Elastic-Plastic Beam

Connects point m on mass k to point n on mass 1
m specified by local x and y coordinates
n specified by local x and y coordinates

Element Physical Properties

elastic modulus
plastic modulus
yield point stress
ultimate stress
number of rectangles defining cross section
heights of rectangles
widths of rectangles
number of divisions of rectangles for numerical integration of stresses
axial damping constant
shear damping constant
angular damping constant

FIGURE 7. ELEMENT TYPE 5



Element Type 6 Pin Joint

Connects point m on mass k to point n on mass 1

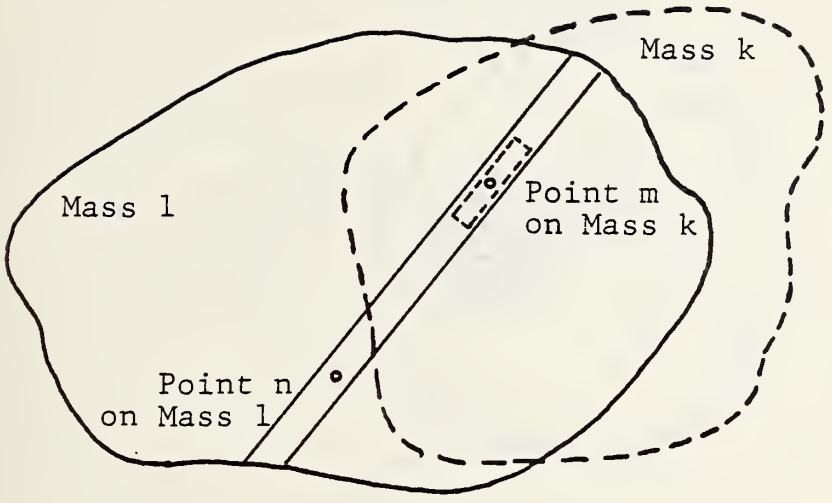
m specified by local x and y coordinates

n specified by local x and y coordinates

Element Physical Properties

friction parameter
(μR)

FIGURE 8. ELEMENT TYPE 6



Element Type 7 Slider Joint

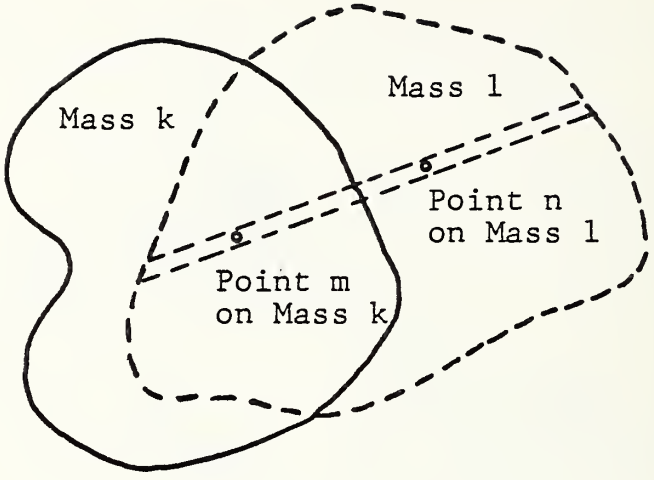
Connects slider centered at point m on mass k to slide having centerline passing through point n on mass 1

m specified by local x and y coordinates
n specified by local x and y coordinates
slider angle specified by local θ coordinate
slide angle specified by local θ coordinate

Element Physical Properties

slider length
slider width
coefficient of friction

FIGURE 9. ELEMENT TYPE 7



Element Type 8 Sliding Pin Joint

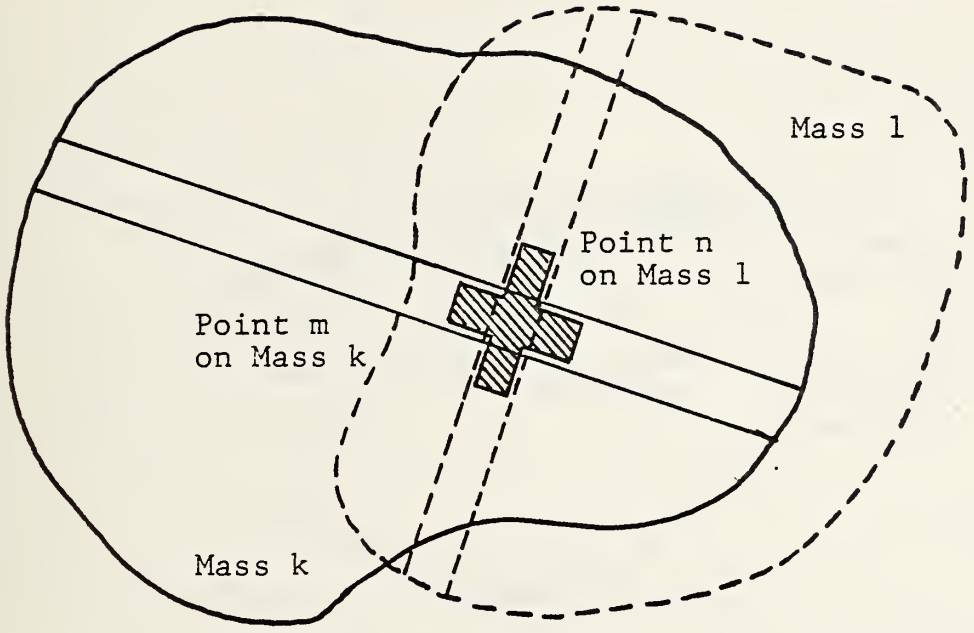
Connects pin at point m on mass k to slide having centerline passing through point n on mass 1

m specified by local x and y coordinates
n specified by local x and y coordinates
slide angle specified by local θ coordinate

Element Physical Properties

coefficient of friction

FIGURE 10. ELEMENT TYPE 8



Element Type 9 Double Slider Joint

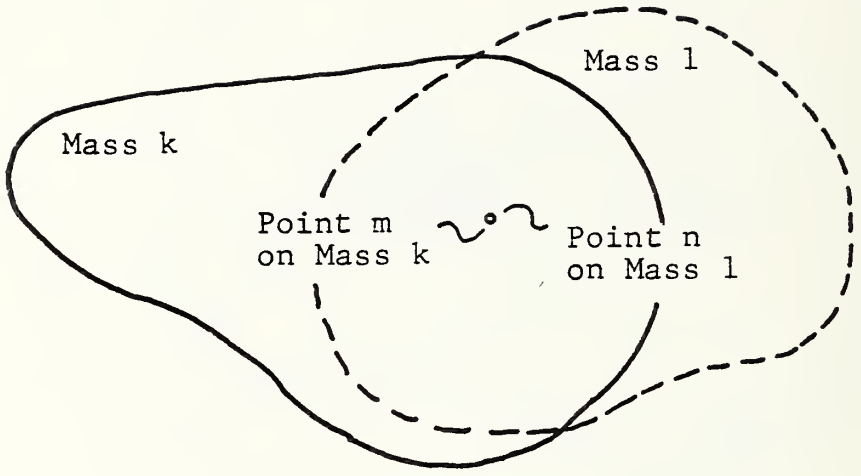
Connects slide on mass k having centerline passing through point m to slide on mass 1 having centerline passing through point n

m specified by local x and y coordinates
n specified by local x and y coordinates
slide angles specified by local θ coordinates

Element Physical Properties

slider length, x motion
slider width, x motion
coefficient of friction, x motion
slider length, y motion
slider width, y motion
coefficient of friction, y motion

FIGURE 11. ELEMENT TYPE 9

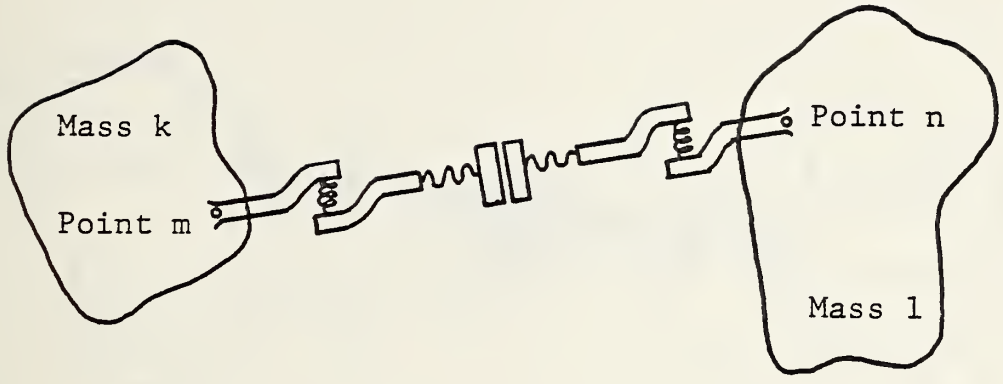


Element Type 10 Rigid Joint

Connects point m on mass k to point n on mass 1
m specified by local x and y coordinates
n specified by local x and y coordinates

Element Physical Properties
none required

FIGURE 12. ELEMENT TYPE 10



Element Type 11 Type 1 Coupling

Connects point m on mass k to point n on mass 1

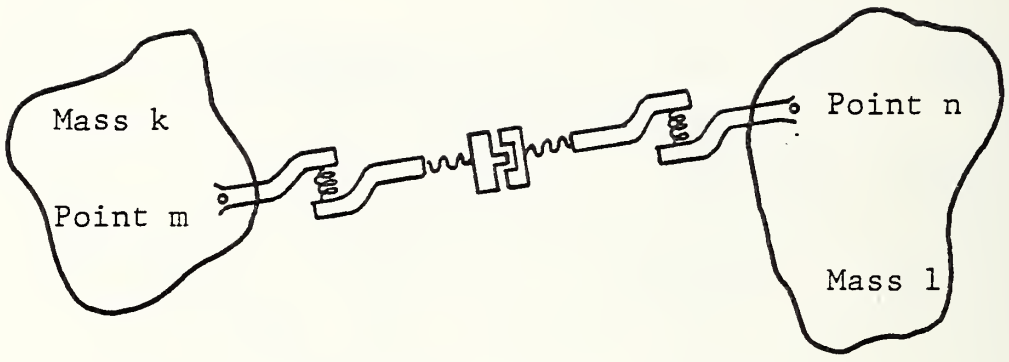
m specified by local x and y coordinates

n specified by local x and y coordinates

Element Physical Properties

draft gear spring constant, end k
draft gear spring travel, end k
car underframe spring constant, end k
draft gear hysteresis load, end k
vertical coupler spring constant, end k
vertical coupler slack, end k
free length, end k
coupler height, end k
draft gear spring constant, end 1
draft gear spring travel, end 1
car underframe spring constant, end 1
draft gear hysteresis load, end 1
vertical coupler spring constant, end 1
vertical coupler slack, end 1
free length, end 1
coupler height, end 1
coefficient of friction
total coupler horizontal slack
initial coupler misalignment

FIGURE 13. ELEMENT TYPE 11



Element Type 12 Type 2 Coupling

Connects point m on mass k to point n on mass 1

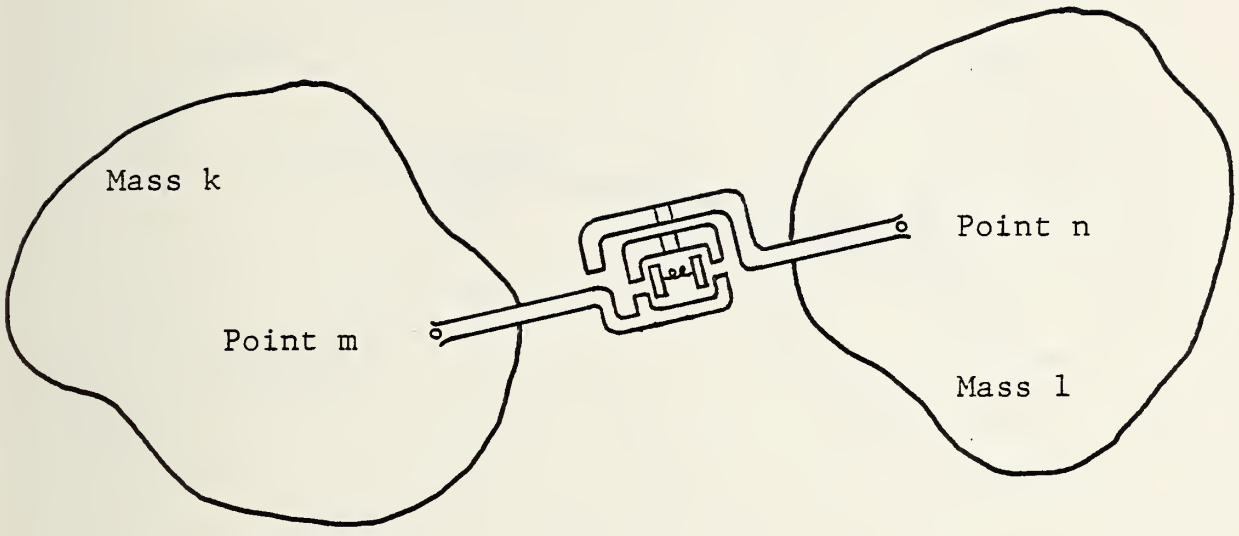
m specified by local x and y coordinates

n specified by local x and y coordinates

Element Physical Properties

draft gear spring constant, end k
draft gear spring travel, end k
car underframe spring constant, end k
draft gear hysteresis load, end k
vertical coupler spring constant, end k
vertical coupler slack, end k
free length, end k
coupler height, end k
draft gear spring constant, end 1
draft gear spring travel, end 1
car underframe spring constant, end 1
draft gear hysteresis load, end 1
vertical coupler spring constant, end 1
vertical coupler slack, end 1
free length, end 1
coupler height, end 1
coefficient of friction
vertical shear force

FIGURE 14. ELEMENT TYPE 12



Element Type 13 Type 3 Draft Gear

Connects point m on mass k to point n on mass 1

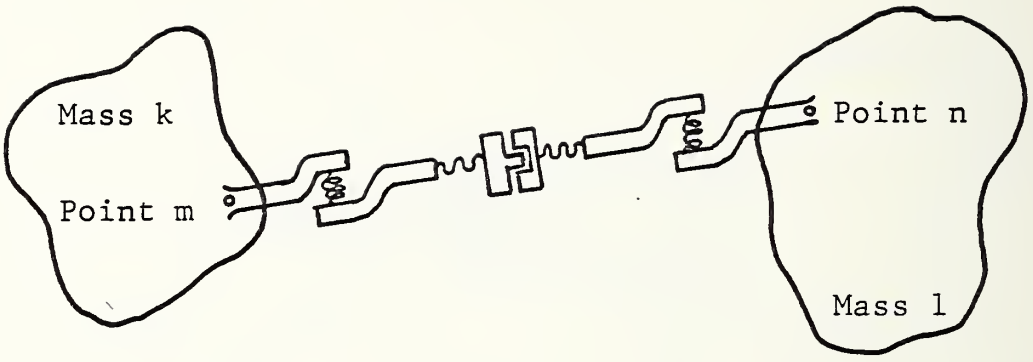
m specified by local x and y coordinates

n specified by local x and y coordinates

Element Physical Properties

draft gear spring constant
draft gear spring travel
spring constant after bottoming
draft gear hysteresis load
shear pin fracture load
postshear free travel
fracture load
drag load

FIGURE 15. ELEMENT TYPE 13



Element Type 14 Type 3 Coupler End Element

Connects point m on mass k to point n on mass 1

m specified by local x and y coordinates

n specified by local x and y coordinates

coupler direction specified by local θ coordinates

Element Physical Properties

vertical coupler spring constant, end k

horizontal coupler spring constant, end k

free length, end k

coupler height, end k

vertical coupler slack, end k

vertical coupler spring constant, end 1

horizontal coupler spring constant, end 1

free length, end 1

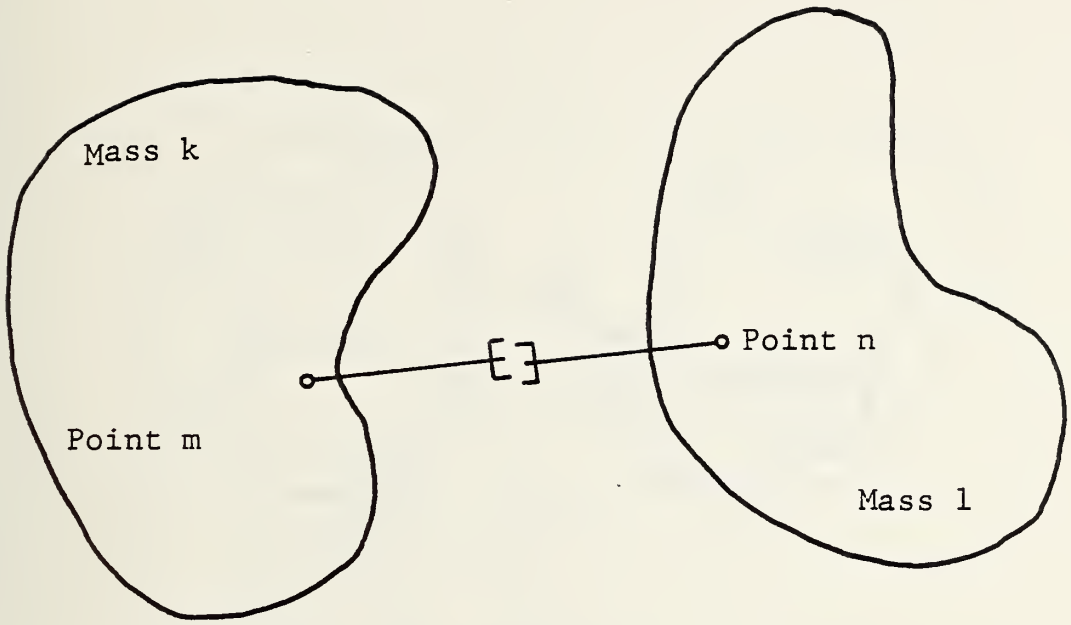
vertical coupler slack, end 1

coefficient of friction

total coupler horizontal slack

initial coupler misalignment

FIGURE 16. ELEMENT TYPE 14



Element Type 15 Anticlimber

Connects point m on mass k to point n on mass 1

m specified by local x and y coordinates

n specified by local x and y coordinates

anticlimber direction specified by local θ coordinates

Element Physical Properties

vertical, horizontal and torsional elastic spring constants, ends k and 1

vertical, horizontal and torsional plastic spring constants, ends k and 1

vertical, horizontal and torsional yield deflections, ends k and 1

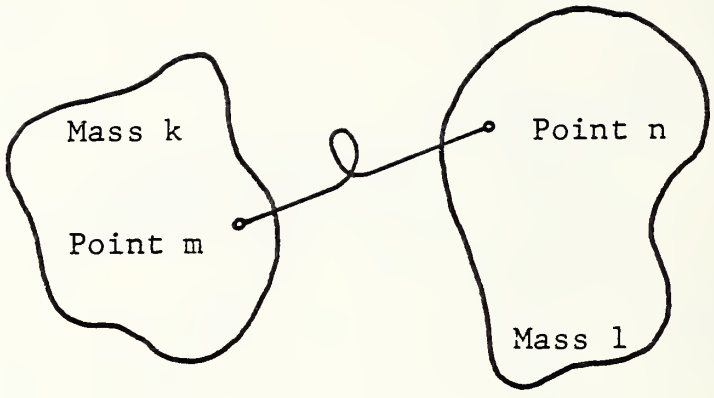
vertical, horizontal and torsional rupture deflections, ends k and 1

anticlimber height, ends k and 1

anticlimber length, ends k and 1

initial anticlimber misalignment

FIGURE 17. ELEMENT TYPE 15



Element Type 16 Nonlinear Torsional Spring

Connects point m on mass k to point n on mass 1

m specified by local x and y coordinates

n specified by local x and y coordinates

Element Physical Properties

Extensional spring constant, deflection less than θ_1

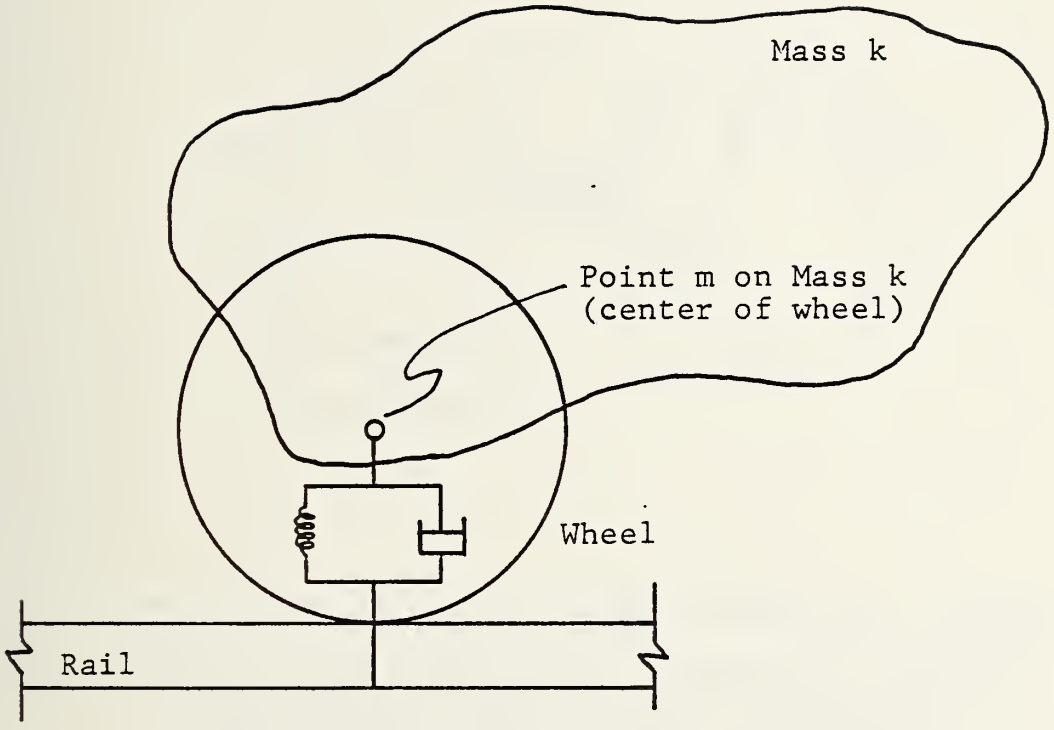
θ_1
Extensional spring constant, deflection less than θ_2 but greater than θ_1

θ_2
Compressive spring constant, compression less than θ_3

θ_3
Compressive spring constant, compression less than θ_4 but greater than θ_3

θ_4
Free length modulus 360 degrees

FIGURE 18. ELEMENT TYPE 16

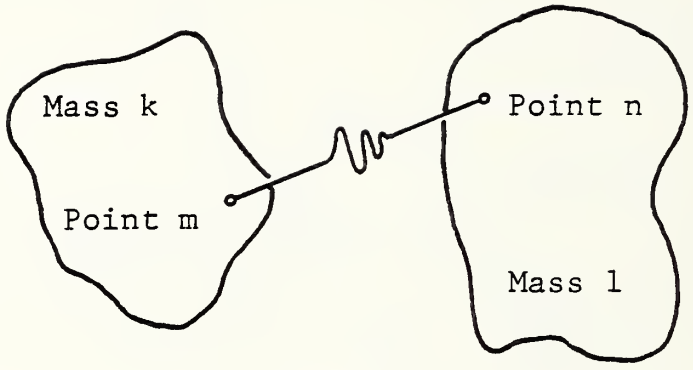


Element Type 18 Wheel-Rail Interaction

Connects point m on mass k to the rail
m specified by local x and y coordinates

Element Physical Properties
spring constants
damping constant
effective coefficient of friction
wheel radius
y rail intercept
rail angle

FIGURE 19. ELEMENT TYPE 18

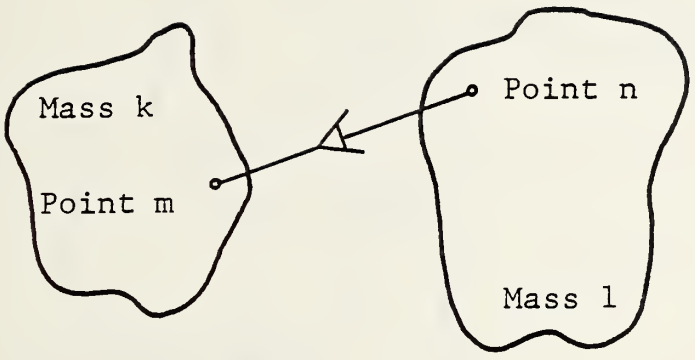


Element Type 19 Nonlinear Spring

Connects point m on mass k to point n on mass 1
m specified by local x and y coordinates
n specified by local x and y coordinates

Element Physical Properties
spring rate, compression, deflection less than δ_c
spring rate, compression, deflection greater than δ_c
spring rate, tension, deflection less than δ_t
spring rate, tension, deflection greater than δ_t
δ_c
δ_t
free length
damping constant

FIGURE 20. ELEMENT TYPE 19

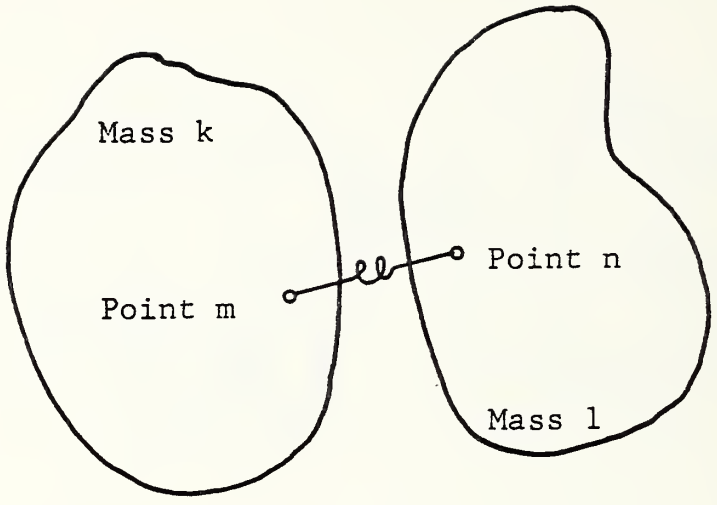


Element Type 20 Nonlinear Dashpot

Connects point m on mass k to point n on mass 1
m specified by local x and y coordinates
n specified by local x and y coordinates

Element Physical Properties
damping constant compressive velocity less than V_c
damping constant, compressive velocity greater than V_c
damping constant, extension velocity less than V_t
damping constant, extension velocity greater than V_t
V_c
V_t

FIGURE 21. ELEMENT TYPE 20



Element Type 21 Special Nonlinear Spring
(Compression Only)

Connects point m on mass k to point n on mass 1

m specified by local x and y coordinates

n specified by local x and y coordinates

Element Physical Properties

compressive spring constant, compression less than δ_c

compressive spring constant, compression greater than δ_c

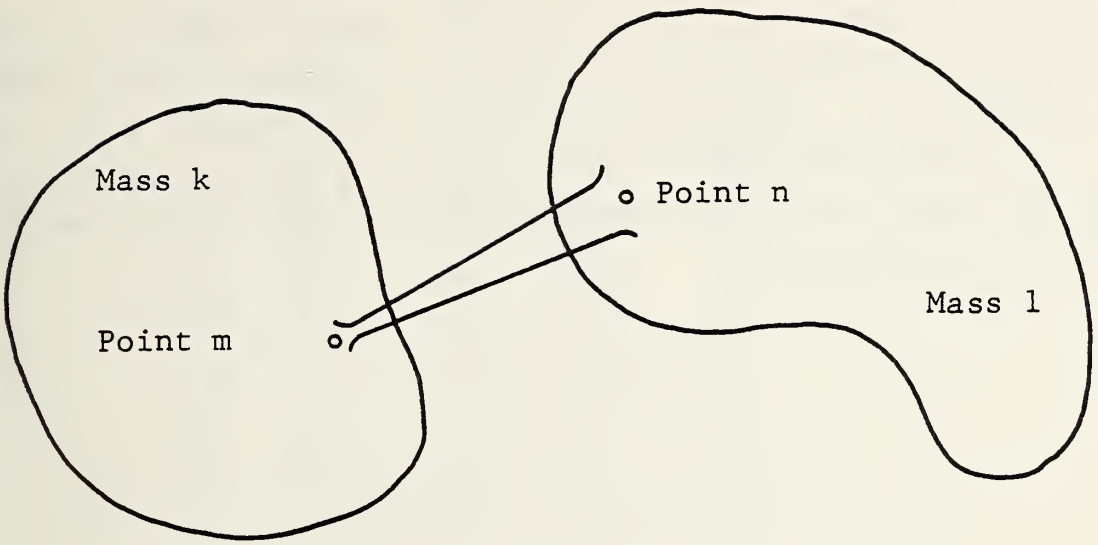
δ_c

preload at zero deflection

compressive fracture load

free length

FIGURE 22. ELEMENT TYPE 21



Element Type 22 Elastic Plastic Tapered Beam

Connects point m on mass k to point n on mass 1

m specified by local x and y coordinates

n specified by local x and y coordinates

Element Physical Properties

- elastic modulus
- plastic modulus
- yield point stress
- ultimate stress
- height of cross section, end k
- width of cross section, end k
- height of cross section, end 1
- width of cross section, end 1
- number of line division of cross section for stress-force calculations
- x damping constant
- y damping constant
- angular damping constant

FIGURE 23. ELEMENT TYPE 22

3. IITRAIN INPUT AND OUTPUT FORMAT

Table 3 contains a complete description of the input required to execute the IITRAIN computer code. The output from IITRAIN is self-explanatory. The computer program is written in such a manner that the pound-inch-second system of units must be used for all input and output data. For this reason all units in this manual are given in this system.

TABLE 3.—IITRAIN INPUT FORMAT.

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
1	1	1-5	I5	IUNIT		Read file identification for restart. If IUNIT is not blank or zero the program will restart in accordance with the restart information stored in file IUNIT.
		5-10	I5	JUNIT		Write file identification for restart. If JUNIT is blank or zero no restart file will be made.
2*	1	1-10	F10.0	TF	sec	Final time
3	1	1-80	20A4	TITL		Title (run description or identification)
4	1	1-5	I5	IM		Integration method (IM = 1 Euler Integration)
		6-20	E15.0	TD	sec	Basic time step
		21-30	E10.0	TF	sec	Final time
		31-55	5I5	IS(I)		Time step multiple for mass Class I I - 1,2,3,4,5

*This concludes the data for a restarted run (IUNIT≠0). Omit this card if IUNIT = 0.

TABLE 3 (Continued)

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
5	1	1-5	I5	NM		Number of motion outputs
		6-10	I5	NF		Number of force outputs
		11-15	I5	IWW		Number of time steps per print-out
6*	NM/8	1-3	I3	JI(I)		Mass identification for motion output I
		4-6	I3	JJ(I)		Point identification on mass JI
		7-8	I2	JTY(I)		Type of output desired
						JTY(I) = 1 displacement
						JTY(I) = 2 velocity
						JTY(I) = 3 acceleration
		9-10	I2	JDR(I)		Direction of motion
						JDR(I) = 1 x
						JDR(I) = 2 y
						JDR(I) = 3 θ
		Repeat above in columns 11-20 21-30, etc. until all NM outputs are described. If NM is greater than 8, use more cards.				

*Omit this card group if NM equal zero.

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
7*	NF/8	1-3	I3	KE(I)		Element identification for force output I
		4-6	I3	KI(I)		Mass to which element KE attaches
		7-8	I2	KJ(I)		Point on mass KI to which element KE attaches
		9-10	I2	KDR(I)		Direction of force KDR(I) = 1 x force KDR(I) = 2 y force KDR(I) = 3 θ moment
		Repeat above in columns 11-20, 21-30, etc. until all NF outputs are described. If NF is greater than 8, use more cards.				
8	1	1-10	E10.0	GA	deg	Track elevation angle, positive counterclockwise
		11-20	E10.0	GG	$in./sec^2$	Acceleration due to gravity
9	1	1-5	I5	IN		Number of masses (fixed nodes count as masses)

*Omit this card group if NF equal zero.

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
10	IN	1-10	E10.0	WT(I)	lb	Weight of Mass I
		11-20	E10.0	RI(I)	$lb\text{-}sec^2\text{-}in.$	Mass moment of inertia of Mass I
		21-25	I5	II(I)		Initial time step class for Mass I
		26-30	I5	IC(I)		Number of contact points on Mass I, IC $\geq$ 1 always

		31-50	4I5	IF(I,J)		Fixity for Mass I
						IF(I,1) = 0 x free
						= 1 x fixed
						IF(I,2) = 0 y free
						= 1 y fixed
						IF(I,3) = 0 θ free
						= 1 θ fixed
						IF(I,4) = 0 motion free at angle FA(I)
						= 1 motion fixed at angle FA(I)

		51-60	E10.0	FA(I)	deg	Angle defining fixed direction for Mass I

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
11	$\sum_{I=1}^{IN} IC(I)$	1-10	E10.0	XC(I,J)	in.	X local coordinate of Jth contact point for Mass I
		11-20	E10.0	YC(I,J)	in.	y local coordinate of Jth contact point for Mass I
		21-30	E10.0	AC(I,J)	deg	Local angle associated with Jth contact point for Mass I
12	IN	1-10	E10.0	XP(I)	in.	Initial global x position of C.G. of Mass I
		11-20	E10.0	YP(I)	in.	Initial global y position of C.G. of Mass I
		21-30	E10.0	AP(I)	rad	Initial global θ position of Mass I
		31-40	E10.0	XV(I)	in./sec	Initial global x velocity of C.G. of Mass I
		41-50	E10.0	YV(I)	in./sec	Initial global y velocity of C.G. of Mass I
		51-60	E10.0	AV(I)	rad/sec	Initial global θ velocity of Mass I
13	1	1-5	I5	IE		Number of elements

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
14	IE	1-5	I5	IT(I)		Element Type IT(I) = 1 linear spring = 2 linear dashpot = 3 torsional spring = 4 torsional dashpot = 5 elastic-plastic beam = 6 pin joint = 7 slider joint = 8 sliding pin joint = 9 double slider joint = 10 rigid joint = 11 Type 1 coupling = 12 Type 2 coupling = 13 Type 3 draft gear = 14 Type 3 coupler end element = 15 Type 1 anticlimber = 16 Nonlinear torsional spring = 18 wheel-rail interaction = 19 nonlinear spring = 20 nonlinear dashpot = 21 special linear spring = 22 special beam element
		6-10	I5	ID(I)		Identification of physical parameters describing this element

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
		11-25	315	IA(I,J)		Identification of masses attached to this element, J $\leq$ 3
		26-70	915	IP(I,J,K)		Attachment points for element I, attached to mass specified by IA(I,J), up to K=3 points per mass
15	*	1-4	I4	ITP		Element type described on this data card
		5-8	I4	IDP		Physical parameter identification for element described on this data card
		9-80	9E8.0	PP(I,IDP,ITP)		Physical parameters for element type ITP, and physical parameter identification IDP
						Element Type 1
					lb/in.	PP(1,IDP,1) = spring constant
					in.	PP(2,IDP,1) = free length
					lb	PP(3,IDP,1) = fracture load (0 if infinitely strong)
					lb sec/in.	PP(4,IDP,1) = damping constant
						PP(5 to 9,IDP,1) not used
						Element Type 2
					lb sec/in.	PP(2,IDP,2) = damping constant
						PP(2 to 9,IDP,2) not used

* The end of this card group is indicated by placing a blank card as the last card of this group. Some elements require two or three cards to specify the physical parameters. These cards will have format 10E8.0.

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
						Element Type 3
					in. lb/rad	PP(1,IDP,3) = torsional spring constant
					deg	PP(2,DIP,3) = effective free angle length of spring modulus 360°
						PP(3 to 9,IDP,3) not used
						Element Type 4
					in. lb sec/rad	PP(1,IDP,4) = torsional dashpot constant
						PP(2 to 9,IDP,4) not used
						Element Type 5
					psi	PP(1,IDP,5) = elastic modulus
					psi	PP(2,IDP,5) = plastic modulus
					psi	PP(3,IDP,5) = yield point stress
					psi	PP(4,IDP,5) = ultimate stress
						PP(5,IDP,5) = number of section blocks* defining beam cross section
					in.	PP(6,IDP,5) = h_1 - height of Section 1 sections numbered from top to bottom

*Note: Maximum Number of Sections = 7; Maximum Total Number of Divisions = 10

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
					in.	PP(7,IDP,5) = w_1 - width of Section 1
						PP(8,IDP,5) = number of equal height* divisions of cross-section 1 for stress-force calculation
						PP(9,IDP,5) PP(10,IDP,5) and PP(11,IDP,5) are repeats of 7,8 and 9 for the second section if there is one. This is continued until all sections are described. The remaining three parameters follow immediately after the last section.
					lb sec/in.	PP(,IDP,5) x damping constant
					lb sec/in.	PP(,IDP,5) y damping constant
					in lb sec/rad	PP(,IDP,5) angular damping constant
						Element Type 6
						PP(1,IDP,6) = friction parameter (μR)
						PP(2 to 9,IDP,6) not used
						Element Type 7
					in.	PP(1,IDP,7) = slider length
					in.	PP(2,IDP,7) = slider width
						PP(3,IDP,7) = coefficient of friction
						PP(4 to 9,IDP,7) not used

*Note: Maximum Number of Sections = 7; Maximum Total Number of Divisions = 10

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
						Element Type 8
					in.	PP(1,IDP,8) = coefficient of friction
						PP(2 to 9,IDP 8) not used
						Element Type 9
					in.	PP(1,IDP,9) = slider length, x motion
					in.	PP(2,IDP,9) = slider width, x motion
						PP(3,IDP,9) = coefficient of friction, x motion
					in.	PP(4,IDP,9) = slider, length, y motion
					in.	PP(5,IDP,9) = slider width, y motion
						PP(6,IDP,9) = coefficient of friction, y motion
						PP(7 to 9,IDP,9) not used
						Element Type 10
						no data required

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
						Element Type 11
					lb/in.	PP(1,IDP,11) = draft gear spring constant, end k
					in.	PP(2,IDP,11) = draft gear spring travel, end k
					lb/in.	PP(3,IDP,11) = car underframe spring constant, end k
					lb	PP(4,IDP,11) = draft gear hysteresis load, end k
					lb/in.	PP(5,IDP,11) = vertical coupler spring constant, end k
					in.	PP(6,IDP,11) = vertical coupler slack, end k
					in.	PP(7,IDP,11) = free length, end k
					in.	PP(8,IDP,11) = coupler height, end k
					in.	PP(9,IDP,11) = draft gear spring constant, end 1
					in.	PP(10,IDP,11)= draft gear spring travel, end 1
					lb/in.	PP(11,IDP,11)= car underframe spring constant, end 1

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
					lb	PP(12,IDP,11)= draft gear hysteresis load, end 1
					lb/in.	PP(13,IDP,11)= vertical coupler spring constant, end 1
					in.	PP(14,IDP,11)= vertical coupler slack, end 1
					in.	PP(15,IDP,11)= free length, end 1
					in.	PP(16,IDP,11)= coupler height, end 1
						PP(17,IDP,11)= coefficient of friction
					in.	PP(18,IDP,11)= total coupler horizontal slack
					in.	PP(19,IDP,11)= initial coupler misalignment
						Element Type 12
					lb/in.	PP(1,IDP,12) = draft gear spring constant, end k
					in.	PP(2,IDP,12) = draft gear spring travel, end k
					lb/in.	PP(3,IDP,12) = car underframe spring constant, end k

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
					lb	PP(4,IDP,12) = draft gear hysteresis load, end k
					lb/in.	PP(5,IDP,12) = vertical coupler spring constant, end k
					in.	PP(6,IDP,12) = vertical coupler slack, end k
					in.	PP(7,IDP,12) = free length, end k
					in.	PP(8,IDP,12) = coupler height, end k
					in.	PP(9,IDP,12) = draft gear spring constant, end 1
					in.	PP(10,IDP,12)= draft gear spring travel, end 1
					lb/in.	PP(11,IDP,12)= car underframe spring constant, end 1
					lb	PP(12,IDP,12)= draft gear hysteresis load, end 1
					lb/in.	PP(13,IDP,12)= vertical coupler spring constant, end 1
					in.	PP(14,IDP,12)= vertical coupler slack, end 1

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
					in.	PP(15,IDP,12)= free length, end 1
					in.	PP(16,IDP,12)= coupler height, end 1
					in.	PP(17,IDP,12)= coefficient of friction
					lb	PP(18,IDP,12)= coupler shear limit
						Element Type 13
					lb/in.	PP(1,IDP,13) = draft gear spring constant
					in.	PP(2,IDP,13) = draft gear spring travel
					lb/in.	PP(3,IDP,13) = spring constant after draft gear bottoming out
					lb	PP(4,IDP,13) = draft gear hysteresis load
					lb	PP(5,IDP,13) = shear pin fracture load
					in.	PP(6,IDP,13) = post-shear free travel
					lb	PP(7,IDP,13) = fracture load
					lb	PP(8,IDP,13) = drag load

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
						Element Type 14
					lb/in.	PP(1,IDP,14) = vertical coupler spring constant, end k
					lb/in.	PP(2,IDP.14) = horizontal coupler spring constant, end k
					in.	PP(3,IDP,14) = free length, end k
					in.	PP(4,IDP,14) = coupler height, end k
					in.	PP(5,IDP,14) = vertical coupler slack, end k
					lb/in.	PP(6,IDP,14) = vertical coupler spring constant, end 1
					lb/in.	PP(7,IDP,14) = horizontal coupler spring constant, end 1
					in.	PP(8,IDP,14) = free length, end 1
					in.	PP(9,IDP,14) = coupler height, end 1
					in.	PP(10,IDP,14)= horizontal coupler spring constant, end 1
						PP(11,IDP,14)= coefficient of friction
					in.	PP(12,IDP,14)= total coupler horizontal slack
					in.	PP(13,IDP,14)= initial coupler misalignment

Card Group	No.of Cards	Columns	Format	Program Name	Units	Description
						Element Type 15
					lb/in.	PP(1,IDP,15) = vertical elastic spring constant, end k
					lb/in.	PP(2,IDP,15) = vertical plastic spring constant, end k
					in.	PP(3,IDP,15) = vertical yield deflection, end k
					in.	PP(4,IDP,15) = vertical rupture deflection, end k
					lb/in.	PP(5,IDP,15) = horizontal elastic spring constant, end k
					lb/in.	PP(6,IDP,15) = horizontal plastic spring constant, end k
					in.	PP(7,IDP,15) = horizontal yield deflection end k
					in.	PP(8,IDP,15) = horizontal rupture deflection, end k
					in.-lb/rad.	PP(9,IDP,15) = torsional elastic spring constant, end k
					in.-lb/rad.	PP(10,IDP,15)= torsional plastic spring constant, end k

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
					rad.	PP(11,IDP,15)= torsional yield deflection, end k
					rad.	PP(12,IDP,15)= torsional rupture deflection, end k
					in.	PP(13,IDP,15)= anticlimber height, end k
					lb/in.	PP(14,IDP,15)= vertical elastic spring constant, end 1
					lb/in.	PP(15,IDP,15)= vertical plastic spring constant, end 1
					in.	PP(16,IDP,15)= vertical yield deflection, end 1
					in.	PP(17,IDP,15)= vertical rupture deflection, end 1
					lb/in.	PP(18,IDP,15)= horizontal elastic spring constant, end 1
					lb/in.	PP(19,IDP,15)= horizontal plastic spring constant, end 1
					in.	PP(20,IDP,15)= horizontal yield deflection, end 1
					in.	PP(21,IDP,15)= horizontal rupture deflection, end 1

Card Group	No. of Cards	Columns	Program Name	Units	Description
				lb/rad.	PP(22,IDP,15)= torsional elastic spring constant, end 1
				lb/rad.	PP(23,IDP,15)= torsional plastic spring constant, end 1
				rad.	PP(24,IDP,15)= torsional yield deflection, end 1
				rad.	PP(25,IDP,15)= torsional rupture deflection, end 1
				in.	PP(26,IDP,15)= anticlimber height, end 1
				in.	PP(27,IDP,15)= initial anticlimber misalignment
				in.	PP(28,IDP,15)= length of anticlimber, end k
				in.	PP(29,IDP,15)= length of anticlimber, end 1
					Element Type 16
				in.lb/rad	PP(1,IDP,16) = extensional spring constant, deflection less than θ_1
				rad	PP(2,IDP,16) = θ_1
				in.lb/rad	PP(3,IDP,16) = extensional spring constant, deflection less than θ_2 but greater than θ_1

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
					rad	PP(4,IDP,16) = θ_2
					in.lb/rad	PP(5,IDP,16) = compressive spring constant, compression less than θ_3
					rad	PP(6,IDP,16) = θ_3
					in.lb/rad	PP(7,IDP,16) = compressive spring constant, compression less than θ_4 but greater than θ_3
					rad	PP(8,IDP,16) = θ_4
					rad	PP(9,IDP,16) = free length
						Element Type 18
					lb/in.	PP(1,IDP,18) = spring constant - deflection less than δ_L
					lb/in.	PP(2,IDP,18) = spring constant - deflection greater than δ_L
					in.	PP(3,IDP,18) = δ_L

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
					in.	PP(4,IDP,18) = wheel radius
					lb sec/in.	PP(5,IDP,18) = damping constant
						PP(6,IDP,18) = coefficient of friction
					in.	PP(7,IDP,18) = rail global y intercept
					deg	PP(8,IDP,18) = rail angle
						Element Type 19
						(Nonlinear Spring)
					lb/in.	PP(1,IDP,19) = compressive spring rate, compression less than δ_c
					lb/in.	PP(2,IDP,19) = Compressive spring rate, compression greater than δ_c
					lb/in.	PP(3,IDP,19) = tensive spring rate, extension less than δ_t
					lb/in.	PP(4,IDP,19) = tensive spring rate, extension greater than δ_t
					in.	PP(5,IDP,19) = δ_c
					in.	PP(6,IDP,19) = δ_t
					in.	PP(7,IDP,19) = free length
					lb sec/in.	PP(8,IDP,19) = damping constant
						PP(9,IDP,19) = blank

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
						Element Type 20 (Nonlinear Dashpot)
					lb-sec/in.	PP(1,IDP,20) = damping constant, compressive velocity less than V_c
					lb-sec/in.	PP(2,IDP,20) = damping constant, compressive velocity greater than V_c
					lb-sec/in.	PP(3,IDP,20) = damping constant, extension velocity less than V_t
					lb-sec/in.	PP(4,IDP,20) = damping constant, exterior velocity greater than V_t
					in./sec	PP(5,IDP,20) = V_c
					in./sec	PP(6,IDP,20) = V_t
						PP(7to9,IDP,20)=blank
						Element Type 21 (Special Nonlinear Spring)
					lb/in.	PP(1,IDP,21) = compressive spring constant, compression less than δ_c
					lb/in.	PP(2,IDP,21) = compressive spring constant, compression greater than δ_c
					in.	PP(3,IDP,21) = δ_c

TABLE 3 (Concluded)

Card Group	No. of Cards	Columns	Format	Program Name	Units	Description
					lb	PP(4,IDP,21) = preload at zero deflection
					lb	PP(5,IDP,21) = compressive fracture load
					in.	PP(6,IDP,21) = free length
						PP(7to9,IDP,21)= blank
						Element Type 22 (Elastic Plastic Tapered Beam)
					psi	PP(1,IDP,22) = elastic modulus
					psi	PP(2,IDP,22) = plastic modulus
					psi	PP(3,IDP,22) = yield point stress
					psi	PP(4,IDP,22) = ultimate stress
					in.	PP(5,IDP,22) = height of cross-section, end k
					in.	PP(6,IDP,22) = width of cross-section, end k
					in.	PP(7,IDP,22) = height of cross-section, end l
					in.	PP(8,IDP,22) = width of cross-section, end l
						PP(9,IDP,22) = number of line division of cross-section for stress-force calculations
					lb sec/in.	PP(10,IDP,22)= x damping constant
					lb sec/in.	PP(11,IDP,22)= y damping constant
					in.lb sec/rad	PP(12,IDP,22)= angular damping constant

4. SAMPLE PROBLEM

To illustrate the use of the IITRAIN computer code a sample problem is given. A four-car unloaded transit car consist moving at 20 mph is assumed to crash into a standing, loaded two-car consist. A schematic of the two consists and the model used for the computer simulation is shown in Figure 24. Some of the versatility of the IITRAIN code is shown in the various degrees of complexity chosen for the models of the various cars in the two consists. The striking cars, cars 1 and 5, are modeled with nine masses and 26 connecting elements each. Five masses and 18 elements are used for car 2 while only three masses and 14 elements model car 6. Finally cars 3 and 4 are simply modeled with a single mass and two elements each. Interaction between the cars is provided with two elements for each set of adjacent cars.

Inertia data for the various masses comprising the model are given in Table 4 along with the initial positions of the masses. Table 5 contains the connection point data for the 102 connecting elements. The physical properties of the various connecting elements are presented in Table 6. A listing of the IITRAIN data deck is given in Table 7 and the IITRAIN computer output is contained in Table 8. The reader is referred to Parts 1 and 2 of this final report for interpretation of computer results.

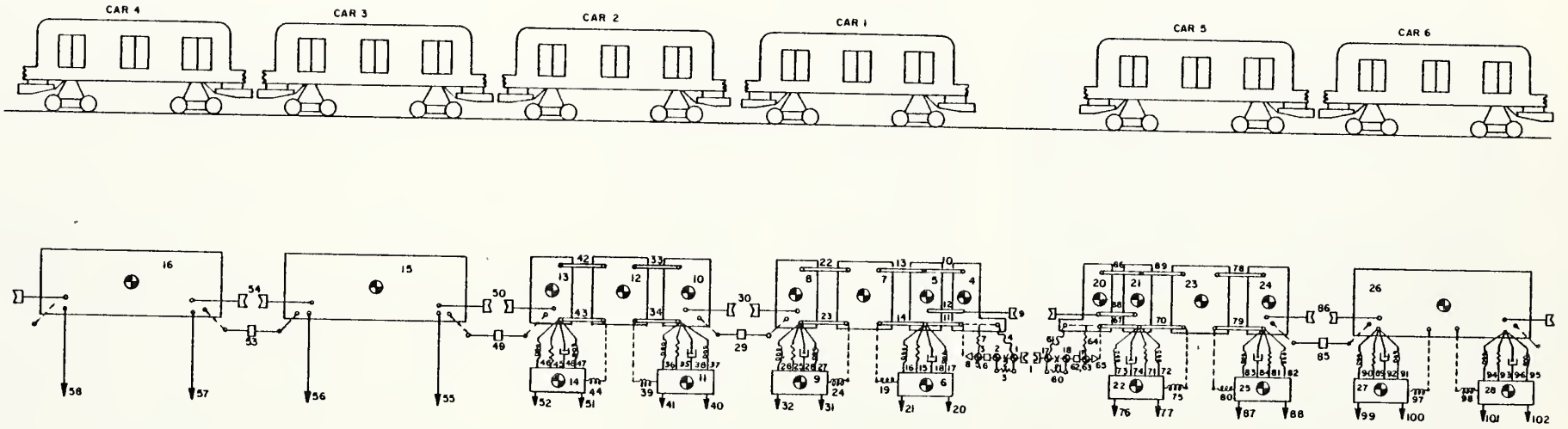


FIGURE 24. SAMPLE PROBLEM CONSIST MODELS

TABLE 4.—MASS DATA

Description	Mass	Weight (lb)	Inertia ($lb\text{-}sec^2\text{-}inch$)	Global* X-Position (inch)	Global* Y-Position (inch)
Car 1					
Coupler end mass	1	75	60	- 8.350	31.600
Draft gear yoke mass	2	90	70	- 25.400	31.600
Draft gear housing mass	3	150	100	- 40.000	31.600
Front car end mass	4	5,595	3,000	- 42.500	58.400
Front mass over body bolster	5	2,230	2,000	- 109.880	80.000
Front truck assembly mass	6	12,700	44,200	- 110.510	18.000
Center body mass	7	20,350	1,158,100	- 415.880	66.800
Rear body mass	8	7,825	29,750	- 775.063	64.556
Rear truck assembly mass	9	12,700	44,200	- 721.250	18.000
Car 2					
Front body mass	10	7,825	29,750	- 888.437	64.556
Front truck assembly mass	11	12,700	44,200	- 942.260	18.000
Center body mass	12	20,350	1,158,100	-1,247.630	66.800
Rear body mass	13	7,825	29,750	-1,606.813	64.556
Rear truck assembly mass	14	12,700	44,200	-1,553.000	18.000
Car 3					
Car mass	15	61,400	7,310,826	-2,079.380	46.040
Car 4					
Car Mass	16	61,400	7,310,826	-2,911.130	46.040

* Global positions are measured from rail level and from the initial position of the impacting coupler faces.

TABLE 4 (Concluded)

Description	Mass	Weight (lb)	Inertia ($lb\text{-}sec^2\text{-}inch$)	Global* X-Position (inch)	Global* Y-Position (inch)
Car 5					
Coupler end mass	17	75	60	8.350	31.600
Draft gear yoke mass	18	90	70	25.400	31.600
Draft gear housing mass	19	150	100	40.000	31.600
Front car end mass	20	6,180	4,000	44.250	59.670
Front mass over body bolster	21	3,230	3,000	109.880	77.480
Front truck assembly mass	22	12,700	44,200	110.510	18.000
Center body mass	23	47,923	253,200	415.880	69.710
Rear body mass	24	9,408	35,000	773.500	65.790
Rear truck assembly mass	25	12,700	44,200	721.250	18.000
Car 6					
Car body mass	26	66,739	8,856,844	1,247.630	68.600
Front truck assembly mass	27	12,700	44,200	942.260	18.000
Rear truck assembly mass	28	12,700	44,200	1,553.000	18.000

TABLE 5. - CONNECTION POINT DATA.

Connection Description	Element	Type	Local* X-Position (inch)	Local* Y-Position (inch)	Angle (deg)
Car 1					
• Mass 1 - Coupler End Mass					
Coupling between coupler faces	1	Coupler end	7.350	0.000	0.
Pin between coupler end and draft gear yoke	2	Pin joint	- 8.700	0.000	
Coupler leveler spring	3	Special spring (Type 1)	- 8.150	- 6.300	
Interference between coupler end and underside of end sill	4	Special spring (Type 2)	8.000	5.800	
• Mass 2 - Draft Gear Yoke Mass					
Pin between coupler end and draft gear yoke	2	Pin joint	8.350	0.000	
Coupler leveler spring	3	Special spring (Type 1)	7.650	- 6.300	
Draft gear connection	5	Draft gear (Type 1)	0.000	0.000	
	6	Slider joint	0.000	0.000	
• Mass 3 - Draft Gear Housing Mass					
Draft gear connection	5	Draft gear (Type 1)	0.000	0.000	
	6	Slider joint	0.000	0.000	
Rail slider connection to end sill	7	Nonlinear spring	18.000	5.300	
Draw bar and draft pocket assembly connection to car body	8	Tapered beam	0.000	0.000	
• Mass 4 - Front Car End Mass					
Draw bar and draft pocket assembly connection to car body	8	Tapered beam	-17.500	-26.800	
Rail slider connection to end sill	7	Nonlinear spring (Type 1)	20.500	-18.500	
Interference between coupler end and underside of end sill	4	Special spring (Type 2)	40.600	-10.400	

TABLE 5 (Continued)

CONNECTION POINT DATA

Connection Description	Element	Type	Local* X-Position (inch)	Local* Y-Position (inch)	Angle (deg)
End sill/anticlimber	9	Anticlimber	-17.500	-11.900	0.
Roof sill beam	10	Beam (Type 1)	0.000	86.600	
Side sill beam	11	Beam (Type 2)	0.000	-14.800	
Draft sill beam	12	Beam (Type 3)	-17.500	-14.800	
• Mass 5 - Front Mass over Body Bolster					
Roof sill beam	10	Beam (Type 1)	0.000	65.000	
Roof sill beam	13	Beam (Type 1)	0.000	65.000	
Side sill beam	11	Beam (Type 2)	0.000	-36.400	
Side sill beam	14	Beam (Type 2)	0.000	-36.400	
Draft sill beam	12	Beam (Type 3)	0.000	-36.400	
Suspension attachment at bolster	15	Linear spring	0.000	-34.000	
	16	Nonlinear spring (Type 2)			
	17	Nonlinear spring (Type 3)			
	18	Nonlinear dashpot			
• Mass 6 - Front Truck Assembly Mass					
Suspension attachment at bolster	15	Linear spring	0.630	12.750	
	16	Nonlinear spring (Type 2)			
	17	Nonlinear spring (Type 3)			
	18	Nonlinear dashpot			
Truck anchor connection	19	Nonlinear spring (Type 4)	-20.370	0.000	
Front wheel-rail interaction	20	Wheel-rail (Type 1)	41.630	-4.000	
Rear wheel-rail interaction	21	Wheel-rail (Type 1)	-40.370	-4.000	
• Mass 7 - Center Body Mass					
Roof sill beam	13	Beam (Type 1)	0.000	78.200	
	22	Beam (Type 1)	0.000	78.200	

CONNECTION POINT DATA

Connection Description	Element	Type	Local* X-Position (inch)	Local* Y-Position (inch)	Angle (deg)
Side sill beam	14	Beam (Type 2)	0.000	-23.200	
	23	Beam (Type 2)	0.000	-23.200	
Front truck anchor connection	19	Nonlinear spring (Type 4)	252.000	-48.800	
Rear truck anchor connection	24	Nonlinear spring (Type 4)	-252.000	-48.800	
• Mass 8 - Rear Body Mass					
Roof sill beam	22	Beam (Type 1)	0.000	80.440	
Side sill beam	23	Beam (Type 2)	0.000	-20.956	
Suspension attachment at bolster	25	Linear spring			
	26	Nonlinear spring Type 2)	53.183	-18.556	
	27	Nonlinear spring (Type 3)			
	28	Nonlinear dashpot			
Draw bar connection to second car	29	Draft gear (Type 2)	3.303	-32.956	
End sill/anticlimber	30	Anticlimber	3.303	-19.556	180.
• Mass 9 - Rear truck assembly mass					
Suspension attachment at bolster	25	Linear spring	- 0.630	12.750	
	26	Nonlinear spring (Type 2)			
	27	Nonlinear spring (Type 3)			
	28	Nonlinear dashpot			
Truck anchor connection	24	Nonlinear spring (Type 4)	20.370	0.000	
Front wheel-rail interaction	31	Wheel-rail (Type 1)	40.370	- 4.000	
Rear wheel-rail interaction	32	Wheel-rail (Type 1)	-41.630	- 4.000	

CONNECTION POINT DATA

Connection Description	Element	Type	Local X-Position (inch)	Local Y-Position (inch)	Angle (deg)
Car 2					
• Mass 10 - Front Body Mass					
Draw bar connection to first car	29	Draft gear (Type 2)	- 3.303	-32.956	
End sill/anticlimber	30	Anticlimber	- 3.303	-19.556	0.
Roof sill beam	33	Beam (Type 1)	0.000	80.444	
Side sill beam	34	Beam (Type 2)	0.000	-20.956	
Suspension attachment at bolster	35	Linear spring	-53.183	-18.556	
	36	Nonlinear spring (Type 2)			
	37	Nonlinear spring (Type 3)			
	38	Nonlinear dashpot			
• Mass 11 - Front Truck Assembly Mass					
Suspension attachment at bolster	35	Linear spring	0.630	12.750	
	36	Nonlinear spring (Type 2)			
	37	Nonlinear spring (Type 3)			
	38	Nonlinear dashpot			
Truck anchor connection	39	Nonlinear spring (Type 4)	-20.370	0.000	
Front wheel-rail interaction	40	Wheel-rail (Type 1)	41.630	- 4.000	
Rear wheel-rail interaction	41	Wheel-rail (Type 1)	-40.370	- 4.000	
• Mass 12 - Center Body Mass					
Roof sill beam	33	Beam (Type 1)	0.000	78.200	
	42	Beam (Type 1)	0.000	78.200	
Side sill beam	34	Beam (Type 2)	0.000	-23.200	
	43	Beam (Type 2)	0.000	-23.200	

CONNECTION POINT DATA

Connection Description	Element	Type	Local * X-Position (inch)	Local* Y-Position (inch)	Angle (deg)
Front truck anchor connection	39	Nonlinear spring (Type 4)	252.000	-48.800	
Rear truck anchor connection	44	Nonlinear spring (Type 4)	-252.000	-48.800	
• Mass 13 - Rear Body Mass					
Roof sill beam	42	Beam (Type 1)	0.000	80.444	
Side sill beam	43	Beam (Type 2)	0.000	-20.956	
Suspension attachment at bolster	45	Linear spring	- 53.180	-18.556	
	46	Nonlinear spring (Type 2)			
	47	Nonlinear spring (Type 3)			
	48	Nonlinear dashpot			
Draw bar connection to third car	49	Draft gear (Type 2)	3.300	-32.956	
End sill/anticlimber	50	Anticlimber	3.300	-19.556	180.
• Mass 14 - Rear Truck Assembly Mass					
Suspension attachment at bolster	45	Linear spring	- 0.630	12.750	
	46	Nonlinear spring (Type 2)			
	47	Nonlinear spring (Type 3)			
	48	Nonlinear dashpot			
Truck anchor connection	44	Nonlinear spring (Type 4)	20.370	0.000	
Front wheel-rail interaction	51	Wheel-rail (Type 1)	40.370	- 4.000	
Rear wheel-rail interaction	52	Wheel-rail (Type 1)	-41.630	- 4.000	

CONNECTION POINT DATA

Connection Description	Element	Type	Local* X-Position (inch)	Local* Y-Position (inch)	Angle (deg)
Car 3					
• Mass 15 - Car Mass					
Drawbar connection to car 2	49	Draft gear (Type 2)	355.900	-14.440	
Drawbar connection to car 4	53	Draft gear (Type 2)	-355.900	-14.440	
Front end sill/anticlimber	50	Anticlimber	355.900	- 1.040	0.
Rear end sill/anticlimber	54	Anticlimber	-355.900	- 1.040	180.
Front wheel-rail interaction	55	Wheel-rail (Type 2)	306.000	-32.040	
Rear wheel-rail interaction	56	Wheel-rail (Type 2)	-306.000	-32.040	
Car 4					
• Mass 16 - Car Mass					
Drawbar connection to car 3	53	Draft gear (Type 2)	355.900	-14.440	
Front end sill/anticlimber	54	Anticlimber	355.900	- 1.040	0.
Front wheel-rail interaction	57	Wheel-rail (Type 2)	306.000	-32.040	
Rear wheel-rail interaction	58	Wheel-rail (Type 2)	-306.000	-32.040	
Car 5					
• Mass 17 - Coupler End Mass					
Coupling between coupler faces	1	Coupler end	- 7.350	0.000	180.
Pin between coupler end and draft gear yoke	59	Pin joint	8.700	0.000	
Coupler leveler spring	60	Special spring (Type 1)	8.150	- 6.300	
Interference between coupler end and underside of end sill	61	Special spring (Type 2)	- 8.000	5.800	

CONNECTION POINT DATA

Connection Description	Element	Type	Local* X-Position (inch)	Local* Y-Position (inch)	Angle (deg)
• Mass 18 - Draft Gear Yoke Mass					
Pin between coupler end and draft gear yoke	59	Pin joint	- 8.350	0.000	
Coupler leveler spring	60	Special spring (Type 1)	- 7.650	- 6.300	
Draft gear connection	62	Draft gear (Type 1)	0.000	0.000	
	63	Slider joint	0.000	0.000	
• Mass 19 - Draft Gear Housing Mass					
Draft gear connection	62	Draft gear (Type 1)	0.000	0.000	
	63	Slider joint	0.000	0.000	
Rail slider connection to end sill	64	Nonlinear spring (Type 1)	-18.000	5.300	
Drawbar and draft pocket assembly connection to car body	65	Tapered beam	0.000	0.000	
• Mass 20 - Front Car End Mass					
Drawbar and draft pocket assembly connection to car body	65	Tapered beam	15.750	-28.070	
Rail slider connection to end sill	64	Nonlinear spring (Type 1)	-22.250	-19.770	
Interference between coupler end and underside of end sill	61	Special spring (Type 2)	-42.350	-11.670	
End sill/anticlimber	9	Anticlimber	15.750	-14.670	180.
Roof sill beam	66	Beam (Type 1)	0.000	85.330	
Side sill beam	67	Beam (Type 2)	0.000	-16.070	
Draft sill beam	68	Beam (Type 3)	15.750	-16.070	
• Mass 21 - Front Mass over Body Bolster					
Roof sill beam	66	Beam (Type 1)	0.000	67.520	
Roof sill beam	69	Beam (Type 1)	0.000	67.520	
Side sill beam	67	Beam (Type 2)	0.000	-33.880	
Side sill beam	70	Beam (Type 2)	0.000	-33.880	
Draft sill beam	68	Beam (Type 3)	0.000	-33.880	

CONNECTION POINT DATA

Connection Description	Element	Type	Local* X-Position (inch)	Local* Y-Position (inch)	Angle (deg)
Suspension attachment at bolster	71	Linear spring	0.000	-31.480	
	72	Nonlinear spring (Type 2)			
	73	Nonlinear spring (Type 3)			
	74	Nonlinear dashpot			
• Mass 22 - Front Truck Assembly Mass					
Suspension attachment at bolster	71	Linear spring	- 0.630	12.750	
	72	Nonlinear spring (Type 2)			
	73	Nonlinear spring (Type 3)			
	74	Nonlinear dashpot			
Truck anchor connection	75	Nonlinear spring (Type 4)	20.370	0.000	
Front wheel-rail interaction	76	Wheel-rail (Type 1)	-41.630	- 4.000	
Rear wheel-rail interaction	77	Wheel-rail (Type 1)	40.370	- 4.000	
• Mass 23 - Center Body Mass					
Roof sill beam	69	Beam (Type 1)	0.000	75.290	
	78	Beam (Type 1)	0.000	75.290	
Side sill beam	70	Beam (Type 2)	0.000	-26.110	
	79	Beam (Type 2)	0.000	-26.110	
Front truck anchor connection	75	Nonlinear spring (Type 4)	-252.000	-51.710	
Rear truck anchor connection	80	Nonlinear spring (Type 4)	252.000	-51.710	
Car 5					
• Mass 24 - Rear Body Mass					
Roof sill beam	78	Beam (Type 1)	0.000	79.205	
Side sill beam	79	Beam (Type 2)	0.000	-22.195	
Suspension attachment at bolster	81	Linear spring	-51.620	-19.795	
	82	Nonlinear spring (Type 2)			

CONNECTION POINT DATA

Connection Description	Element	Type	Local* X-Position (inch)	Local* Y-Position (inch)	Angle (deg)
	83	Nonlinear spring (Type 3)			
	84	Nonlinear dashpot			
Drawbar connection to second car	85	Draft gear (Type 2)	- 1.740	-33.195	
End sill/anticlimber	86	Anticlimber	- 1.740	-19.295	0.
• Mass 25 - Rear Truck Assembly Mass					
Suspension attachment at bolster	81	Linear spring	0.630	12.750	
	82	Nonlinear spring (Type 2)			
	83	Nonlinear spring (Type 3)			
	84	Nonlinear dashpot			
Truck anchor connection	80	Nonlinear spring (Type 4)	-20.370	0.000	
Front wheel-rail interaction	87	Wheel-rail (Type 1)	-40.370	- 4.000	
Rear wheel-rail interaction	88	Wheel-rail (Type 1)	41.630	- 4.000	
Car 6					
• Mass 26 - Car Body Mass					
Drawbar connection to car 5	85	Draft gear (Type 2)	-355.900	-37.000	
Front end sill/anticlimber	86	Anticlimber	-355.900	-23.600	180.
Front suspension attachment at bolster	89	Linear spring	-306.000	-22.600	
	90	Nonlinear spring (Type 2)	-306.000	-22.600	
	91	Nonlinear spring (Type 3)	-306.000	-22.600	
	92	Nonlinear dashpot	-306.000	-22.600	
Rear suspension attachment at bolster	93	Linear spring	306.000	-22.600	
	94	Nonlinear spring (Type 2)	306.000	-22.600	
	95	Nonlinear spring (Type 3)	306.000	-22.600	
	96	Nonlinear dashpot	306.000	-22.600	

TABLE 5 (Concluded)

CONNECTION POINT DATA

Connection Description	Element	Type	Local* X-Position (inch)	Local* Y-Position (inch)	Angle (deg)
Front truck anchor connection	97	Nonlinear spring (Type 4)	-252.000	-50.600	
Rear truck anchor connection	98	Nonlinear spring (Type 4)	252.000	-50.600	
• Mass 27 - Front Truck Assembly Mass					
Suspension attachment at bolster	89	Linear spring	- 0.630	12.750	
	90	Nonlinear spring (Type 2)	- 0.630	12.750	
	91	Nonlinear spring (Type 3)	- 0.630	12.750	
	92	Nonlinear dashpot	- 0.630	12.750	
Front truck anchor connection	97	Nonlinear spring (Type 4)	20.370	0.000	
Front wheel-rail interaction	99	Wheel-rail (Type 1)	- 41.630	- 4.000	
Rear wheel-rail interaction	100	Wheel-rail (Type 1)	40.370	- 4.000	
• Mass 28 - Rear Truck Assembly Mass					
Suspension attachment at bolster	93	Linear spring	0.630	12.750	
	94	Nonlinear spring (Type 2)	0.630	12.750	
	95	Nonlinear spring (Type 3)	0.630	12.750	
	96	Nonlinear dashpot	0.630	12.750	
Rear truck anchor connection	98	Nonlinear spring (Type 4)	- 20.370	0.000	
Front wheel-rail interaction	101	Wheel-rail (Type 1)	- 40.370	- 4.000	
Rear wheel-rail interaction	102	Wheel-rail (Type 1)	41.630	- 4.000	

TABLE 6.—PHYSICAL PROPERTIES OF ELEMENTS.

Linear Spring

spring constant	3,110.00
free length	21.04
fracture load	0.00

Elastic-Plastic Beam (Type 1)

elastic modulus	10,000,000.
plastic modulus	20,000.
yield point stress	60,000.
ultimate stress	100,000.
number of section blocks defining beam cross section	1.
h_1 - height of section 1 sections numbered from top to bottom	19.550
w_1 - width of section 1	0.676
number of equal height divisions of cross section 1 for stress-force calculation	1.
x damping constant	2,500.
y damping constant	200.
angular damping constant	20.

Elastic Plastic Beam (Type 2)

elastic modulus	10,000,000.
plastic modulus	20,000.
yield point stress	60,000.
ultimate stress	100,000.
number of section blocks defining beam cross section	1.
h_1 - height of section 1 sections numbered from top to bottom	9.790
w_1 - width of section 1	2.082
number of equal height divisions of cross section 1 for stress-force calculation	1.
x damping constant	5,500.
y damping constant	200.
angular damping constant	30.

TABLE 6 (Continued)

Elastic Plastic Beam (Type 3)	
elastic modulus	30,000,000.
plastic modulus	180,000.
yield point stress	100,000.
ultimate stress	200,000.
number of section blocks defining beam cross section	1.
h_1 - height of section 1 sections numbered from top to bottom	10.360
w_1 - width of section 1	0.776
number of equal height divisions of cross section 1 for stress-force calculation	1.
x damping constant	7,800.
y damping constant	200.
angular damping constant	40.
Pin Joint	
friction parameter (μR)	0.300
Slider Joint	
slider length	10.00
slider width	1.00
coefficient of friction	0.01
Type 3 Draft Gear (Type 1)	
draft gear spring constant	24,000.
draft gear spring travel	1.250
spring constant after draft gear bottoming out	320,000.
draft gear hysteresis load	10,000.
shear pin fracture load	150,000.
postshear free travel	1.375
fracture load	250,000.
drag load	30.

Type 3 Draft Gear (Type 2)

draft gear spring constant	12,000.
draft gear spring travel	2.500
spring constant after draft gear bottoming out	160,000.
draft gear hysteresis load	10,000.
shear pin fracture load	150,000.
postshear free travel	100.
fracture load	250,000.
drag load	30.

Type 3 Coupler End Element

vertical coupler spring constant, end k	1.
horizontal coupler spring constant, end k	360,000.
free length, end k	1.
coupler height, end k	12.
vertical coupler slack, end k	0.
vertical coupler spring constant, end l	1.
horizontal coupler spring constant, end l	360,000.
free length, end l	1.
coupler height, end l	12,300,000.
vertical coupler slack, end l	0.
coefficient of friction	0.2
total coupler horizontal slack	0.
initial coupler misalignment	0.

Type 1 Anticlimber

vertical elastic spring constant, end k	175,000.
vertical plastic spring constant, end k	1,633.
vertical yield deflection, end k	0.200
vertical rupture deflection, end k	5.
horizontal elastic spring constant, end k	4,450,000.
horizontal plastic spring constant, end k	20,620.
horizontal yield deflection, end k	0.053
horizontal rupture deflection, end k	56.
torsional elastic spring constant, end k	100.
torsional plastic spring constant, end k	25.

torsional yield deflection, end k	0.001
torsional rupture deflection, end k	1.
anticlimber height, end k	6.
vertical elastic spring constant, end 1	175,000.
vertical plastic spring constant, end 1	1,633.
vertical yield deflection, end 1	0.200
vertical rupture deflection, end 1	5.
horizontal elastic spring constant, end 1	4,450,000.
horizontal plastic spring constant, end 1	20,620.
horizontal yield deflection, end 1	0.053
horizontal rupture deflection, end 1	56.
torsional elastic spring constant, end 1	100.
torsional plastic spring constant, end 1	25.
torsional yield deflection, end 1	0.001
torsional rupture deflection, end 1	1.
anticlimber height, end 1	6.
initial anticlimber misalignment	1.500
length of anticlimber, end k	58.125
length of anticlimber, end 1	58.125

Wheel-Rail Interaction (Type 1)

spring constant - deflection less than δ_L	3,234,000.
spring constant - deflection greater than δ_L	3,234,000.
δ_L	5.
wheel radius	14.005
damping constant	1,000.
coefficient of friction	0.
rail global y intercept	0.
rail angle	0.

Wheel-Rail Interaction (Type 2)

spring constant - deflection less than δ_L	6,468,000.
spring constant - deflection greater than δ_L	6,468,000.
δ_L	5.
wheel radius	14.005
damping constant	1,000.
coefficient of friction	0.
rail global y intercept	0.
rail angle	0.

Nonlinear Spring (Type 1)

compressive spring rate, compression less than δ_c	3,000,000.
compressive spring rate, compression greater than δ_c	3,000,000.
tensive spring rate, extension less than δ_t	0.
tensive spring rate, extension greater than δ_t	0.
δ_c	2.
δ_t	2.
free length	3.
damping constant	10.

Nonlinear Spring (Type 2)

compressive spring rate, compression less than δ_c	0.
compressive spring rate, compression greater than δ_c	30,000,000.
tensive spring rate, extension less than δ_t	0.
tensive spring rate, extension greater than δ_t	30,000,000.
δ_c	3.750
δ_t	2.000
free length	15.250
damping constant	0.

Nonlinear Spring (Type 3)

compressive spring rate, compression less than δ_c	0.
compressive spring rate, compression greater than δ_c	26,890.
tensive spring rate, extension less than δ_t	0.
tensive spring rate, extension greater than δ_t	0.
δ_c	2.790
δ_t	1.000
free length	15.250
damping constant	1,180.

Nonlinear Spring (Type 4)

compressive spring rate, compression less than δ_c	500,000.
compressive spring rate, compression greater than δ_c	4,500,000.
tensive spring rate, extension less than δ_t	500,000.
tensive spring rate, extension greater than δ_t	4,500,000.
δ_c	.625
δ_t	.625
free length	33.
damping constant	8,100.

Nonlinear Dashpot

damping constant, compressive velocity less than V_c	1,180.
damping constant, compressive velocity greater than V_c	173.
damping constant, extension velocity less than V_t	1,180.
damping constant, extension velocity greater than V_t	173.
V_c	4.500
V_t	4.500

TABLE 6 (Concluded)

Special Linear Spring	
compressive spring constant, compression less than δ_c	5,000.
compressive spring constant, compression greater than δ_c	30,000,000.
δ_c	1.250
preload at zero deflection	1,250.
compressive fracture load	400,000.
free length	1.250
Special Linear Spring	
compressive spring constant, compression less than δ_c	0.
compressive spring constant, compression greater than δ_c	175,000.
δ_c	4.
preload at zero deflection	0.
compressive fracture load	700,000.
free length	10.
Elastic Plastic Tapered Beam	
elastic modulus	30,000,000.
plastic modulus	180,000.
yield point stress	100,000.
ultimate stress	200,000.
height of cross section, end k	3.273
width of cross section, end k	2.830
height of cross section, end 1	5.475
width of cross section, end 1	1.827
number of line division of cross section for stress-force calculations	2.
x damping constant	4,820.
y damping constant	150.
angular damping constant	20.

TABLE 7.—IITRAIN DECK DATA

```
                                                    COLUMN
          5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
          .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .
CARD      ....................................................................................

  1
  2   IITRAIN SAMPLE PROBLEM
  3       1 .00005          0.40          1
  4      24   16  100
  5     7    2 1 12    2 1 15    2 1 16    2 1 23    2 1 26    2 1  7    1 1 23    1 1
  6     7    3 1 12    3 1 15    3 1 16    3 1 23    3 1 26    3 1  7    1 2 23    1 2
  7     7    3 2 12    3 2 15    3 2 16    3 2 23    3 2 26    3 2  7    3 3 23    3 3
  8     1  1 1 1 29 10 2 1 49 15 1 1 53 16 1 1 85 24 4 1  9 4 4 2 30 10 1 2 86 24 5 2
  9     9  4 4 1 30 10 1 1 50 15 3 1 54 16 2 1 86 24 5 1  9 4 4 3 30 10 1 3 86 24 5 3
 10              386.088
 11      28
 12      75.          60.       1    4
 13      90.          70.       1    3
 14     150.         100.       1    2
 15    5595.        3000.       1    7
 16    2230.        2000.       1    3
 17   12700.       44200.       1    4
 18   20350.     1158100.       1    4
 19    7825.       29750.       1    5
 20   12700.       44200.       1    4
 21    7825.       29750.       1    5
 22   12700.       44200.       1    4
 23   20350.     1158100.       1    4
 24    7825.       29750.       1    5
 25   12700.       44200.       1    4
 26   61400.     7310826.       1    6
 27   61400.     7310826.       1    4
 28      75.          60.       1    4
 29      90.          70.       1    3
 30     150.         100.       1    2

          .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .
          ....................................................................................
```

TABLE 7 (Continued)

COLUMN

5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80

CARD				
31	6180.	4000.	1	7
32	3230.	3000.	1	3
33	12700.	44200.	1	4
34	47293.	253200.	1	4
35	9408.	35000.	1	5
36	12700.	44200.	1	4
37	66739.	8856844.	1	6
38	12700.	44200.	1	4
39	12700.	44200.	1	4
40	7.35	0.		
41	-8.7	0.		
42	-8.15	-6.3		
43	8.	5.8		
44	8.35	0.		
45	7.65	-6.3		
46	0.	0.		
47	0.	0.		
48	18.	5.3		
49	-17.5	-26.8		
50	20.5	-18.5		
51	40.6	-10.4		
52	-17.5	-11.9	0.0	
53	0.	86.6		
54	0.	-14.8		
55	-17.5	-14.8		
56	0.	65.		
57	0.	-36.4		
58	0.	-34.		
59	.63	12.75		
60	- 20.37	.0		

COLUMN

CARD	1–10	11–20	21–30
61	41.63	-4.	
62	- 40.37	-4.0	
63	0.	78.2	
64	0.	-23.2	
65	252.	-48.8	
66	-252.	-48.8	
67	0.	80.444	
68	0.	-20.956	
69	53.183	-18.556	
70	3.303	-32.956	
71	3.303	-19.556	180.0
72	-.63	12.75	
73	20.37	.0	
74	40.37	-4.0	
75	-41.63	-4.0	
76	-3.303	-32.956	
77	-3.303	-19.556	0.0
78	0.	80.444	
79	0.	-20.956	
80	-53.183	-18.556	
81	.63	12.75	
82	-20.37	.0	
83	41.63	-4.0	
84	-40.37	-4.0	
85	0.	78.2	
86	0.	-23.2	
87	252.	-48.8	
88	-252.	-48.8	
89	0.	80.444	
90	0.	-20.956	

COLUMN

CARD	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80
91	53.183		-18.556													
92	3.303		-32.956													
93	3.303		-19.556		180.											
94	-.63		12.75													
95	20.37		.0													
96	40.37		-4.0													
97	-41.63		-4.													
98	355.9		-14.44													
99	-355.9		-14.44													
100	355.9		-1.04		0.0											
101	-355.9		-1.04		180.0											
102	306.		-32.04													
103	-306.		-32.04													
104	355.9		-14.44													
105	355.9		-1.04		0.0											
106	306.		-32.04													
107	-306.		-32.04													
108	-7.35		0.		180.											
109	8.7		0.													
110	8.15		-6.3													
111	-8.		5.8													
112	-8.35		0.													
113	-7.65		-6.3													
114	0.		0.													
115	0.		0.													
116	-18.		5.3													
117	15.75		-28.07													
118	-22.25		-19.77													
119	-42.35		-11.67													
120	15.75		-14.67		180.0											

CARD	COLUMN		
121		85.33	
122		-16.07	
123	15.75	-16.07	
124		67.52	
125		-33.88	
126		-31.48	
127	-.63	12.75	
128	20.37	.0	
129	-41.63	-4.	
130	40.37	-4.0	
131		75.29	
132		-26.11	
133	-252.	-51.71	
134	252.	-51.71	
135		79.205	
136		-22.195	
137	-51.62	-19.795	
138	-1.74	-33.195	
139	-1.74	-19.295	0.0
140	.63	12.75	
141	-20.37	.0	
142	-40.37	-4.0	
143	41.63	-4.0	
144	-355.9	-37.	
145	-355.9	-23.6	180.0
146	-306.	-22.6	
147	306.	-22.6	
148	-252.	-50.6	
149	252.	-50.6	
150	-.63	12.75	

COLUMN

CARD	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80
181	773.5		65.79		0.0		0.0		0.0		0.0					
182	721.25		18.0		0.0		0.0		0.0		0.0					
183	1247.63		68.60		0.0		0.0		0.0		0.0					
184	942.26		18.0		0.0		0.0		0.0		0.0					
185	1553.0		18.0		0.0		0.0		0.0		0.0					
186	102															
187	14	1	1	17		1			1							
188	6	1	1	2		2			1							
189	21	1	1	2		3			2							
190	21	2	1	4		4			3							
191	13	1	2	3		3			1							
192	7	1	2	3		3			1							
193	19	1	3	4		2			2							
194	22	1	3	4		1			1							
195	15	1	4	20		4			4							
196	5	1	4	5		5			1							
197	5	2	4	5		6			2							
198	5	3	4	5		7			2							
199	5	1	5	7		1			1							
200	5	2	5	7		2			2							
201	1	1	5	6		3			1							
202	19	2	5	6		3			1							
203	19	3	5	6		3			1							
204	20	1	5	6		3			1							
205	19	4	6	7		2			3							
206	18	1	6			3										
207	18	1	6			4										
208	5	1	7	8		1			1							
209	5	2	7	8		2			2							
210	19	4	7	9		4			2							

COLUMN

CARD	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80
151	20.37		.0													
152	-41.63		-4.													
153	40.37		-4.0													
154	.63		12.75													
155	-20.37		.0													
156	-40.37		-4.0													
157	41.63		-4.0													
158	-8.35	31.6			0.0		352.0		0.0		0.0					
159	-25.4	31.6			0.0		352.0		0.0		0.0					
160	-40.0	31.6			0.0		352.0		0.0		0.0					
161	-42.5	58.4			0.0		352.0		0.0		0.0					
162	-109.88	80.0			0.0		352.0		0.0		0.0					
163	-110.51	18.0			0.0		352.0		0.0		0.0					
164	-415.88	66.8			0.0		352.0		0.0		0.0					
165	-775.063	64.556			0.0		352.0		0.0		0.0					
166	-721.25	18.0			0.0		352.0		0.0		0.0					
167	-888.437	64.556			0.0		352.0		0.0		0.0					
168	-942.26	18.0			0.0		352.0		0.0		0.0					
169	-1247.63	66.8			0.0		352.0		0.0		0.0					
170	-1606.813	64.556			0.0		352.0		0.0		0.0					
171	-1553.0	18.0			0.0		352.0		0.0		0.0					
172	-2079.38	46.04			0.0		352.0		0.0		0.0					
173	-2911.13	46.04			0.0		352.0		0.0		0.0					
174	8.35	31.6			0.0		0.0		0.0		0.0					
175	25.4	31.6			0.0		0.0		0.0		0.0					
176	40.0	31.6			0.0		0.0		0.0		0.0					
177	44.25	59.67			0.0		0.0		0.0		0.0					
178	109.88	77.48			0.0		0.0		0.0		0.0					
179	110.51	18.0			0.0		0.0		0.0		0.0					
180	415.88	69.71			0.0		0.0		0.0		0.0					

COLUMN

CARD	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80
211	1	1	8	9		3			1							
212	19	2	8	9		3			1							
213	19	3	8	9		3			1							
214	20	1	8	9		3			1							
215	13	2	10	8		1			4							
216	15	1	10	8		2			5							
217	18	1	9			3										
218	18	1	9			4										
219	5	1	10	12		3			1							
220	5	2	10	12		4			2							
221	1	1	10	11		5			1							
222	19	2	10	11		5			1							
223	19	3	10	11		5			1							
224	20	1	10	11		5			1							
225	19	4	11	12		2			3							
226	18	1	11			3										
227	18	1	11			4										
228	5	1	12	13		1			1							
229	5	2	12	13		2			2							
230	19	4	12	14		4			2							
231	1	1	13	14		3			1							
232	19	2	13	14		3			1							
233	19	3	13	14		3			1							
234	20	1	13	14		3			1							
235	13	2	15	13		1			4							
236	15	1	15	13		3			5							
237	18	1	14			3										
238	18	1	14			4										
239	13	2	16	15		1			2							
240	15	1	16	15		2			4							

CARD	COLUMN 5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80
241	18	2	15			5										
242	18	2	15			6										
243	18	2	16			3										
244	18	2	16			4										
245	6	1	17	18		2			1							
246	21	1	17	18		3			2							
247	21	2	17	20		4			3							
248	13	1	18	19		3			1							
249	7	1	18	19		3			1							
250	19	1	19	20		2			2							
251	22	1	19	20		1			1							
252	5	1	20	21		5			1							
253	5	2	20	21		6			2							
254	5	3	20	21		7			2							
255	5	1	21	23		1			1							
256	5	2	21	23		2			2							
257	1	1	21	22		3			1							
258	19	2	21	22		3			1							
259	19	3	21	22		3			1							
260	20	1	21	22		3			1							
261	19	4	22	23		2			3							
262	18	1	22			3										
263	18	1	22			4										
264	5	1	23	24		1			1							
265	5	2	23	24		2			2							
266	19	4	23	25		4			2							
267	1	1	24	25		3			1							
268	19	2	24	25		3			1							
269	19	3	24	25		3			1							
270	20	1	24	25		3			1							

COLUMN

	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80
CARD																
271	13	2	24	26		4			1							
272	15	1	24	26		5			2							
273	18	1	25			3										
274	18	1	25		4											
275	1	1	26	27		3			1							
276	19	2	26	27		3			1							
277	19	3	26	27		3			1							
278	20	1	26	27		3			1							
279	1	1	26	28		4			1							
280	19	2	26	28		4			1							
281	19	3	26	28		4			1							
282	20	1	26	28		4			1							
283	19	4	26	27		5			2							
284	19	4	26	28		6			2							
285	18	1	27			3										
286	18	1	27			4										
287	18	1	28			3										
288	18	1	28			4										
289	1	1	3110.		21.04		0.									
290	5	1	1.E07	20000.	60000.		100000.	1.		19.55		.6758	1.		2500.	
291	200.		20.													
292	5	2	1.E07	20000.	60000.		100000.	1.		9.79		2.082	1.		5500.	
293	200.		30.													
294	5	3	3.E07	180000.	100000.		200000.	1.		10.36		.776	1.		7800.	
295	200.		40.													
296	6	1	0.3	0.												
297	7	1	10.	1.	.01											
298	13	1	24000.	1.25	320000.		10000.	150000.		1.375		250000.	30.			
299	13	2	12000.	2.5	160000.		10000.	150000.		100.		250000.	30.			
300	14	1	1.	360000.	1.		12.	0.		1.		360000.	1.		123	

TABLE 7 (Concluded)

```
                                    COLUMN

          5    10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
          .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .
CARD  ................................................................................

301     0.   0.2      0.              .03      .03
302     15   1175000. 1633.   0.2     5.0     4450000.20620.  0.053    56.       100.
303       25. 0.001   1.      6.      175000. 1633.   0.2     5.0     4450000.20620.
304     0.053 56.        100.    25.  0.001   1.      6.       1.50   58.125   58.125
305     18   13234000.3234000.    5.  14.00475 1000.     0.      0.       0.
306     18   26468000.6468000. 5.     14.00475 1000.   0.      0.      0.
307     19   13000000.3000000.   000.   000.  2.       2.     5.       10.
308     19   2  0.       3.E07  0.       3.E07  3.75  2.      15.25
309     19   3 0.     26890.   0.     0.      2.79     1.     15.25    1180.
310     19   4 500000.4500000.500000. 4500000.  0.625 0.625   33.      8100.
311     20   1 1180.   173.    1180.   173.    4.5     4.5
312     21   1 5000.     3.E07 1.25   1250.   400000. 1.25
313     21   2 0.     175000. 4.      0.      700000. 10.
314     22   1   3.E07 180000. 100000. 200000. 3.273  2.830   5.475   1.827   2.
315    4820.   150.    20.
316

          .    .    .    .    .    .    .    .    .    .    .    .    .    .    .    .
      ................................................................................
```

TABLE 8.—SAMPLE PROBLEM COMPUTER OUTPUT

```
IITRAIN SAMPLE PROBLEM

IM          1
ID     .0000500000
TF     .3999999985
I,IS(I)        1,     1
               2,     0
               3,     0
               4,     0
               5,     0

NM--NO. MOTIONS OUT    24
NF--NO. FORCES OUT     16
IWW-STEPS PER PRINT   100


GA       .000
GG    386.088


MASS DATA
```

I	WT	RI	II	IC	IF	FA
1	75.000	60.000	1	4	0 0 0 0	.000
2	90.000	70.000	1	3	0 0 0 0	.000
3	150.000	100.000	1	2	0 0 0 0	.000
4	5595.000	3000.000	1	7	0 0 0 0	.000
5	2230.000	2000.000	1	3	0 0 0 0	.000
6	12700.000	44200.000	1	4	0 0 0 0	.000
7	20350.000	1158100.000	1	4	0 0 0 0	.000
8	7825.000	29750.000	1	5	0 0 0 0	.000
9	12700.000	44200.000	1	4	0 0 0 0	.000
10	7825.000	29750.000	1	5	0 0 0 0	.000
11	12700.000	44200.000	1	4	0 0 0 0	.000
12	20350.000	1158100.000	1	4	0 0 0 0	.000
13	7825.000	29750.000	1	5	0 0 0 0	.000
14	12700.000	44200.000	1	4	0 0 0 0	.000
15	61400.000	7310826.000	1	6	0 0 0 0	.000
16	61400.000	7310826.000	1	4	0 0 0 0	.000
17	75.000	60.000	1	4	0 0 0 0	.000
18	90.000	70.000	1	3	0 0 0 0	.000
19	150.000	100.000	1	2	0 0 0 0	.000
20	6180.000	4000.000	1	7	0 0 0 0	.000
21	3230.000	3000.000	1	3	0 0 0 0	.000
22	12700.000	44200.000	1	4	0 0 0 0	.000
23	47293.000	253200.000	1	4	0 0 0 0	.000
24	9408.000	35000.000	1	5	0 0 0 0	.000
25	12700.000	44200.000	1	4	0 0 0 0	.000
26	66739.000	8856844.000	1	6	0 0 0 0	.000
27	12700.000	44200.000	1	4	0 0 0 0	.000
28	12700.000	44200.000	1	4	0 0 0 0	.000

CONTACT POINTS

I	K	XC	YC	AC
1	1	7.350	.000	.000
1	2	-8.700	.000	.000
1	3	-8.150	-6.300	.000
1	4	8.000	5.800	.000

I	K	XC	YC	AC
2	1	8.350	.000	.000
2	2	7.650	-6.300	.000
2	3	.000	.000	.000

I	K	XC	YC	AC
3	1	.000	.000	.000
3	2	18.000	5.300	.000

I	K	XC	YC	AC
4	1	-17.500	-26.800	.000
4	2	20.500	-18.500	.000
4	3	40.600	-10.400	.000
4	4	-17.500	-11.900	.000
4	5	.000	86.600	.000
4	6	.000	-14.800	.000
4	7	-17.500	-14.800	.000

I	K	XC	YC	AC
5	1	.000	65.000	.000
5	2	.000	-36.400	.000
5	3	.000	-34.000	.000

I	K	XC	YC	AC
6	1	.630	12.750	.000
6	2	-20.370	.000	.000
6	3	41.630	-4.000	.000
6	4	-40.370	-4.000	.000

I	K	XC	YC	AC
7	1	.000	78.200	.000
7	2	.000	-23.200	.000
7	3	252.000	-48.800	.000
7	4	-252.000	-48.800	.000

I	K	XC	YC	AC
8	1	.000	80.444	.000
8	2	.000	-20.956	.000
8	3	53.183	-18.556	.000
8	4	3.303	-32.956	.000
8	5	3.303	-19.556	180.000

I	K	XC	YC	AC
9	1	-.630	12.750	.000
9	2	20.370	.000	.000
9	3	40.370	-4.000	.000
9	4	-41.630	-4.000	.000

I	K	XC	YC	AC
10	1	-3.303	-32.956	.000
10	2	-3.303	-19.556	.000
10	3	.000	80.444	.000
10	4	.000	-20.956	.000
10	5	-53.183	-18.556	.000

I	K	XC	YC	AC
11	1	.630	12.750	.000
11	2	-20.370	.000	.000
11	3	41.630	-4.000	.000
11	4	-40.370	-4.000	.000

I	K	XC	YC	AC
12	1	.000	78.200	.000
12	2	.000	-23.200	.000
12	3	252.000	-48.800	.000
12	4	-252.000	-48.800	.000

I	K	XC	YC	AC
13	1	.000	80.444	.000
13	2	.000	-20.956	.000
13	3	53.183	-18.556	.000
13	4	3.303	-32.956	.000
13	5	3.303	-19.556	180.000

I	K	XC	YC	AC
14	1	-.630	12.750	.000
14	2	20.370	.000	.000
14	3	40.370	-4.000	.000
14	4	-41.630	-4.000	.000

I	K	XC	YC	AC
15	1	355.900	-14.440	.000
15	2	-355.900	-14.440	.000
15	3	355.900	-1.040	.000
15	4	-355.900	-1.040	180.000
15	5	306.000	-32.040	.000
15	6	-306.000	-32.040	.000

I	K	XC	YC	AC
16	1	355.900	-14.440	.000
16	2	355.900	-1.040	.000

16	3	306.000	-32.040	.000
16	4	-306.000	-32.040	.000

I	K	XC	YC	AC
17	1	-7.350	.000	180.000
17	2	8.700	.000	.000
17	3	8.150	-6.300	.000
17	4	-8.000	5.800	.000

I	K	XC	YC	AC
18	1	-8.350	.000	.000
18	2	-7.650	-6.300	.000
18	3	.000	.000	.000

I	K	XC	YC	AC
19	1	.000	.000	.000
19	2	-18.000	5.300	.000

I	K	XC	YC	AC
20	1	15.750	-28.070	.000
20	2	-22.250	-19.770	.000
20	3	-42.350	-11.670	.000
20	4	15.750	-14.670	180.000
20	5	.000	85.330	.000
20	6	.000	-16.070	.000
20	7	15.750	-16.070	.000

I	K	XC	YC	AC
21	1	.000	67.520	.000
21	2	.000	-33.880	.000
21	3	.000	-31.480	.000

I	K	XC	YC	AC
22	1	-.630	12.750	.000
22	2	20.370	.000	.000
22	3	-41.630	-4.000	.000
22	4	40.370	-4.000	.000

I	K	XC	YC	AC
23	1	.000	75.290	.000
23	2	.000	-26.110	.000
23	3	-252.000	-51.710	.000
23	4	252.000	-51.710	.000

I	K	XC	YC	AC
24	1	.000	79.205	.000
24	2	.000	-22.195	.000
24	3	-51.620	-19.795	.000
24	4	-1.740	-33.195	.000

24	5	-1.740	-19.295	.000

I	K	XC	YC	AC
25	1	.630	12.750	.000
25	2	-20.370	.000	.000
25	3	-40.370	-4.000	.000
25	4	41.630	-4.000	.000

I	K	XC	YC	AC
26	1	-355.900	-37.000	.000
26	2	-355.900	-23.600	180.000
26	3	-306.000	-22.600	.000
26	4	306.000	-22.600	.000
26	5	-252.000	-50.600	.000
26	6	252.000	-50.600	.000

I	K	XC	YC	AC
27	1	-.630	12.750	.000
27	2	20.370	.000	.000
27	3	-41.630	-4.000	.000
27	4	40.370	-4.000	.000

I	K	XC	YC	AC
28	1	.630	12.750	.000
28	2	-20.370	.000	.000
28	3	-40.370	-4.000	.000
28	4	41.630	-4.000	.000

MASS GLOBAL POSITION AND VELOCITY

I	1
XP	-8.350
YP	31.600
AP	.000
XV	352.000
YV	.000
AV	.000

I	2
XP	-25.400
YP	31.600
AP	.000
XV	352.000
YV	.000
AV	.000

I	3
XP	-40.000
YP	31.600
AP	.000
XV	352.000
YV	.000

AV	.000
I	4
XP	-42.500
YP	58.400
AP	.000
XV	352.000
YV	.000
AV	.000
I	5
XP	-109.880
YP	80.000
AP	.000
XV	352.000
YV	.000
AV	.000
I	6
XP	-110.510
YP	18.000
AP	.000
XV	352.000
YV	.000
AV	.000
I	7
XP	-415.880
YP	66.800
AP	.000
XV	352.000
YV	.000
AV	.000
I	8
XP	-775.063
YP	64.556
AP	.000
XV	352.000
YV	.000
AV	.000
I	9
XP	-721.250
YP	18.000
AP	.000
XV	352.000
YV	.000
AV	.000
I	10
XP	-848.437
YP	64.556
AP	.000
XV	352.000
YV	.000
AV	.000

I	11
XP	-942.260
YP	18.000
AP	.000
XV	352.000
YV	.000
AV	.000

I	12
XP	-1247.630
YP	66.800
AP	.000
XV	352.000
YV	.000
AV	.000

I	13
XP	-1606.813
YP	64.556
AP	.000
XV	352.000
YV	.000
AV	.000

I	14
XP	-1553.000
YP	18.000
AP	.000
XV	352.000
YV	.000
AV	.000

I	15
XP	-2079.380
YP	46.040
AP	.000
XV	352.000
YV	.000
AV	.000

I	16
XP	-2911.130
YP	46.040
AP	.000
XV	352.000
YV	.000
AV	.000

I	17
XP	8.350
YP	31.600
AP	.000
XV	.000
YV	.000
AV	.000

I	18
XP	25.400
YP	31.600
AP	.000
XV	.000
YV	.000
AV	.000
I	19
XP	40.000
YP	31.600
AP	.000
XV	.000
YV	.000
AV	.000
I	20
XP	44.250
YP	59.670
AP	.000
XV	.000
YV	.000
AV	.000
I	21
XP	109.880
YP	77.480
AP	.000
XV	.000
YV	.000
AV	.000
I	22
XP	110.510
YP	18.000
AP	.000
XV	.000
YV	.000
AV	.000
I	23
XP	415.880
YP	69.710
AP	.000
XV	.000
YV	.000
AV	.000
I	24
XP	773.500
YP	65.790
AP	.000
XV	.000
YV	.000
AV	.000
I	25

XP	721.250
YP	18.000
AP	.000
XV	.000
YV	.000
AV	.000

I	26
XP	1247.630
YP	68.600
AP	.000
XV	.000
YV	.000
AV	.000

I	27
XP	942.260
YP	18.000
AP	.000
XV	.000
YV	.000
AV	.000

I	28
XP	1553.000
YP	18.000
AP	.000
XV	.000
YV	.000
AV	.000

ELEMENT CONNECTIONS

I	II	ID	IA			IP								
1	14	1	1	17	0	1	0	0	1	0	0	0	0	0
2	6	1	1	2	0	2	0	0	1	0	0	0	0	0
3	21	1	1	2	0	3	0	0	2	0	0	0	0	0
4	21	2	1	4	0	4	0	0	3	0	0	0	0	0
5	13	1	2	3	0	3	0	0	1	0	0	0	0	0
6	7	1	2	3	0	3	0	0	1	0	0	0	0	0
7	19	1	3	4	0	2	0	0	2	0	0	0	0	0
8	22	1	3	4	0	1	0	0	1	0	0	0	0	0
9	15	1	4	20	0	4	0	0	4	0	0	0	0	0
10	5	1	4	5	0	5	0	0	1	0	0	0	0	0
11	5	2	4	5	0	6	0	0	2	0	0	0	0	0
12	5	3	4	5	0	7	0	0	2	0	0	0	0	0
13	5	1	5	7	0	1	0	0	1	0	0	0	0	0
14	5	2	5	7	0	2	0	0	2	0	0	0	0	0
15	1	1	5	6	0	3	0	0	1	0	0	0	0	0
16	19	2	5	6	0	3	0	0	1	0	0	0	0	0
17	19	3	5	6	0	3	0	0	1	0	0	0	0	0
18	20	1	5	6	0	3	0	0	1	0	0	0	0	0
19	19	4	6	7	0	2	0	0	3	0	0	0	0	0
20	18	1	6	0	0	3	0	0	0	0	0	0	0	0
21	18	1	6	0	0	4	0	0	0	0	0	0	0	0

22	5	1	7	8	0	1	0	0	1	0	0	0	0	0
23	5	2	7	8	0	2	0	0	2	0	0	0	0	0
24	19	4	7	9	0	4	0	0	2	0	0	0	0	0
25	1	1	8	9	0	3	0	0	1	0	0	0	0	0
26	19	2	8	9	0	3	0	0	1	0	0	0	0	0
27	19	3	8	9	0	3	0	0	1	0	0	0	0	0
28	20	1	8	9	0	3	0	0	1	0	0	0	0	0
29	13	2	10	8	0	1	0	0	4	0	0	0	0	0
30	15	1	10	8	0	2	0	0	5	0	0	0	0	0
31	16	1	9	0	0	3	0	0	0	0	0	0	0	0
32	18	1	9	0	0	4	0	0	0	0	0	0	0	0
33	5	1	10	12	0	3	0	0	1	0	0	0	0	0
34	5	2	10	12	0	4	0	0	2	0	0	0	0	0
35	1	1	10	11	0	5	0	0	1	0	0	0	0	0
36	19	2	10	11	0	5	0	0	1	0	0	0	0	0
37	19	3	10	11	0	5	0	0	1	0	0	0	0	0
38	20	1	10	11	0	5	0	0	1	0	0	0	0	0
39	19	4	11	12	0	2	0	0	3	0	0	0	0	0
40	18	1	11	0	0	3	0	0	0	0	0	0	0	0
41	16	1	11	0	0	4	0	0	0	0	0	0	0	0
42	5	1	12	13	0	1	0	0	1	0	0	0	0	0
43	5	2	12	13	0	2	0	0	2	0	0	0	0	0
44	19	4	12	14	0	4	0	0	2	0	0	0	0	0
45	1	1	13	14	0	3	0	0	1	0	0	0	0	0
46	19	2	13	14	0	3	0	0	1	0	0	0	0	0
47	19	3	13	14	0	3	0	0	1	0	0	0	0	0
48	20	1	13	14	0	3	0	0	1	0	0	0	0	0
49	13	2	15	13	0	1	0	0	4	0	0	0	0	0
50	15	1	15	13	0	3	0	0	5	0	0	0	0	0
51	16	1	14	0	0	3	0	0	0	0	0	0	0	0
52	18	1	14	0	0	4	0	0	0	0	0	0	0	0
53	13	2	16	15	0	1	0	0	2	0	0	0	0	0
54	15	1	16	15	0	2	0	0	4	0	0	0	0	0
55	16	2	15	0	0	5	0	0	0	0	0	0	0	0
56	16	2	15	0	0	6	0	0	0	0	0	0	0	0
57	16	2	16	0	0	3	0	0	0	0	0	0	0	0
58	18	2	16	0	0	4	0	0	0	0	0	0	0	0
59	6	1	17	18	0	2	0	0	1	0	0	0	0	0
60	21	1	17	18	0	3	0	0	2	0	0	0	0	0
61	21	2	17	20	0	4	0	0	3	0	0	0	0	0
62	13	1	18	19	0	3	0	0	1	0	0	0	0	0
63	7	1	18	19	0	3	0	0	1	0	0	0	0	0
64	19	1	19	20	0	2	0	0	2	0	0	0	0	0
65	22	1	19	20	0	1	0	0	1	0	0	0	0	0
66	5	1	20	21	0	5	0	0	1	0	0	0	0	0
67	5	2	20	21	0	6	0	0	2	0	0	0	0	0
68	5	3	20	21	0	7	0	0	2	0	0	0	0	0
69	5	1	21	23	0	1	0	0	1	0	0	0	0	0
70	5	2	21	23	0	2	0	0	2	0	0	0	0	0
71	1	1	21	22	0	3	0	0	1	0	0	0	0	0
72	19	2	21	22	0	3	0	0	1	0	0	0	0	0
73	19	3	21	22	0	3	0	0	1	0	0	0	0	0
74	20	1	21	22	0	3	0	0	1	0	0	0	0	0
75	19	4	22	23	0	2	0	0	3	0	0	0	0	0
76	18	1	22	0	0	3	0	0	0	0	0	0	0	0
77	18	1	22	0	0	4	0	0	0	0	0	0	0	0
78	5	1	23	24	0	1	0	0	1	0	0	0	0	0

79	5	2	23	24	0	2	0	0	2	0	0	0	0	0
80	19	4	23	25	0	4	0	0	2	0	0	0	0	0
81	1	1	24	25	0	3	0	0	1	0	0	0	0	0
82	19	2	24	25	0	3	0	0	1	0	0	0	0	0
83	19	3	24	25	0	3	0	0	1	0	0	0	0	0
84	20	1	24	25	0	3	0	0	1	0	0	0	0	0
85	13	2	24	26	0	4	0	0	1	0	0	0	0	0
86	15	1	24	26	0	5	0	0	2	0	0	0	0	0
87	18	1	25	0	0	3	0	0	0	0	0	0	0	0
88	18	1	25	0	4	0	0	0	0	0	0	0	0	0
89	1	1	26	27	0	3	0	0	1	0	0	0	0	0
90	19	2	26	27	0	3	0	0	1	0	0	0	0	0
91	19	3	26	27	0	3	0	0	1	0	0	0	0	0
92	20	1	26	27	0	3	0	0	1	0	0	0	0	0
93	1	1	26	28	0	4	0	0	1	0	0	0	0	0
94	19	2	26	28	0	4	0	0	1	0	0	0	0	0
95	19	3	26	28	0	4	0	0	1	0	0	0	0	0
96	20	1	26	28	0	4	0	0	1	0	0	0	0	0
97	19	4	26	27	0	5	0	0	2	0	0	0	0	0
98	19	4	26	28	0	6	0	0	2	0	0	0	0	0
99	18	1	27	0	0	3	0	0	0	0	0	0	0	0
100	18	1	27	0	0	4	0	0	0	0	0	0	0	0
101	18	1	28	0	0	3	0	0	0	0	0	0	0	0
102	18	1	28	0	0	4	0	0	0	0	0	0	0	0

ELEMENT DISCRIPTIONS

```
ITP    1
IDP    1
 PP   3110.600      21.040        .000        .000        .000        .000        .000        .000        .000

ITP    5
IDP    1
 PP10000000.000  20000.000  60000.000  100000.000       1.000      19.550        .676       1.000    2500.000
        200.000     20.000

ITP    5
IDP    2
 PP10000000.000  20000.000  60000.000  100000.000       1.000       9.790       2.082       1.000    5500.000
        200.000     30.000

ITP    5
IDP    3
 PP30000000.000 100000.000 100000.000  200000.000       1.000      10.360        .776       1.000    7800.000
        200.000     40.000

ITP    6
IDP    1
 PP       .300        .000        .000        .000        .000        .000        .000        .000        .000

ITP    7
IDP    1
 PP     10.000       1.000        .010        .000        .000        .000        .000        .000        .000

ITP   13
IDP    1
```

```
 PP    24000.000        1.250   320000.000    10000.000   150000.000        1.375   250000.000       30.000         .000

ITP    13
IUP     2
 PP    12000.000        2.500   160000.000    10000.000   150000.000      100.000   250000.000       30.000         .000

ITP    14
IUP     1
 PP        1.000   360000.000        1.000       12.000         .000        1.000   360000.000        1.000123000000.000
           .000         .250         .000         .000         .030         .030         .000         .000         .000          .000

ITP    15
IUP     1
 PP   175000.000     1633.000         .200        5.000  4450000.000    20620.000         .053       56.000      100.000
          25.000         .001        1.000        6.000   175000.000     1633.000         .200        5.000 4450000.000    20620.000
           .053       56.000      100.000       25.000         .001        1.000        6.000        1.500       58.125       58.125

ITP    18
IUP     1
 PP 3234000.000  3234000.000        5.000       14.005     1000.000         .000         .000         .000         .000

ITP    18
IUP     2
 PP 6468000.000  6468000.000        5.000       14.005     1000.000         .000         .000         .000         .000

ITP    19
IUP     1
 PP 3000000.000  3000000.000         .000         .000        2.000        2.000        3.000       10.000         .000

ITP    19
IUP     2
 PP         .000300000000.000         .000300000000.000        3.750        2.000       15.250         .000         .000

ITP    19
IUP     3
 PP         .000    26890.000         .000         .000        2.790        1.000       15.250     1180.000         .000

ITP    19
IUP     4
 PP   500000.000  4500000.000   500000.000  4500000.000         .625         .625       33.000     8100.000         .000

ITP    20
IUP     1
 PP     1180.000      173.000     1180.000      173.000        4.500        4.500         .000         .000         .000

ITP    21
IUP     1
 PP     5000.000300000000.000        1.250     1250.000   400000.000        1.250         .000         .000         .000

ITP    21
IUP     2
 PP         .000   175000.000        4.000         .000   700000.000       10.000         .000         .000         .000

ITP    22
IUP     1
 PP300000000.000   180000.000   100000.000   200000.000        3.273        2.830        5.475        1.827        2.000
         4820.000      150.000       20.000         .000         .000         .000         .000         .000         .000          .000
```

TIME

X VEL OF PT 0 ON M 7	X VEL OF PT 0 ON M 12	X VEL OF PT 0 ON M 15	X VEL OF PT 0 ON M 16	X VEL OF PT 0 ON M 23	X VEL OF PT 0 ON M 26	X DISP OF PT 0 ON M 7	X DISP OF PT 0 ON M 23
X ACCL OF PT 0 ON M 7	X ACCL OF PT 0 ON M 12	X ACCL OF PT 0 ON M 15	X ACCL OF PT 0 ON M 16	X ACCL OF PT 0 ON M 23	X ACCL OF PT 0 ON M 26	Y DISP OF PT 0 ON M 7	Y DISP OF PT 0 ON M 23
Y ACCL OF PT 0 ON M 7	Y ACCL OF PT 0 ON M 12	Y ACCL OF PT 0 ON M 15	Y ACCL OF PT 0 ON M 16	Y ACCL OF PT 0 ON M 23	Y ACCL OF PT 0 ON M 26	ANG ACCL OF PT 0 ON M 7	ANG ACCL OF PT 0 ON M 23
FX ON 1 AT PT 1 BY M 1	FX ON 29 AT PT 2 BY M 10	FX ON 49 AT PT 1 BY M 15	FX ON 53 AT PT 1 BY M 16	FX ON 85 AT PT 4 BY M 24	FY ON 9 AT PT 4 BY M 4	FY ON 30 AT PT 1 BY M 10	FY ON 86 AT PT 5 BY M 24
FX ON 9 AT PT 4 BY M 4	FX ON 30 AT PT 1 BY M 10	FX ON 50 AT PT 3 BY M 15	FX ON 54 AT PT 2 BY M 16	FX ON 86 AT PT 5 BY M 24	MZ ON 9 AT PT 4 BY M 4	MZ ON 30 AT PT 1 BY M 10	MZ ON 86 AT PT 5 BY M 24

.00000

.35200+03	.35200+03	.35200+03	.35200+03	.00000	.00000	-.41588+03	.41588+03
-.54280-01	.72374-01	.00000	.00000	.23358-01	-.21964-01	.66800+02	.69710+02
-.38609+03	-.38609+03	.29034+00	.29034+00	-.38603+03	-.17775+03	-.12056-03	.59914-02
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000

.50000-02

.35185+03	.35200+03	.35200+03	.35200+03	.10699+00	-.53499-04	-.41412+03	.41588+03
-.70199+02	.40498+00	.00000	.00000	.38830+02	-.38852-02	.66796+02	.69705+02
-.32151+03	-.33771+03	.27264+00	.27264+00	-.36161+03	-.16508+03	-.69176-02	.69055+00
.59432+02	.00000	.00000	.00000	.00000	.00000	.00000	.00000
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000

.10000-01

.35103+03	.35200+03	.35200+03	.35200+03	.48987+00	-.62983-04	-.41236+03	.41588+03
-.37807+03	.99821-02	.00000	.00000	.15560+03	-.76989-03	.66783+02	.69692+02
-.29276+03	-.30183+03	.25602+00	.25602+00	-.35077+03	-.15232+03	-.17457+01	.80017+01
.75613+05	.00000	.00000	.00000	.00000	.00000	.00000	.00000
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000

.15000-01							
.34298+03	.35201+03	.35200+03	.35200+03	.45104+01	-.64646-04	-.41062+03	.41589+03
-.22366+04	.21718+01	.00000	.00000	.14107+04	-.97539-04	.66763+02	.69669+02
-.20468+03	-.27277+03	.24041+00	.24041+00	-.31125+03	-.14005+03	.65637+00	-.53775+01
.16578+03	.00000	.00000	.00000	.00000	.32156+05	.00000	.00000
.24667+06	.00000	.00000	.00000	.00000	.19005+07	.00000	.00000
.20000-01							
.33362+03	.35200+03	.35200+03	.35200+03	.11319+02	.43724-02	-.40893+03	.41593+03
-.17037+04	-.41898+01	.00000	.00000	.12920+04	.19643+01	.66738+02	.69639+02
-.24266+03	-.24698+03	.22575+00	.22575+00	-.28447+03	-.12947+03	.48878+00	-.38968+01
.15054+03	.45480+03	.00000	.00000	.39189+03	.21422+05	.00000	.00000
.26150+06	.00000	.00000	.00000	.00000	.12932+07	.00000	.00000
.25000-01							
.32514+03	.35196+03	.35200+03	.35200+03	.17549+02	.22251-01	-.40729+03	.41600+03
-.17289+04	-.10679+02	.00000	.00000	.12228+04	.55256+01	.66707+02	.69601+02
-.27694+03	-.22605+03	.21199+00	.21199+00	-.27720+03	-.11898+03	-.15569+00	.11443-01
.13615+03	.16771+04	.00000	.00000	.11606+04	.10274+05	.00000	.00000
.27632+06	.00000	.00000	.00000	.00000	.66116+06	.00000	.00000
.30000-01							
.31603+03	.35188+03	.35200+03	.35200+03	.23785+02	.62392-01	-.40568+03	.41611+03
-.19325+04	-.23321+02	.00000	.00000	.12822+04	.10880+02	.66669+02	.69557+02
-.25166+03	-.20845+03	.19907+00	.19907+00	-.27433+03	-.10927+03	-.49135+00	.26541+01
.14018+06	.34741+04	.00000	.00000	.23540+04	.23667+04	.00000	.00000
.29072+06	.00000	.00000	.00000	.00000	.21602+06	.00000	.00000
.35000-01							
.30477+03	.35178+03	.35200+03	.35200+03	.30723+02	.13386+00	-.40413+03	.41624+03
-.28493+04	-.29637+02	.00000	.00000	.16246+04	.18129+02	.66625+02	.69506+02
-.19511+03	-.19226+03	.18693+00	.18693+00	-.26387+03	-.51779+02	-.23099+01	.13263+02
.15535+06	.58313+04	.00000	.00000	.39799+04	-.13586+04	.00000	.00000
.30377+06	.00000	.00000	.00000	.00000	.22454+05	.00000	.00000
.40000-01							
.28678+03	.35155+03	.35200+03	.35200+03	.40444+02	.24719+00	-.40265+03	.41642+03
-.32773+04	-.57038+02	.00000	.00000	.21952+04	.28114+02	.66576+02	.69448+02
-.12218+03	-.17792+03	.17554+00	.17554+00	-.23384+03	-.41249+02	.38433+01	.10593+02
.20691+03	.88984+04	.00000	.00000	.61371+04	.33232+03	.00000	.00000
.31447+06	.00000	.00000	.00000	.00000	.14594+06	.00000	.00000
.45000-01							
.27817+03	.35123+03	.35200+03	.35200+03	.50617+02	.41276+00	-.40123+03	.41665+03
-.16754+04	-.79042+02	.00000	.00000	.15493+04	.37735+02	.66523+02	.69384+02
-.17376+03	-.16482+03	-.17387+00	-.17387+00	-.24312+03	-.30431+02	.11187+01	-.52143+01
.19032+03	.12846+05	.00000	.00000	.89596+04	.49839+04	.00000	.00000
.32635+06	.00000	.00000	.00000	.00000	.40965+06	.00000	.00000

.50000-01

.26933+03	.35077+03	.35200+03	.35200+03	.57086+02	.64121+00	-.39986+03	.41692+03
-.18795+04	-.10563+03	.00000	.00000	.12590+04	.53464+02	.66467+02	.69314+02
-.17129+03	-.15225+03	-.16327+00	-.16327+00	-.27281+03	-.31096+02	-.27039+00	.28220+01
.16843+03	.17418+05	.00000	.00000	.12357+05	.81528+04	.00000	.00000
.33853+06	.00000	.00000	.00000	.00000	.59530+06	.00000	.00000

.55000-01

.25952+03	.35016+03	.35200+03	.35200+03	.63490+02	.94472+00	-.39854+03	.41722+03
-.20263+04	-.13709+03	.00000	.00000	.13003+04	.69275+02	.66406+02	.69238+02
-.13631+03	-.14052+03	-.15331+00	-.15331+00	-.26124+03	-.24637+02	.25335-01	.66542+00
.14903+03	.22453+05	.00000	.00000	.16179+05	.90663+04	.00000	.00000
.34955+06	.00000	.00000	.00000	.00000	.65068+06	.00000	.00000

.60000-01

.24920+03	.34939+03	.35200+03	.35200+03	.70244+02	.13314+01	-.39727+03	.41755+03
-.20964+04	-.17169+03	.00000	.00000	.14019+04	.85929+02	.66342+02	.69155+02
-.12131+03	-.12940+03	-.14397+00	-.14397+00	-.23882+03	-.19336+02	.59258+00	-.13815+01
.12743+03	.27971+05	.00000	.00000	.20348+05	.89301+04	.00000	.00000
.35932+06	.00000	.00000	.00000	.00000	.64561+06	.00000	.00000

.65000-01

.23891+03	.34816+03	.35200+03	.35200+03	.77450+02	.18076+01	-.39605+03	.41792+03
-.19072+04	-.43210+03	-.92254+00	-.32095-03	.14764+04	.10520+03	.66274+02	.69066+02
-.10676+03	-.12110+03	-.13614+00	-.13519+00	-.22399+03	-.18071+02	.15751+01	-.32321+01
.10340+03	.81424+05	.14676+03	.51041-01	.24853+05	.85818+04	.00000	.00000
.36796+06	.00000	.00000	.00000	.00000	.62782+06	.00000	.00000

.70000-01

.23144+03	.34474+03	.35199+03	.35200+03	.84930+02	.23813+01	-.39488+03	.41833+03
-.14402+04	-.80668+03	-.25589+01	-.18465-02	.15114+04	.12514+03	.66204+02	.68971+02
-.11416+03	-.11886+03	-.12965+00	-.12695+00	-.21473+03	-.14249+02	.26312+01	-.37607+01
.00000	.00000	.40724+03	.29366+00	.29706+05	.90678+04	.00000	.00000
.37578+06	.00000	.00000	.00000	.00000	.65537+06	.00000	.00000

.75000-01

.22075+03	.34374+03	.35197+03	.35200+03	.92277+02	.37582+01	-.39374+03	.41877+03
-.16136+04	-.71706+03	-.53549+01	-.54504-02	.13821+04	.43316+03	.66131+02	.68871+02
-.77298+02	-.11487+03	-.12503+00	-.11921+00	-.21362+03	-.13768+02	.23578+00	-.67443+01
.00000	.00000	.85246+03	.86678+00	.93325+05	.88022+04	.34896+05	.00000
.38259+06	.23635+06	.00000	.00000	.00000	.64485+06	-.11171+07	.00000

.80000-01

.21615+03	.33650+03	.35194+03	.35200+03	.98656+02	.60716+01	-.39265+03	.41925+03
-.79328+03	-.16427+04	-.91899+01	-.12122-01	.13127+04	-.16774+03	.66056+02	.68766+02
-.40498+02	-.16322+03	-.12218+00	-.11194+00	-.22461+03	-.72512+01	-.70402+00	-.46949+01
.00000	.00000	.14634+04	.19278+01	.00000	.82414+04	.25975+05	.00000
.38849+06	.24106+06	.00000	.00000	.00000	.61656+06	-.16829+07	.00000

.65000-01
.21197+03 .32641+03 .35188+03 .35200+03 .10648+03 .57040+01 -.39158+03 .41976+03
-.67919+03 -.15929+04 -.15634+02 -.23513-01 .16863+04 .33336+03 .65981+02 .68655+02
-.29569+02 -.16638+03 -.12304+00 -.10512+00 -.21071+03 .20779+03 -.95195+00 -.18882+01
.00000 .00000 .24900+04 .37392+01 .00000 .70603+04 .14582+05 .37233+05
.39382+06 .24527+06 .00000 .00000 .62236+05 .55332+06 -.23851+07 .21519+07

.90000-01
.20746+03 .32056+03 .35177+03 .35200+03 .11216+03 .11229+02 -.39054+03 .42031+03
-.92161+03 -.15557+04 -.25398+02 -.42523-01 .80017+03 .10670+04 .65904+02 .68538+02
-.40172+02 -.13474+03 -.12853+00 -.98712-01 -.25343+03 .10033+03 -.51018+00 -.10227+02
.00000 .00000 .40458+04 .67624+01 .00000 .53778+04 .48570+04 .26215+05
.39868+06 .24986+06 .00000 .00000 .23964+06 .46025+06 -.30011+07 .14865+07

.95000-01
.20279+03 .31278+03 .35162+03 .35200+03 .11576+03 .16358+02 -.38951+03 .42088+03
-.94677+03 -.15626+04 -.38361+02 -.72254-01 .67707+03 .10027+04 .65826+02 .68416+02
-.53087+02 -.86618+02 -.13854+00 -.92694-01 -.27384+03 -.91980+01 .12489+00 -.43882+01
.00000 .00000 .61120+04 .11491+02 .00000 .31786+04 -.13151+04 .77637+04
.40311+06 .25480+06 .00000 .00000 .24377+06 .33614+06 -.34189+07 .43115+06

.10000+00
.19804+03 .30489+03 .35139+03 .35200+03 .11908+03 .21353+02 -.38851+03 .42147+03
-.94368+03 -.15898+04 -.54165+02 -.19838+00 .65668+03 .10000+04 .65747+02 .68286+02
-.58205+02 -.50020+02 -.15275+00 -.87041-01 -.26746+03 -.11112+03 .58081+00 -.11861+01
.00000 .00000 .86454+04 .31549+02 .00000 .86411+03 -.37426+04 -.10115+05
.40720+06 .25990+06 .00000 .00000 .24805+06 .20418+06 -.36245+07 -.59459+06

.10500+00
.19332+03 .29692+03 .35107+03 .35200+03 .12233+03 .26392+02 -.38753+03 .42207+03
-.94312+03 -.16009+04 -.72701+02 -.43726+00 .64301+03 .10172+04 .65667+02 .68150+02
-.60209+02 -.27089+02 -.17082+00 -.81736-01 -.24697+03 -.15009+03 .83536+00 .99539+00
.00000 .00000 .11631+05 .69538+02 .00000 -.11019+04 -.39319+04 -.24481+05
.41096+06 .26502+06 .00000 .00000 .25257+06 .91261+05 -.37022+07 -.14209+07

.11000+00
.18861+03 .28888+03 .35066+03 .35200+03 .12550+03 .31535+02 -.38658+03 .42269+03
-.94136+03 -.16151+04 -.93932+02 -.76203+00 .62056+03 .10408+04 .65585+02 .68008+02
-.51552+02 -.12404+02 -.19214+00 -.76753-01 -.22197+03 -.19930+03 .11065+01 .26363+01
.00000 .00000 .15059+05 .12119+03 .00000 -.23398+04 -.29359+04 -.34369+05
.41436+06 .27006+06 .00000 .00000 .25723+06 .19518+05 -.37098+07 -.19907+07

.11500+00
.18393+03 .28076+03 .35013+03 .35199+03 .12854+03 .36799+02 -.38564+03 .42332+03
-.92770+03 -.16308+04 -.11770+03 -.12640+01 .60101+03 .10655+04 .65502+02 .67860+02
-.36955+02 -.56971+01 -.21580+00 -.72073-01 -.19905+03 -.18774+03 .12235+01 .33401+01
.00000 .00000 .18918+05 .20102+03 .00000 -.27039+04 -.15225+04 -.37367+05
.41739+06 .27494+06 .00000 .00000 .26191+06 -.23788+04 -.36910+07 -.21624+07

.12000+00

.17933+03	.27260+03	.34948+03	.35198+03	.13153+03	.42192+02	-.38474+03	.42397+03
-.91197+03	-.16362+04	-.14396+03	-.19802+01	.59435+03	.10918+04	.65418+02	.67707+02
-.25266+02	-.46013+01	-.24077+00	-.67683-01	-.18153+03	-.15617+03	.11510+01	.31230+01
.00000	.00000	.23209+05	.31491+03	.00000	-.22002+04	-.27312+03	-.37419+05
.42004+06	.27965+06	.00000	.00000	.26652+06	.25460+05	-.36795+07	-.21636+07
.12500+00							
.17480+03	.26441+03	.34869+03	.35197+03	.13449+03	.47713+02	-.38385+03	.42464+03
-.89926+03	-.16382+04	-.17261+03	-.29727+01	.58741+03	.11167+04	.65333+02	.67550+02
-.16747+02	-.50840+01	-.26561+00	-.63556-01	-.16840+03	-.12845+03	.10777+01	.27995+01
.00000	.00000	.27924+05	.47275+03	.00000	-.10565+04	.73319+03	-.37159+05
.42231+06	.28419+06	.00000	.00000	.27099+06	.89969+05	-.36799+07	-.21469+07
.13000+00							
.17035+03	.25639+03	.34742+03	.35195+03	.13740+03	.53355+02	-.38299+03	.42532+03
-.87731+03	-.14740+04	-.43455+03	-.43106+01	.58023+03	.11400+04	.65248+02	.67388+02
-.11489+02	-.95880+01	-.60827+00	-.59688-01	-.15813+03	-.10060+03	.95638+00	.23587+01
.00000	.00000	.69792+05	.68552+03	.00000	.55535+03	.16065+04	-.35474+05
.42419+06	.28854+06	.00000	.00000	.27533+06	.18112+06	-.36858+07	-.20475+07
.13500+00							
.16601+03	.25011+03	.34421+03	.35193+03	.14031+03	.59110+02	-.38215+03	.42601+03
-.85837+03	-.10440+04	-.84965+03	-.64775+01	.58193+03	.11623+04	.65162+02	.67223+02
-.10066+02	-.26938+02	-.93064+00	-.56074-01	-.14994+03	-.68693+02	.75420+00	.15022+01
.00000	.00000	.13615+06	.10301+04	.00000	.21141+04	.23703+04	-.32840+05
.42569+06	.29272+06	.00000	.00000	.27950+06	.26924+06	-.36963+07	-.18929+07
.14000+00							
.16177+03	.24262+03	.34321+03	.35189+03	.14322+03	.64974+02	-.38133+03	.42672+03
-.83622+03	-.19364+04	-.96576+01	-.96576+01	.58481+03	.11831+04	.65077+02	.67054+02
-.11297+02	-.16677+02	.15112+01	-.52644-01	-.14408+03	-.40950+02	.55553+00	.64844+00
.00000	.00000	.00000	.15359+04	.00000	.34245+04	.31286+04	-.29888+05
.42682+06	.29681+06	.00000	.00000	.28351+06	.34265+06	-.37064+07	-.17197+07
.14500+00							
.15765+03	.23754+03	.33779+03	.35183+03	.14615+03	.70938+02	-.38053+03	.42744+03
-.81108+03	-.22696+03	-.14886+04	-.13641+02	.58752+03	.12025+04	.64991+02	.66881+02
-.14494+02	.16896+02	-.75427+02	-.49687-01	-.14028+03	-.19618+02	.35485+00	-.22692+00
.00000	.00000	.00000	.21694+04	.00000	.44399+04	.39516+04	-.27129+05
.42759+06	.30075+06	.23890+06	.00000	.28736+06	.39868+06	-.37113+07	-.15573+07
.15000+00							
.15366+03	.23648+03	.33030+03	.35175+03	.14909+03	.76995+02	-.37975+03	.42818+03
-.78536+03	-.23561+03	-.15076+04	-.20333+02	.58788+03	.12202+04	.64904+02	.66704+02
-.17829+02	.41660+02	.18516+03	-.46893-01	-.13815+03	-.30749+01	.17356+00	-.83738+00
.00000	.00000	.00000	.32336+04	.00000	.52005+04	.43517+04	-.24925+05
.42800+06	.30467+06	.24302+06	.00000	.29103+06	.43947+06	-.37397+07	-.14268+07
.15500+00							
.14980+03	.23515+03	.32272+03	.35162+03	.15202+03	.83138+02	-.37900+03	.42894+03

-.75227+03	-.29553+03	-.15235+04	-.29841+02	.58035+03	.12364+04	.64817+02	.66524+02
-.20446+02	.59059+02	.12903+03	-.42831-01	-.13662+03	.64844+01	.65969-01	-.12753+01
.00000	.00000	.00000	.47457+04	.00000	.58076+04	.43359+04	-.23490+05
.42485+06	.30874+06	.24703+06	.00000	.29454+06	.46984+06	-.37933+07	-.13406+07
.16000+00							
.14645+03	.23358+03	.31507+03	.35144+03	.15476+03	.89356+02	-.37826+03	.42970+03
-.55535+03	-.33096+03	-.15362+04	-.42108+02	.49429+03	.12507+04	.64730+02	.66341+02
-.23060+02	.68568+02	-.12369+03	-.41680-01	-.13644+03	.97558+01	.29859-01	-.21895+01
.00000	.00000	.00000	.66965+04	.00000	.62476+04	.40470+04	-.22891+05
.39832+06	.31296+06	.25100+06	.00000	.29787+06	.48140+06	-.38642+07	-.13026+07
.16500+00							
.14421+03	.23165+03	.30736+03	.35120+03	.15689+03	.95641+02	-.37753+03	.43048+03
-.35122+03	-.35524+03	-.15451+04	-.57180+02	.35845+03	.12634+04	.64642+02	.66154+02
-.25305+02	.72619+02	-.12452+03	-.41468-01	-.13800+03	.89059+01	-.37861+00	-.17996+01
.00000	.00000	.00000	.90934+04	.00000	.64016+04	.35204+04	-.23130+05
.37615+06	.31728+06	.25482+06	.00000	.30102+06	.47602+06	-.39492+07	-.13128+07
.17000+00							
.14292+03	.23006+03	.29962+03	.35087+03	.15804+03	.10199+03	-.37681+03	.43127+03
-.15621+03	-.35821+03	-.15498+04	-.75088+02	.17292+03	.12755+04	.64554+02	.65964+02
-.18926+02	.67595+02	.38041+02	-.35198-01	-.15444+03	.51176+01	-.67880+00	-.17300+01
.00000	.00000	.00000	.11941+05	.00000	.64338+04	.28365+04	-.23815+05
.34999+06	.32160+06	.25840+06	.00000	.30406+06	.46210+06	-.40429+07	-.13556+07
.17500+00							
.14265+03	.22824+03	.29187+03	.35044+03	.15917+03	.10839+03	-.37610+03	.43206+03
.37252+02	-.37484+03	-.15500+04	-.95756+02	.24669+03	.12839+04	.64465+02	.65770+02
-.11919+02	.57147+02	.16320+03	-.31502-01	-.13078+03	-.16554+02	-.96754+00	-.57899+01
.00000	.00000	.00000	.15228+05	.00000	.65791+04	.20372+04	-.28170+05
.32677+06	.32587+06	.26173+06	.00000	.30667+06	.45587+06	-.41419+07	-.16003+07
.18000+00							
.14321+03	.22632+03	.28413+03	.34991+03	.16028+03	.11482+03	-.37539+03	.43286+03
.19211+03	-.39076+03	-.15457+04	-.11917+03	.15930+03	.12903+04	.64375+02	.65573+02
-.66819+01	.46144+02	.10090+02	-.34827-01	-.83692+02	-.21549+02	-.10919+01	-.56578+01
.00000	.00000	.00000	.18952+05	.00000	.65765+04	.11480+04	-.30306+05
.30490+06	.33004+06	.26476+06	.00000	.30878+06	.44239+06	-.42442+07	-.17077+07
.18500+00							
.14456+03	.22434+03	.27642+03	.34925+03	.16057+03	.12130+03	-.37467+03	.43366+03
.34377+03	-.40062+03	-.15368+04	-.14527+03	-.31688+02	.12993+04	.64286+02	.65374+02
.12809+01	.34085+02	-.15794+03	-.31219-01	-.52777+02	.64678+01	-.11320+01	-.21815+01
.00000	.00000	.00000	.23103+05	.00000	.60594+04	.23382+03	-.26487+05
.28541+06	.33406+06	.26751+06	.00000	.31065+06	.40053+06	-.43456+07	-.14726+07
.19000+00							
.14658+03	.22230+03	.26877+03	.34845+03	.16012+03	.12782+03	-.37394+03	.43447+03
.45754+03	-.41902+03	-.15236+04	-.17402+03	-.13356+03	.13113+04	.64196+02	.65173+02

.11567+02	.23819+02	-.56310+02	-.27013-01	-.38803+02	.48524+02	-.11732+01	-.11387+01
.00000	.00000	.00000	.27674+05	.00000	.49239+04	-.66139+03	-.19038+05
.27162+06	.33788+06	.26997+06	.00000	.31234+06	.32516+06	-.44431+07	-.10319+07
.19500+00							
.14905+03	.22016+03	.26145+03	.34725+03	.15935+03	.13441+03	-.37320+03	.43526+03
.52364+03	-.43222+03	-.13033+04	-.40802+03	-.16911+03	.13225+04	.64107+02	.64971+02
.20647+02	.13312+02	.13084+03	-.28757-01	-.35940+02	.82827+02	-.11466+01	-.11050+01
.00000	.00000	.00000	.64889+05	.00000	.33648+04	-.15179+04	-.11688+05
.26418+06	.34149+06	.27216+06	.00000	.31372+06	.22774+06	-.45353+07	-.59719+06
.20000+00							
.15174+03	.21796+03	.25596+03	.34413+03	.15848+03	.14104+03	-.37245+03	.43606+03
.54419+03	-.44864+03	-.87728+03	-.84653+03	-.17378+03	.13303+04	.64019+02	.64769+02
.28644+02	.34935+01	.83131+02	-.13135-01	-.41553+02	.10067+03	-.10206+01	-.12580+01
.00000	.00000	.00000	.13462+06	.00000	.16632+04	-.23132+04	-.63821+04
.26300+06	.34485+06	.27414+06	.00000	.31471+06	.12432+06	-.46209+07	-.27995+06
.20500+00							
.15443+03	.21568+03	.24832+03	.34315+03	.15764+03	.14770+03	-.37168+03	.43685+03
.53028+03	-.46618+03	-.17351+04	.00000	-.16165+03	.13330+04	.63931+02	.64565+02
.35724+02	-.25520+00	-.10114+03	-.23261-01	-.49733+02	.99690+02	-.82548+00	-.11296+01
.00000	.00000	.00000	.00000	.00000	.20573+03	-.30305+04	-.38803+04
.26686+06	.34796+06	.27593+06	.00000	.31527+06	.37014+05	-.46987+07	-.12423+06
.21000+00							
.15699+03	.21330+03	.24478+03	.33799+03	.15688+03	.15436+03	-.37090+03	.43763+03
.48909+03	-.48151+03	-.24384+03	-.15011+04	-.13865+03	.13267+04	.63844+02	.64360+02
.39811+02	-.20856+00	.15874+03	-.78567+02	-.57557+02	.11144+03	-.59681+00	-.69190+00
.00000	.00000	.00000	.00000	.00000	-.83060+03	-.36288+04	-.40412+04
.27492+06	.35082+06	.27750+06	.23872+06	.31468+06	-.24108+05	-.47666+07	-.12280+06
.21500+00							
.15929+03	.21088+03	.24360+03	.33041+03	.15644+03	.16057+03	-.37011+03	.43842+03
.42445+03	-.49029+03	-.22489+03	-.15302+04	.30484+01	.11255+04	.63758+02	.64154+02
.40942+02	-.84312+00	.18976+03	.18689+03	-.62951+02	.91072+02	-.34986+00	.20372+00
.00000	.00000	.00000	.00000	.00000	-.15525+04	-.40606+04	-.58216+04
.28631+06	.35344+06	.27912+06	.24336+06	.27496+06	-.65417+05	-.48220+07	-.23151+06
.22000+00							
.16122+03	.20841+03	.24252+03	.32269+03	.15716+03	.16563+03	-.36931+03	.43920+03
.34508+03	-.49855+03	-.20853+03	-.15572+04	.29365+03	.91248+03	.63673+02	.63946+02
.40305+02	.16142+00	-.71435+02	.12095+03	-.64700+02	.64120+02	-.17523-01	.12553+00
.00000	.00000	.00000	.00000	.00000	-.22151+04	-.43072+04	-.78147+04
.29888+06	.35584+06	.28081+06	.24765+06	.22467+06	-.10210+06	-.48640+07	-.36268+06
.22500+00							
.16276+03	.20590+03	.24151+03	.31484+03	.15929+03	.16985+03	-.36850+03	.43999+03
.27673+03	-.50396+03	-.19483+03	-.15821+04	.56190+03	.77645+03	.63589+02	.63736+02
.37534+02	.21446+01	-.30484+02	-.13402+03	-.59550+02	.44177+02	.37656+00	.60533+00

.00000	.00000	.00000	.00000	.00000	-.26434+04	-.43695+04	-.96501+04
.30940+06	.35802+06	.28259+06	.25160+06	.18848+06	-.12552+06	-.48930+07	-.48310+06
.23000+00							
.16411+03	.20337+03	.24057+03	.30688+03	.16278+03	.17335+03	-.36769+03	.44080+03
.27234+03	-.50575+03	-.18388+03	-.16047+04	.82621+03	.61346+03	.63506+02	.63525+02
.36793+02	.24461+01	.15124+03	-.11541+03	-.19136+02	.27502+02	.49882+00	.26277-02
.00000	.00000	.00000	.00000	.00000	-.28425+04	-.42501+04	-.11547+05
.31215+06	.36001+06	.28444+06	.25519+06	.14936+06	-.13615+06	-.49093+07	-.60835+06
.23500+00							
.16558+03	.20083+03	.23967+03	.29880+03	.16729+03	.17602+03	-.36686+03	.44162+03
.32305+03	-.50820+03	-.17574+03	-.16249+04	.94531+03	.47202+03	.63423+02	.63314+02
.39654+02	.27205+01	-.31241+01	.33339+02	-.16399+01	.19520+02	.29825+00	-.19155+01
.00000	.00000	.00000	.00000	.00000	-.32429+04	-.39890+04	-.11896+05
.30812+06	.36182+06	.28637+06	.25841+06	.11356+06	-.15875+06	-.49155+07	-.64197+06
.24000+00							
.16735+03	.19830+03	.23881+03	.29063+03	.17199+03	.17831+03	-.36603+03	.44247+03
.39085+03	-.50399+03	-.17046+03	-.16429+04	.92399+03	.45695+03	.63342+02	.63102+02
.41904+02	.43730+01	-.15318+03	.16316+03	-.13498+02	.28242+02	.15463+00	-.12208+01
.00000	.00000	.00000	.00000	.00000	-.41154+04	-.36412+04	-.98765+04
.30015+06	.36344+06	.28836+06	.26127+06	.10455+06	-.20990+06	-.49146+07	-.52785+06
.24500+00							
.16952+03	.19579+03	.23796+03	.28238+03	.17642+03	.18060+03	-.36519+03	.44334+03
.47966+03	-.49971+03	-.16810+03	-.16584+04	.84584+03	.45410+03	.63262+02	.62890+02
.42391+02	.56371+01	.83738+00	.19128+02	-.25231+02	.37393+02	-.47053-01	-.30873+00
.00000	.00000	.00000	.00000	.00000	-.52512+04	-.32617+04	-.77938+04
.28902+06	.36484+06	.29047+06	.26374+06	.10297+06	-.27793+06	-.49096+07	-.40694+06
.25000+00							
.17213+03	.19331+03	.23712+03	.27405+03	.18050+03	.18277+03	-.36433+03	.44423+03
.57176+03	-.49320+03	-.16867+03	-.16715+04	.79064+03	.41467+03	.63183+02	.62678+02
.46596+02	.64220+01	.36440+02	-.15433+03	-.33862+02	.34616+02	-.24090+00	.34629+00
.00000	.00000	.00000	.00000	.00000	-.63262+04	-.28887+04	-.74870+04
.27732+06	.36601+06	.29265+06	.26582+06	.95357+05	-.34332+06	-.49024+07	-.39116+06
.25500+00							
.17521+03	.19086+03	.23627+03	.26567+03	.18438+03	.18479+03	-.36347+03	.44514+03
.65452+03	-.48266+03	-.17217+03	-.16822+04	.76558+03	.39570+03	.63105+02	.62465+02
.53462+02	.83691+01	-.14957+03	-.63208+02	-.28411+02	.22130+02	-.38942+00	.17745+01
.00000	.00000	.00000	.00000	.00000	-.70802+04	-.25455+04	-.91227+04
.26566+06	.36692+06	.29491+06	.26752+06	.91316+05	-.39052+06	-.48942+07	-.48621+06
.26000+00							
.17869+03	.18848+03	.23539+03	.25724+03	.18811+03	.18675+03	-.36258+03	.44607+03
.73473+03	-.46939+03	-.17861+03	-.16905+04	.72260+03	.38998+03	.63029+02	.62251+02
.60955+02	.11021+02	-.13715+03	.12855+03	-.10549+02	-.21584+02	-.49818+00	.16039+01
.00000	.00000	.00000	.00000	.00000	-.74019+04	-.22412+04	-.12106+05

.25394+06	.36755+06	.29724+06	.26883+06	.90569+05	-.41274+06	-.48852+07	-.65748+06
.26500+00							
.18255+03	.18616+03	.23448+03	.24877+03	.19152+03	.18872+03	-.36168+03	.44702+03
.80746+03	-.45599+03	-.18794+03	-.16962+04	.63001+03	.40242+03	.62954+02	.62036+02
.68171+02	.13167+02	.40746+01	.91627+02	-.12224+02	-.33727+02	-.60260+00	.12650+01
.00000	.00000	.00000	.00000	.00000	-.73787+04	-.19797+04	-.14964+05
.24351+06	.36789+06	.29965+06	.26976+06	.93677+05	-.41463+06	-.48754+07	-.81909+06
.27000+00							
.18672+03	.18394+03	.23351+03	.24028+03	.19438+03	.19085+03	-.36076+03	.44799+03
.84417+03	-.42002+03	-.20013+03	-.16996+04	.52174+03	.45079+03	.62880+02	.61821+02
.73504+02	.15503+02	-.72245+02	-.97325+02	-.23780+02	-.37755+02	-.71477+00	.15965+01
.00000	.00000	.00000	.00000	.00000	-.72025+04	-.17271+04	-.17038+05
.23556+06	.36242+06	.30212+06	.27029+06	.10475+06	-.40698+06	-.47915+07	-.93103+06
.27500+00							
.19057+03	.18233+03	.23247+03	.23178+03	.19682+03	.19317+03	-.35981+03	.44897+03
.67799+03	-.20592+03	-.21849+03	-.16971+04	.46178+03	.47452+03	.62809+02	.61606+02
.78964+02	.21527+02	-.16937+03	-.11379+03	-.18758+02	-.35933+02	-.69867+00	.26092+01
.00000	.00000	.00000	.00000	.00000	-.70100+04	-.13320+04	-.18909+05
.23095+06	.33394+06	.30465+06	.26989+06	.11193+06	-.39753+06	-.44030+07	-.10331+07
.28000+00							
.19363+03	.18175+03	.23079+03	.22385+03	.19904+03	.19556+03	-.35885+03	.44996+03
.55863+03	-.36891+02	-.53291+03	-.13986+04	.42697+03	.47895+03	.62739+02	.61391+02
.83341+02	.28282+02	-.52765+02	.55093+02	-.60168+01	-.30508+02	-.38644+00	.29013+01
.00000	.00000	.00000	.00000	.00000	-.66570+04	-.10436+04	-.20588+05
.22553+06	.31443+06	.30718+06	.22243+06	.11439+06	-.37819+06	-.41390+07	-.11265+07
.26500+00							
.19617+03	.18196+03	.22668+03	.21826+03	.20107+03	.19797+03	-.35788+03	.45096+03
.45260+03	.12961+03	-.11318+04	-.81492+03	.38095+03	.48920+03	.62672+02	.61175+02
.85432+02	.33746+02	.80698+01	.12039+03	-.31108+01	-.23262+02	-.17684+00	.23676+01
.00000	.00000	.00000	.00000	.00000	-.60914+04	-.79378+03	-.21661+05
.21909+06	.29134+06	.30959+06	.12960+06	.11699+06	-.34615+06	-.38314+07	-.11848+07
.28999+00							
.19817+03	.18302+03	.21968+03	.21549+03	.20285+03	.20045+03	-.35689+03	.45197+03
.35097+03	.29528+03	-.16232+04	-.33666+03	.33498+03	.50009+03	.62607+02	.60959+02
.85120+02	.38492+02	-.93476+02	-.17294+02	-.51030+01	-.13405+02	-.16697-01	.21618+01
.00000	.00000	.00000	.00000	.00000	-.53875+04	-.62936+03	-.21942+05
.21228+06	.26955+06	.31168+06	.53539+05	.11948+06	-.30581+06	-.35452+07	-.11975+07
.29499+00							
.19972+03	.18486+03	.21119+03	.21416+03	.20445+03	.20295+03	-.35590+03	.45298+03
.27281+03	.43514+03	-.16885+04	-.28151+03	.30851+03	.49744+03	.62543+02	.60743+02
.83149+02	.42609+02	-.72966+02	-.12042+03	.18284+01	-.89610+00	.10172+00	.25055+01
.00000	.00000	.00000	.00000	.00000	-.46632+04	-.56876+03	-.21655+05
.20501+06	.25045+06	.31330+06	.44769+05	.11911+06	-.26416+06	-.32992+07	-.11796+07

.29999+00							
.20095+03	.18733+03	.20358+03	.21190+03	.20595+03	.20540+03	-.35489+03	.45401+03
.22508+03	.55146+03	-.12859+04	-.69092+03	.29166+03	.48113+03	.62482+02	.60527+02
.79774+02	.44577+02	.49844+02	-.19121+02	.13985+02	.10305+02	.17737+00	.24727+01
.00000	.00000	.00000	.00000	.00000	-.38330+04	-.61590+03	-.21032+05
.19631+06	.23425+06	.31438+06	.10988+06	.11538+06	-.21640+06	-.30961+07	-.11448+07
.30499+00							
.20203+03	.19031+03	.19864+03	.20695+03	.20738+03	.20775+03	-.35389+03	.45504+03
.20964+03	.63779+03	-.68441+03	-.12963+04	.27783+03	.45868+03	.62423+02	.60311+02
.75133+02	.44574+02	.19209+02	.10847+03	.21001+02	.20277+02	.18343+00	.19198+01
.00000	.00000	.00000	.00000	.00000	-.28838+04	-.77669+03	-.20120+05
.18677+06	.22182+06	.31500+06	.20616+06	.10993+06	-.16182+06	-.29474+07	-.10949+07
.30999+00							
.20309+03	.19366+03	.19637+03	.19931+03	.20874+03	.20998+03	-.35287+03	.45608+03
.22088+03	.69813+03	-.28274+03	-.16998+04	.26755+03	.43336+03	.62366+02	.60096+02
.69947+02	.43832+02	-.45068+02	.48133+02	.24422+02	.55945+02	.13414+00	.15334+01
.00000	.00000	.00000	.00000	.00000	-.18430+04	-.10421+04	-.18988+05
.17662+06	.21274+06	.31528+06	.27031+06	.10364+06	-.10202+06	-.28474+07	-.10333+07
.31499+00							
.20427+03	.19724+03	.19488+03	.19090+03	.21006+03	.21208+03	-.35186+03	.45713+03
.25581+03	.71819+03	-.34352+03	-.16093+04	.26425+03	.40238+03	.62310+02	.59881+02
.64260+02	.41674+02	.46018+02	-.86844+02	.32005+02	.63175+02	.45093-01	.14664+01
.00000	.00000	.00000	.00000	.00000	-.76867+03	-.14059+04	-.17852+05
.16624+06	.20701+06	.31056+06	.25593+06	.96016+05	-.40316+05	-.27959+07	-.97270+06
.31999+00							
.20567+03	.20040+03	.19264+03	.18390+03	.21139+03	.21400+03	-.35083+03	.45818+03
.30581+03	.50338+03	-.58588+03	-.11302+04	.26926+03	.36630+03	.62257+02	.59668+02
.57658+02	.46041+02	.10768+03	-.68298+02	.43120+02	.41034+02	-.73853-01	.12541+01
.00000	.00000	.00000	.00000	.00000	.28507+03	-.18629+04	-.16918+05
.15614+06	.20356+06	.27292+06	.17974+06	.87174+05	.20203+05	-.27785+07	-.92489+06
.32499+00							
.20729+03	.20210+03	.18887+03	.17988+03	.21276+03	.21574+03	-.34980+03	.45924+03
.33506+03	.16664+03	-.92332+03	-.46884+03	.28087+03	.33109+03	.62204+02	.59455+02
.51441+02	.60463+02	.34120+02	.60756+02	.51240+02	.41949+02	-.19158+00	.70331+00
.00000	.00000	.00000	.00000	.00000	.12619+04	-.23572+04	-.16279+05
.14706+06	.19779+06	.22140+06	.74560+05	.78311+05	.76335+05	-.27322+07	-.89428+06
.32999+00							
.20895+03	.20209+03	.18379+03	.17891+03	.21419+03	.21731+03	-.34876+03	.46031+03
.32174+03	-.18337+03	-.10131+04	.00000	.28813+03	.29808+03	.62153+02	.59244+02
.47047+02	.79725+02	-.18106+03	.28553+03	.56495+02	.66738+02	-.24103+00	.47538+00
.00000	.00000	.00000	.00000	.00000	.21932+04	-.27758+04	-.15947+05
.13809+06	.18646+06	.16112+06	.00000	.70045+05	.12991+06	-.26083+07	-.88113+06

.33499+00
.21043+03 .20031+03 .17972+03 .17891+03 .21563+03 .21873+03 -.34771+03 .46139+03
.26317+03 -.51605+03 -.61898+03 .00000 .28658+03 .26916+03 .62103+02 .59034+02
.44960+02 .10404+03 -.26882+03 -.31734+02 .62885+02 .63579+02 -.20926+00 .41344+00
.00000 .00000 .00000 .00000 .00000 .30732+04 -.29791+04 -.15955+05
.12874+06 .16826+06 .98437+05 .00000 .62969+05 .18056+06 -.23820+07 -.88695+06

.33999+00
.21152+03 .19721+03 .17741+03 .17891+03 .21704+03 .22001+03 -.34666+03 .46247+03
.16220+03 -.69312+03 -.31860+03 .00000 .27421+03 .24356+03 .62054+02 .58825+02
.45439+02 .12101+03 .52285+03 -.27911+03 .70408+02 .58631+02 -.95870-01 .37247+00
.00000 .00000 .00000 .00000 .00000 .38785+04 -.28392+04 -.16331+05
.11829+06 .14279+06 .50667+05 .00000 .56777+05 .22690+06 -.20410+07 -.91346+06

.34499+00
.21201+03 .19363+03 .17640+03 .17891+03 .21835+03 .22117+03 -.34560+03 .46356+03
.32427+02 -.71930+03 -.97303+02 .00000 .24923+03 .22095+03 .62007+02 .58619+02
.48570+02 .12204+03 .49408+03 -.54005+02 .76966+02 .26356+02 .85414-01 .29573+00
.00000 .00000 .00000 .00000 .00000 .45603+04 -.23545+04 -.17008+05
.10618+06 .11184+06 .15474+05 .00000 .51341+05 .26602+06 -.16082+07 -.95687+06

.34999+00
.21164+03 .19025+03 .17659+03 .17859+03 .21950+03 .22222+03 -.34454+03 .46465+03
-.10132+03 -.57881+03 .13664+03 -.13663+03 .20719+03 .20109+03 .61960+02 .58414+02
.53664+02 .12711+03 -.43951+01 .18304+03 .81472+02 .19722+02 .29479+00 .23695+00
.00000 .00000 .00000 .00000 .00000 .51142+04 -.16055+04 -.17936+05
.91672+05 .78112+05 .00000 .21729+05 .46677+05 .29763+06 -.11230+07 -.10143+07

.35499+00
.21104+03 .18799+03 .17750+03 .17768+03 .22040+03 .22318+03 -.34348+03 .46575+03
-.21145+03 -.32329+03 .20882+03 -.20882+03 .14869+03 .17917+03 .61915+02 .58211+02
.59963+02 .13126+03 -.14315+03 .31438+03 .86827+02 .12966+02 .48028+00 .33796+00
.00000 .00000 .00000 .00000 .00000 .56258+04 -.74436+03 -.19071+05
.73950+05 .44577+05 .00000 .33208+05 .41804+05 .32677+06 -.63229+06 -.10838+07

.35999+00
.20981+03 .18697+03 .17843+03 .17675+03 .22097+03 .22401+03 -.34243+03 .46685+03
-.27167+03 -.88129+02 .13971+03 -.13969+03 .76807+02 .15375+03 .61871+02 .58011+02
.67529+02 .12024+03 .16144+03 -.83269+02 .94596+02 .73757+01 .57935+00 .53909+00
.00000 .00000 .00000 .00000 .00000 .61374+04 .37798+02 -.20284+05
.53329+05 .14837+05 .00000 .22215+05 .36141+05 .35604+06 -.19436+06 -.11581+07

.36499+00
.20844+03 .18694+03 .17878+03 .17640+03 .22116+03 .22470+03 -.34138+03 .46796+03
-.23768+03 .40752+02 .00000 .00000 -.17140+01 .12022+03 .61829+02 .57813+02
.74625+02 .10765+03 -.40805+02 -.15547+03 .10287+03 -.21722+02 .58612+00 .67500+00
.00000 .00000 .00000 .00000 .00000 .67058+04 .00000 -.21385+05
.30267+05 .00000 .00000 .00000 .28767+05 .38900+06 .00000 -.12269+07

TABLE 8 (Concluded)

.36999+00							
.20772+03	.18711+03	.17878+03	.17640+03	.22096+03	.22520+03	-.34034+03	.46906+03
-.54861+02	.19922+02	.00000	.00000	-.77791+02	.80334+02	.61789+02	.57617+02
.74864+02	.93544+02	-.38609+03	-.15048+02	.11043+03	-.23324+02	.38080+00	.76350+00
.00000	.00000	.00000	.00000	.00000	.74528+04	.00000	-.22213+05
.68401+04	.00000	.00000	.00000	.19513+05	.43292+06	.00000	-.12805+07
.37499+00							
.20769+03	.18718+03	.17878+03	.17640+03	.22041+03	.22548+03	-.33930+03	.47017+03
.20086+02	.11226+02	.00000	.00000	-.11764+03	.30741+02	.61751+02	.57424+02
.68455+02	.77274+02	-.38609+03	.14181+03	.11290+03	.28853+01	.36348+00	.12427+01
.00000	.00000	.00000	.00000	.00000	.00000	.00000	-.22688+05
.00000	.00000	.00000	.00000	.80131+04	.00000	.00000	-.13145+07
.37999+00							
.20772+03	.18721+03	.17878+03	.17640+03	.22004+03	.22553+03	-.33827+03	.47127+03
-.11969+02	.25816+01	.00000	.00000	-.53965+01	.60321+01	.61714+02	.57234+02
.63360+02	.64855+02	-.33194+03	.56208+02	.11300+03	.13406+03	.63118+00	-.29803+00
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
.38499+00							
.20759+03	.18721+03	.17878+03	.17640+03	.22016+03	.22560+03	-.33723+03	.47237+03
-.35537+02	-.26823+01	.00000	.00000	.34884+02	.20322+02	.61680+02	.57047+02
.72202+02	.56745+02	-.12352+02	-.11682+03	.10414+03	.13075+03	.95632+00	-.47804+00
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
.38999+00							
.20741+03	.18721+03	.17878+03	.17640+03	.22038+03	.22571+03	-.33619+03	.47347+03
-.33945+02	.34877+01	.00000	.00000	.48657+02	.20606+02	.61646+02	.56862+02
.76738+02	.48013+02	-.27965+03	-.87468+02	.94568+02	.12439+03	.10249+01	-.57075+00
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
.39499+00							
.20724+03	.18722+03	.17878+03	.17640+03	.22063+03	.22581+03	-.33515+03	.47457+03
-.34274+02	.48875+00	.00000	.00000	.51725+02	.17568+02	.61615+02	.56680+02
.74334+02	.43399+02	-.38609+03	.83820+02	.96431+02	.11263+03	.91615+00	-.34137+00
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
.39999+00							
.20709+03	.18723+03	.17878+03	.17640+03	.22089+03	.22588+03	-.33412+03	.47568+03
-.26401+02	-.13825+00	.00000	.00000	.55430+02	.11476+02	.61586+02	.56500+02
.67826+02	.40858+02	.79884+03	.10691+03	.94850+02	.98730+02	.68998+00	-.66771+00
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000
.00000	.00000	.00000	.00000	.00000	.00000	.00000	.00000

5. IITRAIN CODE LISTING

5.1 IITRAIN Subprograms

The IITRAIN computer code is composed of an executive program, 42 subprograms and a procedures definition processor (PDP) deck. Table 9 is a list of these programs and their functions.

TABLE 9.-IITRAIN PROGRAMS

Program	Function
STUFF	PDP program defining dimensions for all arrays used in IITRAIN
EXEC	Executive program
START	Controls restart feature
INPT	Reads and echo prints input data
INIT	Initializes all variables
INTG	Controls integration procedure
FNSH	Terminates calculations and saves data necessary for further processing
HEAD	Prints heading for motion and force output
COMOD	Determines constants for combinations of nonlinear springs
EULR	Euler integration subroutine
FINT	Controls calculation of internal forces
FEXT	Controls calculation of external forces
ACCL	Computes accelerations
OUTP	Controls output of motions and forces
GRUP	Computes accelerations for masses connected to constraint elements
LSPR	Computes internal forces for linear spring elements
LDSH	Computes internal forces for linear dashpot elements
TRSP	Computes internal forces for linear torsional spring elements
TRDP	Calculates internal forces for linear torsional dashpot elements
BEAM	Calculates internal forces for elastic-plastic beam elements
STRS	Calculates stresses in beam elements

TABLE 9.-IITRAIN PROGRAMS (Concl)

Program	Function
PINN	Calculates internal force data for pin element connections
SLDR	Calculates internal force data for slider element connections
SPIN	Calculates internal force data for sliding pin element connections
DSLD	Calculates internal force data for double slider element connections
RIGD	Calculates internal force data for rigid element connections
CPL 1	Calculates internal forces for type 1 coupler elements
CPL 2	Calculates internal forces for type 2 coupler elements
DGSP	Calculates internal forces for type 3 draft gear elements
END 3	Calculates internal forces for type 3 coupler end elements
ACLMR 1	Calculates internal forces for anticlimber elements
TRILN	Calculates force data for ACLMR 1
NLTS	Calculates internal forces for nonlnear torsional spring elements
WRIN	Calculates internal forces for wheel-rail interaction elements
NLSP	Calculates internal forces for nonlinear spring elements
NLDS	Calculates internal forces for nonlinear dashpot elements
SLSPR	Calculates internal forces for special linear spring elements
TAPB	Calculates internal forces for elastic-plastic tapered beam elements
RKIN	Calculates rigid body motion data
LEQS	Linear equation solver
MALG	Performs matrix algebra
DOTP	Forms dot-product
CRSP	Forms cross-product
INTP	Linear interpolation routine

5.2 COMMON Storage Blocks

Tables 10 through 19 list the contents of the COMMON blocks used in the IITRAIN program.

TABLE 10.-TIME COMMON BLOCK

Variable	Definition
IM	Integration method code
TD	Integration time step
TF	Final step
IS(I)	Time step multiple for mass class I
TT	Time
ITS	Initial step time switch

TABLE 11.-STAT COMMON BLOCK

Variable	Definition
XP(I)	x position of c.g. of mass I
YP(I)	y position of c.g. of mass I
AP(I)	Angular position of mass I
XV(I)	x velocity of c.g. of mass I
YV(I)	y velocity of c.g. of mass I
AV(I)	Angular velocity of mass I
XA(I)	x acceleration of c.g. of mass I
YA(I)	y acceleration of c.g. of mass I
AA(I)	Angular acceleration of mass I

TABLE 12.-MASS COMMON BLOCK

Variable	Definition
IN	Number of masses
II(I)	Integration time step class for mass I
IF(I,J)	Fixity for mass I (see input format)
WT(I)	Weight of mass I
RI(I)	Rotational inertia about c.g. for mass I
FA(I)	Fixity angle for mass I (see input format)
FF(I,J)	Fixity parameters for mass I

TABLE 13.-ELEM COMMON BLOCK

Variable	Definition
IE	Number of elements
IT(I)	Element type for element I
ID(I)	Physical parameter identification for element I
IA(I,J)	Identification of masses connected to element I
IP(I,J,K)	Attachment points on masses connected to element I
PP(I,J,K)	Physical parameter set J for element type K

TABLE 14.-CONN COMMON BLOCK

Variable	Definition
IC(I)	Number of contact points on mass I
XC(I,J)	x coordinate of contact point J on mass I
YC(I,J)	y coordinate of contact point J on mass I
AC(I,J)	Angle associated with contact point J on mass I

TABLE 15.-GRAV COMMON BLOCK

Variable	Definition
GA	Track elevation angle
GG	Acceleration due to gravity
FG(I,J)	Components of weight vector for mass I

TABLE 16.-WRITE COMMON BLOCK

Variable	Definition
IW	Counter for output
NM	Number of motion outputs
NF	Number of force outputs
IWW	Number of time steps per printout
IW 1	Counter for output
JI(I)	Mass identification for motion output I
JJ(I)	Point identification for motion output I
JTY(I)	Type of motion required for motion output I
JDR(I)	Direction for motion output I
KE(I)	Element identification for force output I
KI(I)	Mass identification for force output I
KJ(I)	Point identification for force output I
KDR(I)	Direction for force output I

TABLE 17.-FORC COMMON BLOCK

Variable	Definition
FI(I,J,K)	Internal force acting on mass I from element J in K direction
FE(I,J)	External force acting on mass I in J direction
F(I,J)	Total force acting on mass I in J direction

TABLE 18.-SAVE COMMON BLOCK

Variable	Definition
SAVE(I,J)	Save parameter for element J

TABLE 19.-PAIR COMMON BLOCK

Variable	Definition
NG	Number of groups of constraint masses
NP	Number of pairs of constraint masses
MP(I,J)	Mass identification for mass pair I
MG(I,J)	Group identification
NMG(I)	Number of masses in group I
MI(I,J)	Counter for constraint element calculations
MJ(I,J,K)	Counter for constraint element calculations
MK(I,J)	Counter for constraint element calculations
ML(I,J,K)	Counter for constraint element calculations

```
#PDP.ILF STUFF
PDP10 RL70-6 06/02-14130155-(.0)
PE0001       TIME  PROC
  0002              COMMON/TIME/IM.TO.TF.IS(5).TI.ITS
  0003        END
PE0004       STAT  PROC
  0005              COMMON/STAT/XP(NUMM).YP(NUMM).AP(NUMM).
  0006             1            XV(NUMM).YV(NUMM).AV(NUMM).
  0007             2            XA(NUMM).YA(NUMM).AA(NUMM)
  0008        END
PE0009       MASS PROC
  0010              COMMON/MASS/IN.II(NUMM).IF(NUMM.4).MT(NUMM).RI(NUMM).FA(NUMM).
  0011             1  FF(NUMM.4)
  0012        END
PE0013       ELEM  PROC
  0014              COMMON/ELEM/IE.IT(NELE).ID(NELE).IA(NELE.3).IP(NELE.3.3).
  0015             1  PP(29.NEIDS.NTYPES)
  0016        END
PE0017       CONN  PROC
  0018              COMMON/CONN/IC(NUMM).XC(NUMM.NCONPT).YC(NUMM.NCONPT).
  0019             1      AC(NUMM.NCONPT)
  0020        END
PE0021       GRAV  PROC
  0022              COMMON/GRAV/GA.GG.FG(NUMM.3)
  0023        END
PE0024       WRITE PROC
  0025              COMMON/WRITE/IW.NM.NF.IWW.IWI.JI(NMOT).JJ(NMOT).JTY(NMOT).
  0026             1  JDR(NMOT).KE(NFORC).KI(NFORC).KJ(NFORC).KDR(NFORC)
  0027        END
PE0028       FORC  PROC
  0029              COMMON/FORC/FI(NUMM.NELE.3).FE(NUMM.3).F(NUMM.3)
  0030        END
PE0031       SAVE  PROC
  0032              COMMON/SAVE/SV(70.NELE)
  0033        END
PE0034       PAIR  PROC
  0035              COMMON/PAIR/NG.NP.MP(4.NPAIRS).MG(NUMM.2).NMG(NGRUPS).MI(NMPG.
  0036             1  NGRUPS).MJ(NMPG.NGRUPS.NATACH).MK(NMPG.NGRUPS).ML(NMPG.NGRUPS.
  0037             2   NATACH)
  0038        END
PE0039       PARM  PROC
  0040       C     NUMM IS THE TOTAL NUMBER OF MASSES
  0041       C     NELE IS THE TOTAL NUMBER OF ELEMENTS
  0042       C     NEIDS IS THE NUMBER OF DIFFERENT IDIS ALLOWED PER ELEMENT
  0043       C     NTYPES ID THE NUMBER OF DIFFERENT ELEMENT TYPES
  0044       C     NCONPT IS THE NUMBER OF CONNECTION POINTS PER MASS
  0045       C     NMOT  IS THE NUMBER OF MOTION OUTPUTS ALLOWED
  0046       C     NFORC IS THE NUMBER OF FORCE OUTPUTS ALLOWED
  0047       C     NPAIRS IS THE NUMBER OF PAIRS OR CONSTRAINTS
  0048       C     NMPG IS THE NUMBER OF MASSES PER GROUP
  0049       C     NGRUPS IS THE NUMBER OF GROUPS OF MASSES
  0050       C     NATACH IS THE NUMBER OF CONSTRAINT ELEMENTS ATTACHED TO A SINGLE MASS
  0051              PARAMETER    NUMM=                          48
  0052             1.            NELE=                         180
  0053             2.             NEIDS=                         5
  0054             3.             NTYPES=                       22
  0055             4.             NCONPT=                       10
```

```
0056            5,              NMOT=                           30
0057            6,              NFORC=                          18
0058            7,               NPAIRS=                         5
0059            8,               NMPG=                           4
0060            9,               NGRUPS=                         5
0061            A,               NATACH=                         4
0062            B,               NROWF1=                      3*NMPG
0063            C,               NROWF2=                    3*NPAIRS
0064            D,               NOUT=                     NUMM+NELE
0065             PARAMETER KSTAT = 9*NUMM
0066            F,      KMASS1 = 1+5*NUMM
0067            G,      KMASS2 = 7*NUMM
0068            H,      KELEM1 = 1+14*NELE
0069            I,      KELEM2 = 29*NEIDS*NTYPES
0070            J,      KCONN1 = NUMM
0071            K,      KCONN2 = 3*NUMM*NCONPT
0072            L,      KGRAV = 2+3*NUMM
0073            M,      KWRIT = 5+4*NMOT+4*NFORC
0074            N,      KFORC = 3*NUMM*(NELE+2)
0075            O,      KSAVE = 70*NELE
0076            P,      KPAIR = 2*(NMPG*NGRUPS*(NATACH+2))+NGRUPS+2*NUMM+4*NPAIRS+2
0077        END

END PDP  ERRORS : NONE
```

```
*FOR,IS .EXEC
FOR 8E38-06/02/77-14:31:22 (.0)

   MAIN PROGRAM

   STORAGE USED: CODE(1) 000045: DATA(0) 000021: BLANK COMMON(2) 000000


   EXTERNAL REFERENCES (BLOCK, NAME)

    0003    START
    0004    INPT
    0005    INIT
    0006    INTG
    0007    FNSH
    0010    NINTR$
    0011    NRDU$
    0012    NIO2$
    0013    NWDU$
    0014    NSTOP$


   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000041 IL         0000    000002 IOF        0000    000003 IIF        0001    000026 2L         0001    000032 3L
    0000 I  000000 IUNIT      0000 I  000001 JUNIT


00101       1*            COMPILER(XM=1),(ADR=IND)                                                          000000
00103       2*            READ(5,10) IUNIT,JUNIT                                                            000001
00110       3*         10 FORMAT(2I5)                                                                       000010
00111       4*            WRITE(6,11) IUNIT,JUNIT                                                           000010
00115       5*         11 FORMAT('  INPUT FILE NO. =',I5,10X,'OUTPUT FILE NO. =',I5,//)                     000017
00116       6*            IF(IUNIT .EQ. 0) GO TO 2                                                          000017
00120       7*            CALL START(IUNIT)                                                                 000021
00121       8*            GO TO 3                                                                           000024
00122       9*          2 CALL INPT                                                                         000026
00123      10*            CALL INIT                                                                         000027
00124      11*          3 CALL INTG                                                                         000032
00125      12*            IF(JUNIT .EQ. 0) GO TO 1                                                          000033
00127      13*            CALL FNSH(JUNIT)                                                                  000035
00130      14*          1 STOP                                                                              000041
00131      15*            END                                                                               000044

       END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS .START
FOR SE38-06/02/77-14:31:28 (,0)

   SUBROUTINE START      ENTRY POINT 000142

   STORAGE USED: CODE(1) 001471; DATA(0) 000121; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003     TIME     000012
    0004     STAT     000660
    0005     MASS     001101
    0006     ELEM     013117
    0007     CONN     002720
    0010     GRAV     000222
    0011     WRITE    000305
    0012     FORC     063140
    0013     SAVE     030470
    0014     PAIR     000553

   EXTERNAL REFERENCES (BLOCK, NAME)

    0015     NRBU$
    0016     NI03$
    0017     NI01$
    0020     NI02$
    0021     NRDU$
    0022     NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000     000002 11F     0004 R 000000 A      0004   000600 AA     0007   001760 AC     0004   000140 AP
    0004     000350 AV      0005 R 000361 BB     0006 R 004731 CC     0007 R 000060 DD     0010 R 000000 E
    0012     062720 F       0005   000521 FA     0012   062500 FE     0005   000601 FF     0010   000002 FG
    0012     000000 FI      0012 R 000000 G      0010   000000 GA     0010   000001 GG     0013 R 000000 H
    0006     000551 IA      0007   000000 IC     0006   000265 ID     0006   000000 IE     0005   000061 IF
    0005     000001 II      0003 I 000000 IM     0005   000000 IN     0000   000005 INJP$  0006   001605 IP
    0003 I   000003 IS      0006   000001 IT     0003 I 000011 ITS    0011   000000 IW     0011   000003 IWW
    0011     000004 IW1     0005 I 000000 JA     0006 I 000000 JC     0007 I 000000 JD     0011   000137 JDH
    0011 I   000000 JF      0011   000005 JI     0011   000043 JJ     0014 I 000000 JPU    0011   000101 JTY
    0000 I   000000 K       0011   000263 KDR    0011   000175 KE     0011   000217 KI     0011   000241 KJ
    0000 I   000001 L       0014   000026 MG     0014   000173 MI     0014   000217 MJ     0014   000337 MK
    0014     000363 ML      0014   000002 MP     0011   000002 NF     0014   000000 NG     0011   000001 NM
    0014     000166 NMG     0014   000001 NP     0006   004731 PP     0005   000441 RI     0013   000000 SV
    0003 R   000001 TD      0003 R 000002 TF     0003 R 000010 TT     0005   000361 WT     0004   000440 XA
    0007     000060 XC      0004   000000 XP     0004   000220 XV     0004   000520 YA     0007   001020 YC
    0004     000060 YP      0004   000300 YV


00101      1*          COMPILER(XM=1),(ADR=IND)                                          000000
00103      2*          SUBROUTINE START(IUNIT)                                           000000
```

```
00105      3*           INCLUDE PARM                                                                    000000
00110      4*           INCLUDE TIME                                                                    000000
00112      5*           INCLUDE STAT                                                                    000000
00114      6*           INCLUDE MASS                                                                    000000
00116      7*           INCLUDE ELEM                                                                    000000
00120      8*           INCLUDE CONN                                                                    000000
00122      9*           INCLUDE GRAV                                                                    000000
00124     10*           INCLUDE WRITE                                                                   000000
00126     11*           INCLUDE FORC                                                                    000000
00130     12*           INCLUDE SAVE                                                                    000000
00132     13*           INCLUDE PAIR                                                                    000000
00134     14*           DIMENSION A(KSTAT),JB(KMASS1),BB(KMASS2),JC(KELEM1),CC(KELEM2),                 000000
00134     15*          1      JD(KCONN1),DD(KCONN2),E(KGRAV),JF(KWRIT),G(KFORC),H(KSAVE),               000000
00134     16*          2      JPQ(KPAIR)                                                               000000
00135     17*           EQUIVALENCE (A(1),XP(1)),          (JB(1),IN),                                  000000
00135     18*          1                (BB(1),WT(1)),       (JC(1),IE),                               000000
00135     19*          2                (CC(1),PP(1,1,1)),   (JD(1),IC(1)),                           000000
00135     20*          3                (DD(1),XC(1,1)),     (E(1),GA),                                000000
00135     21*          4                (JF(1),IW),          (G(1),FI(1,1,1)),                         000000
00135     22*          5                (H(1),SV(1,1)),      (JPQ(1),NG)                               000000
00136     23*           READ(IUNIT)  IM,TD,TF,(IS(K),K=1,5),TT,ITS                                      000000
00146     24*           READ(IUNIT)  (A(K),K=1,KSTAT)                                                   000014
00151     25*           READ(IUNIT)  (JB(K),K=1,KMASS1),(BB(L),L=1,KMASS2)                              000023
00155     26*           READ(IUNIT)  (JC(K),K=1,KELEM1),(CC(L),L=1,KELEM2)                              000035
00161     27*           READ(IUNIT)  (JD(K),K=1,KCONN1),(DD(L),L=1,KCONN2)                              000047
00165     28*           READ(IUNIT)  (E(K),K=1,KGRAV)                                                   000061
00170     29*           READ(IUNIT)  (JF(K),K=1,KWRIT)                                                  000070
00173     30*           READ(IUNIT)  (G(K),K=1,KFORC)                                                   000077
00176     31*           READ(IUNIT)  (H(K),K=1,KSAVE)                                                   000106
00201     32*           READ(IUNIT)  (JPQ(K),K=1,KPAIR)                                                 000115
00204     33*           READ(5,11) TF                                                                   000124
00207     34*        11 FORMAT(F10.0)                                                                   000132
00210     35*           RETURN                                                                          000132
00211     36*           END                                                                             000146

END OF COMPILATION!        NO  DIAGNOSTICS.
```

```
@FOR.IS .INPT
FOR SE38-06/02/77-14131138 (.0)

   SUBROUTINE INPT      ENTRY POINT 001453

   STORAGE USED: CODE(1) 0014741 DATA(0) 0003471 BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   TIME   000012
    0004   STAT   000660
    0005   MASS   001101
    0006   ELEM   013117
    0007   CONN   002720
    0010   GRAV   000222
    0011   WRITE  000305

   EXTERNAL REFERENCES (BLOCK, NAME)

    0012   NRDU$
    0013   NIO3$
    0014   NIO2$
    0015   NWDU$
    0016   NIO1$
    0017   NBTOP$
    0020   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000    000032 1F        0000    000161 10F       0000    000112 100F      0000    000133 101F      0000    000152 102F
    0000    000203 103F      0000    000224 104F      0000    000163 11F       0000    000201 12F       0000    000210 13F
    0001    001037 14L       0000    000231 15F       0001    000237 153G      0000    000235 16F       0001    001424 17L
    0000    000077 18F       0000    000101 19F       0000    000033 2F        0000    000233 20F       0001    000307 200G
    0000    000056 21F       0001    000337 212G      0000    000075 22F       0000    000057 23F       0001    000405 237G
    0000    000243 24F       0001    000413 245G      0001    000454 261G      0001    000464 270G      0000    000035 3F
    0001    000513 303G      0001    000547 310G      0001    000571 321G      0001    000634 341G      0001    000667 355G
    0001    000722 373G      0001    000726 377G      0000    000041 4F        0001    000732 403G      0001    000733 405G
    0001    000775 421G      0001    001003 426G      0001    001007 432G      0001    001010 434G      0001    001057 452G
    0001    001144 464G      0001    001171 473G      0000    000106 5F        0001    001241 503G      0001    001265 514G
    0001    001345 524G      0001    001372 533G      0001    001417 542G      0000    000115 6F        0000    000137 7F
    0000    000141 8F        0004    000600 AA        0007 R  001760 AC        0004 R  000140 AP        0004 R  000360 AV
    0005 R  000521 FA        0005    000601 FF        0010    000002 FG        0010 R  000000 GA        0010 R  000001 GG
    0000 I  000024 I         0006 I  000551 IA        0007 I  000000 IC        0006 I  000265 ID        0000 I  000030 IDP
    0006 I  000000 IE        0005 I  000061 IF        0005 I  000001 II        0003 I  000000 IM        0005 I  000000 IN
    0000    000256 INJP$     0006 I  001605 IP        0003 I  000003 IR        0006 I  000001 IT        0000 I  000027 ITP
    0003    000011 ITB       0011    000000 IW        0011 I  000003 IWW       0011    000004 IW1       0000 I  000026 J
    0011 I  000137 JDR       0011 I  000005 JI        0011 I  000043 JJ        0011 I  000101 JTY       0000 I  000025 K
    0011 I  000263 KDR       0011 I  000175 KE        0011 I  000217 KI        0011 I  000241 KJ        0000 I  000031 KHED
    0011 I  000002 NF        0011 I  000001 NM        0006 R  004731 PP        0005 R  000441 RI        0003 R  000001 TD
    0003 R  000002 TF        0000 R  000000 TITL      0003    000010 TT        0005 R  000361 WT        0004    000440 XA
    0007 R  000060 XC        0004 R  000000 XP        0004 R  000220 XV        0004    000520 YA        0007 R  001020 YC
    0004 R  000060 YP        0004 R  000300 YV
```

```
00101      1*          COMPILER(XM=1),(ADR=IND)                                                    000167
00103      2*          SUBROUTINE INPT                                                             000167
00105      3*          INCLUDE PARM                                                                000167
00110      4*          INCLUDE TIME                                                                000167
00112      5*          INCLUDE STAT                                                                000167
00114      6*          INCLUDE MASS                                                                000167
00116      7*          INCLUDE ELEM                                                                000167
00120      8*          INCLUDE CONN                                                                000167
00122      9*          INCLUDE GRAV                                                                000167
00124     10*          INCLUDE WRITE                                                               000167
00126     11*          DIMENSION TITL(20)                                                          000167
00127     12*          READ(5,1,END=17) TITL                                                       000167
00132     13*        1 FORMAT(20A4)                                                                000200
00133     14*          WRITE(6,2) TITL                                                             000200
00136     15*        2 FORMAT(1H1,20A4)                                                            000210
00137     16*          READ(5,3) IM,TD,TF,(IS(I),I=1,5)                                            000210
00145     17*        3 FORMAT(I5,E15.0,E10.0,5I5)                                                  000223
00146     18*          WRITE(6,4) IM,TD,TF,(I,IS(I),I=1,5)                                         000223
00160     19*         40FORMAT(///3H IM,I10                                                        000245
00160     20*          1/3H TD,F15.10                                                              000245
00160     21*          2/3H TF,F15.10                                                              000245
00160     22*          3/8H I,IS(I),I7,1H,,I5                                                      000245
00160     23*          4/(I15,1H,,I5))                                                             000245
00161     24*          READ(5,21) NM,NF,IWW                                                        000245
00166     25*       21 FORMAT(3I5)                                                                 000255
00167     26*          WRITE(6,23) NM,NF,IWW                                                       000255
00174     27*       23 FORMAT(/20H NM=NO, MOTIONS OUT,I5                                           000265
00174     28*         1/20H NF=NO, FORCES OUT ,I5                                                 000265
00174     29*         2/20H IWW=STEPS PER PRINT,I5)                                               000265
00175     30*          IF(NM,NE,0) READ(5,22) (JI(K),JJ(K),JTY(K),JDR(K),K=1,NM)                   000265
00207     31*          IF(NF,NE,0) READ(5,22) (KE(K),KI(K),KJ(K),KDR(K),K=1,NF)                    000315
00221     32*       22 FORMAT(8(2I3,2I2))                                                          000345
00222     33*          READ(5,18) GA,GG                                                            000345
00226     34*       18 FORMAT(2E10.0)                                                              000354
00227     35*          WRITE(6,19) GA,GG                                                           000354
00233     36*       19 FORMAT(//3H GA,F10.3/3H GG,F10.3)                                           000363
00234     37*          READ(5,5)IN,(WT(I),RI(I),II(I),IC(I),(IF(I,J),J=1,4),FA(I),I=1,IN)          000363
00253     38*        5 FORMAT(I5/(2E10.0,6I5,E10.0))                                              000430
00254     39*          WRITE(6,100)                                                                000430
00256     40*      100 FORMAT(//10H MASS DATA)                                                     000435
00257     41*          WRITE(6,6) (I,WT(I),RI(I),II(I),IC(I) ,(IF(I,J),J=1,4),                     000435
00257     42*         CFA(I),I=1,IN)                                                               000435
00276     43*         60FORMAT( /5X1HI,10X2HWT,13X2HRI,7X2HII,5X2HIC,8X2HIF,13X2HFA                 000502
00276     44*          1//(I6,2F15.3,3I7,3I2,F15.3))                                               000502
00277     45*          WRITE(6,101)                                                                000502
00301     46*      101 FORMAT(//15H CONTACT POINTS)                                                000513
00302     47*          DO 9 I=1,IN                                                                 000513
00305     48*          J=IC(I)                                                                     000526
00306     49*          READ(5,7) (XC(I,K),YC(I,K),AC(I,K),K=1,J)                                   000530
00316     50*        7 FORMAT(3E10.0)                                                              000554
00317     51*          WRITE(6,8) (I,K,XC(I,K),YC(I,K),AC(I,K),K=1,J)                              000554
00331     52*         80FORMAT( /5X1HI,5X1HK,10X2HXC,13X2HYC,13X2HAC                               000605
00331     53*          1//(2I6,3F15.3))                                                            000605
00332     54*        9 CONTINUE                                                                    000605
00334     55*          WRITE(6,102)                                                                000605
00336     56*      102 FORMAT(//34H MASS GLOBAL POSITION AND VELOCITY)                             000612
```

```
00337     57*           READ(5,10) (XP(I),YP(I),AP(I),XV(I),YV(I),AV(I),I=1,IN)                    000612
00352     58*        10 FORMAT(6E10.0)                                                              000644
00353     59*           WRITE(6,11) (I,XP(I),YP(I),AP(I),XV(I),YV(I),AV(I),I=1,IN)                 000644
00367     60*        11 0FORMAT( /3H  I,I15                                                         000702
00367     61*          1/3H XP,F15.3                                                                000702
00367     62*          2/3H YP,F15.3                                                                000702
00367     63*          3/3H AP,F15.3                                                                000702
00367     64*          4/3H XV,F15.3                                                                000702
00367     65*          5/3H YV,F15.3                                                                000702
00367     66*          6/3H AV,F15.3)                                                               000702
00370     67*          0READ(5,12) IE,((IT(I),ID(I),(IA(I,J),J=1,3)                                 000702
00370     68*          1,((IP(I,J,K),K=1,3),J=1,3),I=1,IE)                                          000702
00413     69*        12 FORMAT(I5/(1415))                                                           000753
00414     70*           WRITE(6,103)                                                                000753
00416     71*       103 FORMAT(//20H ELEMENT CONNECTIONS)                                           000760
00417     72*           WRITE(6,13) (I,IT(I),ID(I),(IA(I,J),J=1,3)                                  000760
00417     73*          1,((IP(I,J,K),K=1,3),J=1,3),I=1,IE)                                          000760
00442     74*        13 0FORMAT( /5X1H1,4X2HIT,4X2HID,7X2HIA,20X2HIP                                001031
00442     75*          1//(4I6,2I3,I6,2I3,I4,2I3,I4,2I3))                                           001031
00443     76*           WRITE(6,104)                                                                001031
00445     77*       104 FORMAT(//21H ELEMENT DISCRIPTIONS)                                          001037
00446     78*        14 READ(5,15) ITP,IDP,(PP(I,IDP,ITP),I=1,9)                                    001037
00456     79*        15 FORMAT(2I4,9E8.0)                                                           001062
00457     80*           IF(ITP.EQ.0) RETURN                                                         001062
00461     81*           IF(ITP .EQ. 11 .OR. ITP .EQ. 12 .OR. ITP .EQ. 14 .OR. ITP .EQ. 15          001067
00461     82*          2  .OR. ITP .EQ. 22)                                                        001067
00461     83*          1              READ(5,20) (PP(I,IDP,ITP),I=10,19)                           001067
00470     84*           IF (ITP.EQ.15)                                                              001147
00470     85*          1              READ(5,20) (PP(1,IDP,ITP),I=20,29)                           001147
00477     86*           KRED=IFIX(PP(5,IDP,ITP)*3.+.2)+8                                           001174
00500     87*           IF(ITP .EQ. 5)                                                              001215
00500     88*          1     READ(5,20) (PP(I,IDP,ITP),I=10,KRED)                                   001215
00507     89*        20 FORMAT(10F8.0)                                                              001244
00510     90*           WRITE(6,16) ITP,IDP,(PP(I,IDP,ITP),I=1,9)                                   001244
00520     91*        16 FORMAT( /4H ITP,I5                                                          001270
00520     92*          1         /4H IDP,I5                                                         001270
00520     93*          2         /4H  PP,9F12.3)                                                    001270
00521     94*           IF(ITP .EQ. 11 .OR. ITP .EQ. 12 .OR. ITP .EQ. 14 .OR. ITP .EQ. 15          001270
00521     95*          2  .OR. ITP .EQ. 22)                                                        001270
00521     96*          1              WRITE(6,24) (PP(I,IDP,ITP),I=10,19)                          001270
00530     97*           IF (ITP.EQ.15)                                                              001350
00530     98*          1              WRITE(6,24) (PP(I,IDP,ITP),I=20,29)                          001350
00537     99*           IF(ITP .EQ. 5)                                                              001375
00537    100*          1     WRITE(6,24) (PP(I,IDP,ITP),I=10,KRED)                                  001375
00546    101*        24 FORMAT(4X,10F12.3)                                                          001422
00547    102*           GO TO 14                                                                    001422
00550    103*        17 STOP                                                                        001424
00551    104*           END                                                                         001473

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
#FOR.IS .INIT
FOR SE38-06/02/77-14:31:57 (.0)

   SUBROUTINE INIT      ENTRY POINT 004314

   STORAGE USED: CODE(1) 004341: DATA(0) 000254: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   TIME    000012
    0004   MASS    001101
    0005   ELEM    013117
    0006   STAT    000660
    0007   CONN    002720
    0010   SAVE    030470
    0011   PAIR    000503
    0012   GRAV    000222
    0013   WRITE   000305
    0014   FORC    063140

   EXTERNAL REFERENCES (BLOCK, NAME)

    0015   HEAD
    0016   RKIN
    0017   COMOD
    0020   SIN
    0021   COS
    0022   ATAN
    0023   SQRT
    0024   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   001655 10L      0001   003435 1026G    0001   003620 1057G    0001   001736 11L      0001   001042 112L
    0001   004211 1163G    0001   001742 12L      0001   002505 13L      0001   002333 14L      0001   000262 143G
    0001   002357 15L      0001   000305 153G     0001   000313 157G     0001   002440 16L      0001   002464 17L
    0001   001364 18L      0001   000347 2L       0001   000402 201G     0001   003065 23L      0001   000576 233G
    0001   002717 24L      0001   002743 25L      0001   003024 26L      0001   003050 27L      0001   001776 28L
    0001   002130 29L      0001   004251 30L      0001   000777 300G     0001   001025 311G     0001   001105 325G
    0001   002024 33L      0001   001133 333G     0001   002122 34L      0001   001217 354G     0001   002104 36L
    0001   003415 37L      0001   003576 3AL      0001   001140 4L       0001   001446 40L      0001   001576 445G
    0001   001167 5L       0001   000603 51L      0001   001760 525G     0001   001766 531G     0001   002010 544G
    0001   002033 556G     0001   002060 562G     0001   002067 566G     0001   001553 6L       0001   002147 611G
    0001   001605 7L       0001   002526 700G     0001   003106 764G     0001   001634 8L       0001   001644 9L
    0006   000600 AA       0007 R 001760 AC       0006 R 000140 AP       0000 R 000043 APART    0000 R 000000 AREA
    0000 R 000044 ATOT     0006   000360 AV       0000 R 000020 AX       0000 R 000021 AY       0000 R 000022 BX
    0000 R 000023 BY       0000 R 000014 C        0000 R 000055 CK       0000 R 000070 CT1      0000 R 000071 CT2
    0000 R 000063 DK       0000 R 000064 DL       0000 R 000066 DX       0000 R 000067 DY       0000 R 000105 D1
    0000 R 000074 D11      0000 R 000075 D12      0000 R 000106 D2       0000 R 000100 D21      0000 R 000101 D22
    0000 R 000107 D3       0000 R 000102 E1       0000 R 000072 E11      0000 R 000073 E12      0000 R 000103 E2
    0000 R 000076 E21      0000 R 000077 E22      0000 R 000104 E3       0014   062720 F        0004 R 000521 FA
    0014   062500 FE       0004 R 000601 FF       0012 R 000002 FG       0014 R 000000 FI       0000 R 000110 FM
    0000 R 000111 FN       0000 R 000112 FO       0000 R 000061 F1       0000 R 000062 F2       0012 R 000000 GA
```

```
0012    000001 GG        0000 R 000046 HGT       0000 R 000045 HNA       0000 R 000047 HNM       0000 R 000051 HTOT
0000 I  000010 I         0005 I 000551 IA        0007   000000 IC        0005 I 000265 ID        0005 I 000000 IE
0004 I  000041 IF        0004   000001 II        0000 I 000024 IJKLM     0003   000000 IM        0004 I 000000 IN
0000    000130 INJPS     0000 I 000060 INMG      0005 I 001605 IR        0003   000003 IS        0005 I 000001 IT
0003 I  000011 ITS       0000 I 000113 IV        0013 I 000000 IW        0013   000003 IWW       0013 I 000004 IW1
0000 I  000012 J         0013   000137 JDR       0013   000005 JI        0013   000043 JJ        0013   000101 JTY
0000 I  000013 K         0000 I 000052 KDIV      0013   000263 KDR       0013   000175 KE        0013   000217 KI
0013    000241 KJ        0000 I 000042 KSECT     0000 I 000050 KSS       0000 I 000016 L         0000 I 000017 M
0011 I  000026 MG        0000 I 000056 MG1       0000 I 000057 MG2       0011 I 000173 MI        0011 I 000217 MJ
0011 I  000337 MK        0011 I 000363 ML        0011 I 000002 MP        0000 I 000053 N         0013   000002 NA
0011 I  000000 NG        0013   000001 NM        0011 I 000166 NMG       0011 I 000001 NP        0005 R 004731 PP
0000 R  000011 RAD       0004   000441 R1        0000 R 000015 S         0000 R 000054 SK        0000 R 000065 SL
0010 R  000000 SV        0000 R 000041 TA        0003   000001 TD        0003   000002 TF        0003 R 000010 TT
0004 R  000361 WT        0006   000440 XA        0007 R 000060 XC        0000 R 000031 XDDL      0000 R 000037 XDDM
0000 R  000027 XDL       0000 R 000035 XDM       0000 R 000025 XL        0000 R 000033 XM        0006 R 000000 XP
0006    000220 XV        0006   000520 YA        0007 R 001020 YC        0000 R 000032 YDDL      0000 R 000040 YDDM
0000 R  000030 YDL       0000 R 000036 YDM       0000 R 000026 YL        0000 R 000034 YM        0006 R 000060 YP
0006    000300 YV
```

```
00101      1*            COMPILER(XM=1),(ADR=IND)                                    000220
00103      2*            SUBROUTINE INIT                                             000220
00105      3*            INCLUDE PARM                                                000220
00110      4*            INCLUDE TIME                                                000220
00112      5*            INCLUDE MASS                                                000220
00114      6*            INCLUDE ELEM                                                000220
00116      7*            INCLUDE STAT                                                000220
00120      8*            INCLUDE CONN                                                000220
00122      9*            INCLUDE SAVE                                                000220
00124     10*            INCLUDE PAIR                                                000220
00126     11*            INCLUDE GRAV                                                000220
00130     12*            INCLUDE WRITE                                               000220
00132     13*            INCLUDE FORC                                                000220
00134     14*             DIMENSION AREA(8)                                          000220
00135     15*            CALL HEAD                                                   000220
00136     16*            TT=0.0                                                      000222
00137     17*            ITS=0                                                       000223
00140     18*            IW=0                                                        000224
00141     19*            IW1=0                                                       000225
00142     20*            DO 2 I=1,IN                                                 000262
00145     21*            RAD=.01745329252*GA                                         000264
00146     22*            FG(I,1)=-WT(I)*SIN(RAD)                                     000266
00147     23*            FG(I,2)=-WT(I)*COS(RAD)                                     000273
00150     24*            FG(I,3)=0.0                                                 000301
00151     25*            MG(I,1)=0                                                   000302
00152     26*            DO 1 J=1,3                                                  000305
00155     27*            FF(I,J)=1.0-FLOAT(IF(I,J))                                  000305
00156     28*            DO 101 K=1,IE                                               000313
00161     29*            FI(I,K,J)=0.0                                               000313
00162     30*        101 CONTINUE                                                    000322
00164     31*          1 CONTINUE                                                    000322
00166     32*            FF(I,4)=0.0                                                 000322
00167     33*            IF(IF(I,4),EQ.0) GO TO 2                                    000324
00171     34*            C=COS(FA(I)*.017453293)                                     000326
00172     35*            S=SIN(FA(I)*.017453293)                                     000332
```

```
00173     36*          FF(1,1)=C*C                                                    000336
00174     37*          FF(1,2)=S*S                                                    000341
00175     38*          FF(1,4)=S*C                                                    000343
00176     39*        2 CONTINUE                                                       000402
00200     40*          DO 5 I=1,IE                                                    000402
00200     41*    C                                                                    000402
00203     42*          IF(IT(I).NE.22) GO TO 51                                       000407
00205     43*          J=IA(I,1)                                                      000412
00206     44*          K=IA(I,2)                                                      000414
00207     45*          L=IP(I,1,1)                                                    000416
00210     46*          M=IP(I,2,1)                                                    000426
00211     47*          AX=0.0                                                         000437
00212     48*          AY=0.0                                                         000440
00213     49*          BX=XC(J,L)                                                     000441
00214     50*          BY=YC(J,L)                                                     000443
00215     51*          IJKLM = 1                                                      000445
00216     52*          CALL RKIN(AX,AY,BX,BY,J,IJKLM,XL,YL,XDL,YDL,XDDL,YDDL)         000447
00217     53*          BX=XC(K,M)                                                     000467
00220     54*          BY=YC(K,M)                                                     000472
00221     55*          CALL RKIN(AX,AY,BX,BY,K,IJKLM,XM,YM,XDM,YDM,XDDM,YDDM)         000474
00222     56*          SV(1,I)=XM-XL                                                  000514
00223     57*          SV(2,I)=YM-YL                                                  000517
00224     58*          TA=ATAN(SV(2,1)/SV(1,I))                                       000522
00225     59*          SV(3,1)=AP(J)-TA                                               000530
00226     60*          SV(4,I)=AP(K)-TA                                               000532
00227     61*          SV(5,I)=SQRT(SV(1,I)**2+SV(2,I)**2)                            000535
00230     62*          L=ID(I)                                                        000547
00231     63*          K=5+6*IFIX(PP(9,L,22)+.1)                                      000553
00232     64*          DO 52 J=6,K,3                                                  000566
00235     65*          SV(J,I)=0.0                                                    000576
00236     66*          SV(J+1,I)=0.0                                                  000576
00237     67*          SV(J+2,I)=PP(3,L,22)                                           000577
00240     68*       52 CONTINUE                                                       000603
00242     69*       51 CONTINUE                                                       000603
00242     70*    C                                                                    000603
00243     71*          IF(IT(I).NE.5) GO TO 4                                         000603
00245     72*          J=IA(I,1)                                                      000605
00246     73*          K=IA(I,2)                                                      000607
00247     74*          L=IP(I,1,1)                                                    000611
00250     75*          M=IP(I,2,1)                                                    000621
00251     76*          AX=0.0                                                         000632
00252     77*          AY=0.0                                                         000633
00253     78*          BX=XC(J,L)                                                     000634
00254     79*          BY=YC(J,L)                                                     000636
00255     80*          IJKLM = 1                                                      000640
00256     81*          CALL RKIN(AX,AY,BX,BY,J,IJKLM,XL,YL,XDL,YDL,XDDL,YDDL)         000642
00257     82*          BX=XC(K,M)                                                     000662
00260     83*          BY=YC(K,M)                                                     000664
00261     84*          CALL RKIN(AX,AY,BX,BY,K,IJKLM,XM,YM,XDM,YDM,XDDM,YDDM)         000667
00262     85*          SV(1,I)=XM-XL                                                  000707
00263     86*          SV(2,I)=YM-YL                                                  000712
00264     87*          TA=ATAN(SV(2,1)/SV(1,1))                                       000715
00265     88*          SV(3,I)=AP(J)-TA                                               000723
00266     89*          SV(4,I)=AP(K)-TA                                               000725
00267     90*          SV(5,I)=SQRT(SV(1,I)**2+SV(2,I)**2)                            000730
00270     91*          L=ID(I)                                                        000742
00270     92*    C     NEUTRAL AXIS DETERMINATION                                     000742
```

```
00270     93*     C      KSECT IS THE NUMBER OF BLOCK SECTIONS IN THE BEAM                                  000742
00271     94*            KSECT=IFIX(PP(5,L,5)+.1)                                                          000751
00272     95*            APART=0.0                                                                         000763
00273     96*            ATOT=0.0                                                                          000764
00274     97*            HNA=0.0                                                                           000765
00275     98*            HGT=0.0                                                                           000766
00276     99*            HNM=0.0                                                                           000767
00277    100*            DO 110 KSS=1,KSECT                                                                000777
00302    101*            AREA(KSS)= PP(3+3*KSS,L,5)*PP(4+3*KSS,L,5)                                        001000
00302    102*     C      HNM/AREA IS THE POSITION OF THE NEUTRAL AXIS IN ELASTIC DEFORMATION               001000
00303    103*            HNM= HNM +(2.*HGT+0.5*PP(3+3*KSS,L,5))*AREA(KSS)                                  001003
00304    104*            HGT=HGT + 0.5*PP(3+3*KSS,L,5)                                                     001011
00305    105*        110 ATOT=ATOT + AREA(KSS)                                                             001013
00307    106*            HTOT = 0.5 * ATOT                                                                 001017
00310    107*            DO 111 KSS=1,KSECT                                                                001025
00313    108*            HNA=HNA+PP(3+3*KSS,L,5)                                                           001025
00314    109*            APART=APART+AREA(KSS)                                                             001030
00315    110*            IF(APART .GT. HTOT) GO TO 112                                                     001033
00317    111*        111 CONTINUE                                                                          001042
00317    112*     C      HNA IS THE POSITION FROM THE TOP OF THE NEUTRAL AXIS IN PLASTICITY                001042
00321    113*        112 HNA = HNA-(APART-HTOT)/AREA(KSS)*PP(3+3*KSS,L,5)                                  001042
00321    114*     C      NEUTRAL AXIS IS AVERAGE OF PLASTIC AND ELASTIC NEUTRAL AXES                       001042
00322    115*            SV(6,I)=(HNM/ATOT + HNA)/2.                                                       001056
00323    116*            KDIV=0                                                                            001063
00324    117*            DO 100 KSS=1,KSECT                                                                001105
00327    118*        100 KDIV=KDIV + IFIX(PP(5+3*KSS,L,5) + .1)                                            001105
00331    119*            K=6+6*(KDIV+1)                                                                    001117
00332    120*            DO 3 J=7,K,3                                                                      001123
00335    121*            SV(J,I)=0.0                                                                       001133
00336    122*            SV(J+1,I)=0.0                                                                     001133
00337    123*            SV(J+2,I)=PP(3,L,5)                                                               001134
00340    124*          3 CONTINUE                                                                          001140
00342    125*          4 CONTINUE                                                                          001140
00343    126*            IF(IT(I),NE.18) GO TO 5                                                           001140
00345    127*            J=ID(I)                                                                           001142
00346    128*            SV(1,I)=SIN(.01745329*PP(8,J,18))                                                 001146
00347    129*            SV(2,I)=COS(.01745329*PP(8,J,18))                                                 001156
00350    130*          5 CONTINUE                                                                          001177
00352    131*            J=0                                                                               001177
00353    132*            DO 6 I=1,IE                                                                       001200
00356    133*            IF(IT(I),LT.6.OR.IT(I),GT,10) GO TO 6                                             001217
00360    134*            J=J+1                                                                             001235
00361    135*            MP(1,J)=IA(I,1)                                                                   001240
00362    136*            MP(2,J)=IA(I,2)                                                                   001244
00363    137*            MP(3,J)=I                                                                         001246
00364    138*            MP(4,J)=0                                                                         001250
00365    139*            IF(IT(I),EQ.6) SV(1,I)=0.0                                                        001251
00367    140*            IF(IT(I),NE.7) GO TO 18                                                           001255
00371    141*            K=IA(I,1)                                                                         001260
00372    142*            L=IA(I,2)                                                                         001262
00373    143*            M=IP(I,1,1)                                                                       001264
00374    144*            N=IP(I,2,1)                                                                       001266
00375    145*            SK=SIN(.01745329*AC(K,M))                                                         001270
00376    146*            CK=COS(.01745329*AC(K,M))                                                         001304
00377    147*            SV(1,I)=0.0                                                                       001310
00400    148*            SV(2,I)=0.0                                                                       001311
00401    149*            SV(3,I)=XC(K,M)*SK-YC(K,M)*CK                                                     001312
```

```
00402   150*          SV(4,I)=XC(L,N)*SIN(.017453293*AC(L,N))-YC(L,N)*COS(.017453293*AC(          001323
00402   151*         1L,N))                                                                     001323
00403   152*          SV(5,I)=XC(K,M)*CK+YC(K,M)*SK                                             001353
00404   153*        18 CONTINUE                                                                 001364
00405   154*          IF(IT(I).EQ.8) SV(1,I)=0.0                                                001364
00407   155*          IF(IT(I).NE.9) GO TO 40                                                   001367
00411   156*          SV(1,I)=0.0                                                               001372
00412   157*          SV(2,I)=1.0                                                               001373
00413   158*          SV(3,I)=1.0                                                               001375
00414   159*          SV(4,I)=1.0                                                               001376
00415   160*          SV(5,1)=1.0                                                               001377
00416   161*          K=IA(I,1)                                                                 001400
00417   162*          L=IA(I,2)                                                                 001402
00420   163*          M=IP(I,1,1)                                                               001404
00421   164*          N=IP(I,2,1)                                                               001406
00422   165*          SV(6,I)=SIN(.017453293*AC(K,M))                                           001410
00423   166*          SV(7,1)=COS(.017453293*AC(K,M))                                           001422
00424   167*          SV(8,I)=SIN(.017453293*AC(L,N))                                           001426
00425   168*          SV(9,1)=COS(.017453293*AC(L,N))                                           001441
00426   169*        40 CONTINUE                                                                 001446
00427   170*          IF(IT(I).NE.10) GO TO 6                                                   001446
00431   171*          SV(1,I)=XC(K,M)*SIN(.017453293*AC(K,M))-YC(K,M)*COS(.017453293            001450
00431   172*         1*AC(K,M))                                                                 001450
00432   173*          SV(2,1)=XC(L,N)*SIN(.017453293*AC(L,N))-YC(L,N)*COS(.017453293            001501
00432   174*         1*AC(L,N))                                                                 001501
00433   175*          SV(3,I)=XC(K,M)*COS(.017453293*AC(K,M))+YC(K,M)*SIN(.017453293            001534
00433   176*         1*AC(K,M))                                                                 001534
00434   177*          SV(4,1)=XC(L,N)*COS(.017453293*AC(L,N))+YC(L,N)*SIN(.017453293            001544
00434   178*         1*AC(L,N))                                                                 001544
00435   179*         6 CONTINUE                                                                 001563
00437   180*          I=0                                                                       001563
00440   181*          NP=0                                                                      001564
00441   182*          NG=0                                                                      001565
00442   183*          IF(J.EQ.0) GO TO 29                                                       001566
00444   184*          DO 32 I=1,IN                                                              001576
00447   185*          MG(I,1)=0                                                                 001576
00450   186*          MG(1,2)=0                                                                 001576
00451   187*        32 CONTINUE                                                                 001600
00453   188*          I=1                                                                       001600
00454   189*          K=0                                                                       001602
00455   190*          L=1                                                                       001603
00456   191*         7 CONTINUE                                                                 001605
00457   192*          MP(4,L)=1                                                                 001605
00460   193*          MG1=MP(1,L)                                                               001611
00461   194*          MG2=MP(2,L)                                                               001613
00462   195*          MG(MG1,1)=I                                                               001615
00463   196*          MG(MG2,1)=1                                                               001620
00464   197*          K=K+1                                                                     001623
00465   198*          IF(K.EQ.J) GO TO 12                                                       001626
00467   199*          M=L+1                                                                     001630
00470   200*         8 CONTINUE                                                                 001634
00471   201*          IF(M.LE.J) GO TO 10                                                       001634
00473   202*          I=I+1                                                                     001637
00474   203*          L=1                                                                       001642
00475   204*         9 CONTINUE                                                                 001644
00476   205*          IF(MP(4,L).EQ.0) GO TO 7                                                  001644
00500   206*          L=L+1                                                                     001650
```

```
00501        GO TO 9                                                              001653
00502     10 IF(MP(4,M).NE.0) GO TO 11                                            001655
00504        IF(MP(1,L).NE.MP(1,M).AND.MP(1,L).NE.MP(2,M).AND.MP(2,L).NE          001661
00504       C.MP(1,M).AND.MP(2,L).NE.MP(2,M)) GO TO 11                            001661
00506        MP(4,M)=I                                                            001714
00507        MG1=MP(1,M)                                                          001716
00510        MG2=MP(2,M)                                                          001720
00511        MG(MG1,1)=I                                                          001722
00512        MG(MG2,1)=I                                                          001725
00513        K=K+1                                                                001730
00514        IF(K.EQ.J) GO TO 12                                                  001733
00516     11 CONTINUE                                                             001736
00517        M=M+1                                                                001736
00520        GO TO 8                                                              001740
00521     12 CONTINUE                                                             001742
00522        NG=I                                                                 001743
00523        NP=J                                                                 001746
00524        DO 28 I=1,NG                                                         001750
00527        K=1                                                                  001760
00530        DO 28 J=1,IN                                                         001766
00533        IF(MG(J,1).NE.I) GO TO 28                                            001766
00535        MG(J,2)=K                                                            001770
00536        NMG(I)=K                                                             001772
00537        K=K+1                                                                001773
00540     28 CONTINUE                                                             002010
00543        DO 33 K=1,IN                                                         002010
00546        IF(MG(K,1).EQ.0) GO TO 33                                            002010
00550        J=MG(K,1)                                                            002012
00551        I=MG(K,2)                                                            002014
00552        MI(I,J)=K                                                            002016
00553     33 CONTINUE                                                             002033
00555        DO 35 J=1,NG                                                         002033
00560        INMG=NMG(J)                                                          002047
00561        DO 35 I=1,INMG                                                       002051
00564        MK(I,J)=0                                                            002063
00565        DO 34 K=1,NP                                                         002067
00570        IF(MI(I,J).NE.MP(1,K)) GO TO 34                                      002067
00572        MK(I,J)=MK(I,J)+1                                                    002071
00573        L=MK(I,J)                                                            002074
00574        MJ(I,J,L)=MP(3,K)                                                    002100
00575        M=0                                                                  002102
00576     36 M=M+1                                                                002104
00577        IF(MP(2,K).NE.MI(M,J)) GO TO 36                                      002115
00601        ML(I,J,L)=M                                                          002120
00602     34 CONTINUE                                                             002134
00604     35 CONTINUE                                                             002134
00607     29 CONTINUE                                                             002134
00610        DO 13 I=1,IE                                                         002134
00613        IF(IT(I).NE.11) GO TO 13                                             002147
00615        J=ID(I)                                                              002151
00616        K=IA(I,1)                                                            002160
00617        L=IA(I,2)                                                            002162
00620        M=IP(I,1,1)                                                          002164
00621        N=IP(I,2,1)                                                          002200
00622        SV(1,I)=XC(K,M)+PP(7,J,11)*COS(AC(K,M)*.017453293)                   002215
00623        SV(2,I)=YC(K,M)+PP(7,J,11)*SIN(AC(K,M)*.017453293)                   002230
00624        SV(3,I)=XC(L,N)+PP(15,J,11)*COS(AC(L,N)*.017453293)                  002242
```

```
00625    264*          SV(4,I)=YC(L,N)+PP(15,J,11)*SIN(AC(L,N)*,0174533293)                                    002255
00626    265*          SV(5,I)=PP(19,J,11)                                                                     002267
00627    266*          SV(6,I)=PP(1,J,11)*PP(9,J,11)/(PP(1,J,11)+PP(9,J,11))                                   002271
00630    267*          F1=PP(1,J,11)*PP(2,J,11)                                                                002277
00631    268*          F2=PP(9,J,11)*PP(10,J,11)                                                               002302
00632    269*          IF(F1,LT,F2) GO TO 14                                                                   002305
00634    270*          SV(7,I)=-PP(2,J,11)-F2/PP(1,J,11)                                                       002310
00635    271*          SV(8,I)=PP(1,J,11)*PP(11,J,11)/(PP(1,J,11)+PP(11,J,11))                                 002314
00636    272*          SV(9,I)=-PP(2,J,11)-PP(10,J,11)-(F1-F2)/PP(11,J,11)                                     002323
00637    273*          GO TO 15                                                                                002331
00640    274*       14 CONTINUE                                                                                002333
00641    275*          SV(7,I)=-PP(10,J,11)-F1/PP(9,J,11)                                                      002335
00642    276*          SV(8,I)=PP(3,J,11)*PP(9,J,11)/(PP(3,J,11)+PP(9,J,11))                                   002341
00643    277*          SV(9,I)=-PP(2,J,11)-PP(10,J,11)-(F2-F1)/PP(3,J,11)                                      002347
00644    278*       15 CONTINUE                                                                                002357
00645    279*          SV(10,I)=PP(3,J,11)*PP(11,J,11)/(PP(3,J,11)+PP(11,J,11))                                002361
00646    280*          DK=(PP(4,J,11)+PP(2,J,11)*(PP(3,J,11)-PP(1,J,11)))/PP(3,J,11)                           002367
00647    281*          DL=(PP(12,J,11)+PP(10,J,11)*(PP(11,J,11)-PP(8,J,11)))/PP(11,J,11)                       002375
00650    282*          SK=PP(4,J,11)/DK                                                                        002403
00651    283*          SL=PP(12,J,11)/DL                                                                       002406
00652    284*          F1=PP(4,J,11)                                                                           002411
00653    285*          F2=PP(12,J,11)                                                                          002413
00654    286*          IF(F1,LT,F2) GO TO 16                                                                   002415
00656    287*          SV(11,I)=-DK-DL-(F1-F2)/PP(11,J,11)                                                     002420
00657    288*          SV(12,I)=SK*PP(11,J,11)/(SK+PP(11,J,11))                                                002425
00660    289*          SV(13,I)=-DL-F2/SK                                                                      002432
00661    290*          GO TO 17                                                                                002436
00662    291*       16 CONTINUE                                                                                002440
00663    292*          SV(11,I)=-DK-DL-(F2-F1)/PP(3,J,11)                                                      002442
00664    293*          SV(12,I)=PP(3,J,11)*SL/(PP(3,J,11)+SL)                                                  002451
00665    294*          SV(13,I)=-DK-F1/SL                                                                      002457
00666    295*       17 CONTINUE                                                                                002464
00667    296*          SV(14,I)=SK*SL/(SK+SL)                                                                  002466
00670    297*          SV(15,I)=(PP(8,J,11)+PP(16,J,11))/2,                                                    002474
00671    298*          SV(16,I)=0,0                                                                            002500
00672    299*          SV(17,I)=0,0                                                                            002501
00673    300*          SV(18,I)=0,0                                                                            002502
00674    301*          SV(19,I)=0,0                                                                            002503
00675    302*       13 CONTINUE                                                                                002526
00677    303*          DO 23 I=1,IE                                                                            002526
00702    304*          IF(IT(I),NE,12) GO TO 23                                                                002526
00704    305*          J=ID(I)                                                                                 002530
00705    306*          K=IA(I,1)                                                                               002537
00706    307*          L=IA(I,2)                                                                               002541
00707    308*          M=IP(I,1,1)                                                                             002543
00710    309*          N=IP(I,2,1)                                                                             002560
00711    310*          PP(19,J,12) =0,0                                                                        002575
00712    311*          SV(1,I)=XC(K,M)+PP(7,J,12)*COS(AC(K,M)*,0174533293)                                     002576
00713    312*          SV(2,I)=YC(K,M)+PP(7,J,12)*SIN(AC(K,M)*,0174533293)                                     002612
00714    313*          SV(3,I)=XC(L,N)+PP(15,J,12)*COS(AC(L,N)*,0174533293)                                    002625
00715    314*          SV(4,I)=YC(L,N)+PP(15,J,12)*SIN(AC(L,N)*,0174533293)                                    002640
00716    315*          SV(5,I)=PP(19,J,12)                                                                     002653
00717    316*          SV(6,I)=PP(1,J,12)*PP(9,J,12)/(PP(1,J,12)+PP(9,J,12))                                   002655
00720    317*          F1=PP(1,J,12)*PP(2,J,12)                                                                002663
00721    318*          F2=PP(9,J,12)*PP(10,J,12)                                                               002666
00722    319*          IF(F1,LT,F2) GO TO 24                                                                   002671
00724    320*          SV(7,I)=-PP(2,J,12)-F2/PP(1,J,12)                                                       002674
```

```
00725    321*          SV(8,I)=PP(1,J,12)*PP(11,J,12)/(PP(1,J,12)+PP(11,J,12))                                                  002700
00726    322*          SV(9,I)=-PP(2,J,12)-PP(10,J,12)-(F1-F2)/PP(11,J,12)                                                       002707
00727    323*          GO TO 25                                                                                                  002715
00730    324*        24 CONTINUE                                                                                                 002717
00731    325*          SV(7,I)=-PP(10,J,12)-F1/PP(9,J,12)                                                                        002721
00732    326*          SV(8,I)=PP(3,J,12)*PP(9,J,12)/(PP(3,J,12)+PP(9,J,12))                                                     002725
00733    327*          SV(9,I)=-PP(2,J,12)-PP(10,J,12)-(F2-F1)/PP(3,J,12)                                                        002733
00734    328*        25 CONTINUE                                                                                                 002743
00735    329*          SV(10,I)=PP(3,J,12)*PP(11,J,12)/(PP(3,J,12)+PP(11,J,12))                                                  002745
00736    330*          DK=(PP(4,J,12)+PP(2,J,12)*(PP(3,J,12)-PP(1,J,12)))/PP(3,J,12)                                             002753
00737    331*          DL=(PP(12,J,12)+PP(10,J,12)*(PP(11,J,12)-PP(8,J,12)))/PP(11,J,12)                                         002761
00740    332*          SK=PP(4,J,12)/DK                                                                                          002767
00741    333*          SL=PP(12,J,12)/DL                                                                                         002772
00742    334*          F1=PP(4,J,12)                                                                                             002775
00743    335*          F2=PP(12,J,12)                                                                                            002777
00744    336*          IF(F1.LT.F2) GO TO 26                                                                                     003001
00746    337*          SV(11,I)=-DK-DL-(F1-F2)/PP(11,J,12)                                                                       003004
00747    338*          SV(12,I)=SK*PP(11,J,12)/(SK+PP(11,J,12))                                                                  003011
00750    339*          SV(13,I)=-DL-F2/SK                                                                                        003016
00751    340*          GO TO 27                                                                                                  003022
00752    341*        26 CONTINUE                                                                                                 003024
00753    342*          SV(11,I)=-DK-DL-(F2-F1)/PP(3,J,12)                                                                        003026
00754    343*          SV(12,I)=PP(3,J,12)*SL/(PP(3,J,12)+SL)                                                                    003035
00755    344*          SV(13,I)=-DK-F1/SL                                                                                        003043
00756    345*        27 CONTINUE                                                                                                 003050
00757    346*          SV(14,I)=SK*SL/(SK+SL)                                                                                    003052
00760    347*          SV(15,I)=(PP(8,J,12)+PP(16,J,12))/2.                                                                      003060
00761    348*        23 CONTINUE                                                                                                 003106
00763    349*          DO 37 I=1,IE                                                                                              003106
00766    350*          IF(IT(I).NE.13) GO TO 37                                                                                  003106
00770    351*          J=ID(I)                                                                                                   003110
00771    352*          K=IA(I,1)                                                                                                 003117
00772    353*          L=IA(I,2)                                                                                                 003131
00773    354*          M=IP(I,1,1)                                                                                               003144
00774    355*          N=IP(I,2,1)                                                                                               003156
00775    356*          SV(1,I)=XC(K,M)                                                                                           003170
00776    357*          SV(2,I)=YC(K,M)                                                                                           003172
00777    358*          SV(3,I)=XC(L,N)                                                                                           003174
01000    359*          SV(4,I)=YC(L,N)                                                                                           003200
01001    360*          SV(5,I)=PP(1,J,13)                                                                                        003202
01002    361*          SV(6,I)=-PP(2,J,13)                                                                                       003206
01003    362*          SV(7,I)=PP(3,J,13)                                                                                        003210
01004    363*          SV(8,I)=-(PP(4,J,13)+PP(2,J,13)*(SV(7,I)-SV(5,I)))/SV(7,I)                                                003212
01005    364*          SV(9,I)=-PP(4,J,13)/SV(8,I)                                                                               003217
01006    365*          SV(10,I)=SV(8,I)-(PP(5,J,13)-PP(4,J,13))/SV(7,I)-PP(6,J,13)                                               003222
01007    366*          SV(11,I)=SV(5,I)                                                                                          003230
01010    367*          SV(12,I)=SV(10,I)-PP(2,J,13)                                                                              003232
01011    368*          SV(13,I)=SV(7,I)                                                                                          003235
01012    369*          SV(14,I)=SV(10,I)+SV(8,I)                                                                                 003237
01013    370*          SV(15,I)=SV(9,I)                                                                                          003242
01014    371*          SV(16,I)=-PP(5,J,13)                                                                                      003244
01015    372*          SV(17,I)=-PP(7,J,13)                                                                                      003246
01016    373*          DX = XP(K)+XC(K,M)*COS(AP(K))-YC(K,M)*SIN(AP(K))                                                          003250
01016    374*         1 - ( XP(L)+XC(L,N)*COS(AP(L))-YC(L,N)*SIN(AP(L)) )                                                        003250
01017    375*          DY = YP(K)+YC(K,M)*COS(AP(K))+XC(K,M)*SIN(AP(K))                                                          003320
01017    376*         I - ( YP(L)+YC(L,N)*COS(AP(L))+XC(L,N)*SIN(AP(L)) )                                                        003320
01020    377*          SV(18,I)=SQRT(DX*DX+DY*DY)                                                                                003367
```

```
01021    378*          SV(19,I)=SV(8,I)-(PP(5,J,13)-PP(4,J,13))/SV(7,I)                                    003400
01022    379*          SV(20,I)=SV(14,I)-(PP(7,J,13)-PP(4,J,13))/SV(7,I)                                   003406
01023    380*       37 CONTINUE                                                                            003435
01025    381*          DO 38 I=1,IE                                                                        003435
01030    382*          IF(IT(I).NE.14) GO TO 38                                                            003435
01032    383*          J=ID(I)                                                                             003437
01033    384*          K=IA(I,1)                                                                           003441
01034    385*          L=IA(I,2)                                                                           003443
01035    386*          M=IP(I,1,1)                                                                         003445
01036    387*          N=IP(I,2,1)                                                                         003447
01037    388*          SV(1,I)=XC(K,M)+PP(3,J,14)*COS(AC(K,M)*.0174533293)                                 003451
01040    389*          SV(2,I)=YC(K,M)+PP(3,J,14)*SIN(AC(K,M)*.0174533293)                                 003475
01041    390*          SV(3,I)=XC(L,N)+PP( 8,J,14)*COS(AC(L,N)*.0174533293)                                003505
01042    391*          SV(4,I)=YC(L,N)+PP( 8,J,14)*SIN(AC(L,N)*.0174533293)                                003524
01043    392*          SV(5,I)=PP(13,J,14)                                                                 003534
01044    393*          SV(6,I)=(PP(4,J,14)+PP(9,J,14))*0.5                                                 003536
01045    394*          SV(7,I)=PP(1,J,14)*PP(6,J,14)/(PP(1,J,14)+PP(6,J,14))                               003542
01046    395*          SV(8,I)=PP(2,J,14)*PP(7,J,14)/(PP(2,J,14)+PP(7,J,14))                               003550
01047    396*          SV(9,I)=PP(12,J,14)                                                                 003556
01050    397*          CT1=PP(2,J,14)*PP(14,J,14)**2                                                       003560
01051    398*          CT2=PP(7,J,14)*PP(15,J,14)**2                                                       003564
01052    399*          SV(10,I)=CT1*CT2/(CT1+CT2)                                                          003570
01053    400*          SV(11,I)=0.0                                                                        003574
01054    401*       38 CONTINUE                                                                            003620
01056    402*          DO 30 I=1,IE                                                                        003620
01061    403*          IF(IT(I) .NE. 15) GO TO 30                                                          003625
01063    404*          J=ID(I)                                                                             003630
01064    405*          K=IA(I,1)                                                                           003634
01065    406*          L=IA(I,2)                                                                           003636
01066    407*          M=IP(I,1,1)                                                                         003640
01067    408*          N=IP(I,2,1)                                                                         003653
01070    409*          SV(1,I)=XC(K,M)+                                                                    003670
01070    410*         1                PP(28,J,15)*COS(AC(K,M)*.0174533293)                               003670
01071    411*          SV(2,I)=YC(K,M)+                                                                    003705
01071    412*         1                PP(28,J,15)*SIN(AC(K,M)*.0174533293)                               003705
01072    413*          SV(3,I)=XC(L,N)+PP(29,J,15)*COS(AC(L,N)*.0174533293)                                003720
01073    414*          SV(4,I)=YC(L,N)+PP(29,J,15)*SIN(AC(L,N)*.0174533293)                                003733
01074    415*          E11 = PP(1,J,15)                                                                    003746
01075    416*          E12 = PP(2,J,15)                                                                    003750
01076    417*          D11 = PP(3,J,15)                                                                    003752
01077    418*          D12 = PP(4,J,15)                                                                    003754
01100    419*          E21 = PP(14,J,15)                                                                   003756
01101    420*          E22 = PP(15,J,15)                                                                   003760
01102    421*          D21 = PP(16,J,15)                                                                   003762
01103    422*          D22 = PP(17,J,15)                                                                   003764
01104    423*          CALL COMOD(E11,E12,D11,D12,E21,E22,D21,D22,E1,E2,E3,D1,D2,D3,FM,FN                   003766
01104    424*         1  ,F0)                                                                              003766
01105    425*          SV(6,1) = E1                                                                        004011
01106    426*          SV(7,I) = E2                                                                        004013
01107    427*          SV(8,I) = E3                                                                        004015
01110    428*          SV(10,I) = D1                                                                       004017
01111    429*          SV(11,I) = D2                                                                       004021
01112    430*          SV(12,I) = D3                                                                       004023
01113    431*          SV(20,I) = FM                                                                       004025
01114    432*          SV(21,I) = FN                                                                       004027
01115    433*          SV(22,I) = F0                                                                       004031
01116    434*          E11 = PP(5,J,15)                                                                    004033
```

```
01117    435*           E12 = PP(6,J,15)                                                    004035
01120    436*           D11 = PP(7,J,15)                                                    004037
01121    437*           D12 = PP(8,J,15)                                                    004041
01122    438*           E21 = PP(18,J,15)                                                   004043
01123    439*           E22 = PP(19,J,15)                                                   004045
01124    440*           D21 = PP(20,J,15)                                                   004047
01125    441*           D22 = PP(21,J,15)                                                   004051
01126    442*           CALL COMOD(E11,E12,D11,D12,E21,E22,D21,D22,E1,E2,E3,D1,D2,D3,FM,FN   004053
01126    443*          1  ,FO)                                                              004053
01127    444*           SV(30,I) = E1                                                       004076
01130    445*           SV(31,I) = E2                                                       004100
01131    446*           SV(32,I) = E3                                                       004102
01132    447*           SV(27,I) = D1                                                       004104
01133    448*           SV(28,I) = D2                                                       004106
01134    449*           SV(29,I) = D3                                                       004110
01135    450*           SV(37,I) = FM                                                       004112
01136    451*           SV(38,I) = FN                                                       004114
01137    452*           SV(39,I) = FO                                                       004116
01140    453*           E11 = PP(9,J,15)                                                    004120
01141    454*           E12 = PP(10,J,15)                                                   004122
01142    455*           D11 = PP(11,J,15)                                                   004124
01143    456*           D12 = PP(12,J,15)                                                   004126
01144    457*           E21 = PP(22,J,15)                                                   004130
01145    458*           E22 = PP(23,J,15)                                                   004132
01146    459*           D21 = PP(24,J,15)                                                   004134
01147    460*           D22 = PP(25,J,15)                                                   004136
01150    461*           CALL COMOD(E11,E12,D11,D12,E21,E22,D21,D22,E1,E2,E3,D1,D2,D3,FM,FN   004140
01150    462*          1  ,FO)                                                              004140
01151    463*           SV(60,I) = E1                                                       004163
01152    464*           SV(61,I) = E2                                                       004165
01153    465*           SV(62,I) = E3                                                       004167
01154    466*           SV(44,I) = D1                                                       004171
01155    467*           SV(45,I) = D2                                                       004173
01156    468*           SV(46,I) = D3                                                       004175
01157    469*           SV(54,I) = FM                                                       004177
01160    470*           SV(55,I) = FN                                                       004201
01161    471*           SV(56,I) = FO                                                       004203
01162    472*           DO 31 IV=1,3                                                        004211
01165    473*           SV(12+IV,I)=-SV(9+IV,I)                                             004211
01166    474*           SV(65+IV,I)=-SV(26+ IV,I)                                           004212
01167    475*        31 SV(46+IV,I)=-SV(43+IV,I)                                            004214
01171    476*           SV(19,I)=SV(12,I)                                                   004217
01172    477*           SV(36,I)=SV(29,I)                                                   004221
01173    478*           SV(53,I)=SV(46,I)                                                   004223
01174    479*           SV(24,I)=0.0                                                        004225
01175    480*           SV(17,I)=0.0                                                        004226
01176    481*           SV(25,I)=SV(20,I)                                                   004227
01177    482*           SV(34,I)=0.0                                                        004231
01200    483*           SV(41,I)=0.0                                                        004232
01201    484*           SV(42,I)=SV(37,I)                                                   004233
01202    485*           SV(51,I)=0.0                                                        004235
01203    486*           SV(58,I)=0.0                                                        004236
01204    487*           SV(59,I)=SV(54,I)                                                   004237
01205    488*           SV(64,I)=0.5*(PP(13,J,15)+PP(26,J,15))                              004241
01206    489*           SV(65,I)=PP(27,J,15)                                                004246
01207    490*        30 CONTINUE                                                            004262
01207    491*     C     WRITE (6,33) (SV(IV,I),IV=1,65)                                     004262
```

```
01207     492*      C  33 FORMAT (1 SV1/(12E10.4))                                  004262
01211     493*           RETURN                                                     004262
01212     494*           END                                                        004340

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR.IS .INTG
FOR 8E38-06/02/77-14:32:37 (.0)

   SUBROUTINE INTG       ENTRY POINT 000033

   STORAGE USED: CODE(1) 000036: DATA(0) 000004: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   TIME   000012
    0004   WRITE  000305

   EXTERNAL REFERENCES (BLOCK, NAME)

    0005   EULR
    0006   NERR2$
    0007   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000011 1L        0001   000016 2L        0001   000021 3L        0001   000024 4L        0003 I 000000 IM
    0000   000000 INJP$     0003   000003 IS        0003   000011 ITS       0004   000000 IW        0004   000003 IWW
    0004   000004 IW1       0004   000137 JDR       0004   000005 JI        0004   000043 JJ        0004   000101 JTY
    0004   000263 KDR       0004   000175 KF        0004   000217 KI        0004   000241 KJ        0004   000002 NF
    0004   000001 NM        0003   000001 TD        0003   000002 TF        0003   000010 TT

00101      1*            COMPILER(XM=1),(ADR=IND)                                          000000
00103      2*            SUBROUTINE INTG                                                   000000
00105      3*            INCLUDE PARM                                                      000000
00110      4*            INCLUDE TIME                                                      000000
00112      5*            INCLUDE WRITE                                                     000000
00112      6*      C                                                                       000000
00112      7*      C  IM = 1      EULER INTEGRATION                                        000000
00112      8*      C                                                                       000000
00114      9*            GO TO (1,2,3,4),IM                                                000000
00115     10*          1 CONTINUE                                                          000011
00116     11*            CALL EULR                                                         000011
00117     12*            RETURN                                                            000012
00120     13*          2 CONTINUE                                                          000016
00121     14*            RETURN                                                            000016
00122     15*          3 CONTINUE                                                          000021
00123     16*            RETURN                                                            000021
00124     17*          4 CONTINUE                                                          000024
00125     18*            RETURN                                                            000024
00126     19*            END                                                               000035

       END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
#FOR:IS .FNSH
FOR SE38-06/02/77-14:32:43 (:0)

   SUBROUTINE FNSH       ENTRY POINT 000134

   STORAGE USED: CODE(1) 000141: DATA(0) 000007: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    TIME    000012
    0004    STAT    000660
    0005    MASS    001101
    0006    ELEM    013117
    0007    CONN    002720
    0010    GRAV    000222
    0011    WRITE   000305
    0012    FORC    063140
    0013    SAVE    030470
    0014    PAIR    000553

   EXTERNAL REFERENCES (BLOCK, NAME)

    0015    NWBUS
    0016    NIO3S
    0017    NIO1S
    0020    NIO2S
    0021    NERR3S

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0004 R 000000 A         0004   000600 AA        0007   001760 AC        0004   000140 AP        0004   000360 AV
    0005 R 000361 BB        0006 R 004731 CC        0007 R 000060 DD        0010 R 000000 E         0012   062720 F
    0005   000521 FA        0012   062500 FE        0005   000601 FF        0010   000002 FG        0012   000000 FI
    0012 R 000000 G         0010   000000 GA        0010   000001 GG        0013 R 000000 H         0006   000551 IA
    0007   000000 IC        0006   000265 ID        0006   000000 IE        0005   000061 IF        0005   000001 II
    0003 I 000000 IM        0005   000000 IN        0000   000002 INJPS     0006   001605 IP        0003 I 000003 IS
    0006   000001 IT        0003 I 000011 ITS       0011   000000 IW        0011   000003 IWW       0011   000004 IW1
    0005 I 000000 JB        0006 I 000000 JC        0007 I 000000 JD        0011   000137 JDH       0011 I 000000 JF
    0011   000005 JI        0011   000043 JJ        0014 I 000000 JPQ       0011   000101 JTY       0000 I 000000 K
    0011   000263 KDR       0011   000175 KE        0011   000217 KI        0011   000241 KJ        0000 I 000001 L
    0014   000026 ML        0014   000173 MI        0014   000217 MJ        0014   000337 MK        0014   000363 ML
    0014   000002 MP        0011   000002 NF        0014   000000 NG        0011   000001 NM        0014   000166 NMG
    0014   000001 NP        0006   004731 PP        0005   000441 RI        0013   000000 SV        0003 R 000001 TD
    0003 R 000002 TF        0003 R 000010 TT        0005   000361 WT        0004   000440 XA        0007   000060 XC
    0004   000000 XP        0004   000220 XV        0004   000520 YA        0007   001020 YC        0004   000060 YP
    0004   000300 YV

00101       1*              COMPILER(XM=1),(ADR=IND)                                                 000000
00103       2*              SUBROUTINE FNSH(JUNIT)                                                   000000
00105       3*              INCLUDE PARM                                                             000000
```

```
00110      4*          INCLUDE TIME                                                                    000000
00112      5*          INCLUDE STAT                                                                    000000
00114      6*          INCLUDE MASS                                                                    000000
00116      7*          INCLUDE ELEM                                                                    000000
00120      8*          INCLUDE CONN                                                                    000000
00122      9*          INCLUDE GRAV                                                                    000000
00124     10*          INCLUDE WRITE                                                                   000000
00126     11*          INCLUDE FORC                                                                    000000
00130     12*          INCLUDE SAVE                                                                    000000
00132     13*          INCLUDE PAIR                                                                    000000
00134     14*          DIMENSION A(KSTAT),JB(KMASS1),BB(KMASS2),JC(KELEM1),CC(KELEM2),                 000000
00134     15*         1     JD(KCONN1),DD(KCONN2),F(KGRAV),JF(KWRIT),G(KFORC),H(KSAVE),               000000
00134     16*         2     JPQ(KPAIR)                                                                000000
00135     17*          EQUIVALENCE (A(1),XP(1)),          (JH(1),IN),                                  000000
00135     18*         1            (BB(1),WT(1)),      (JC(1),IE),                                    000000
00135     19*         2            (CC(1),PP(1,1,1)),  (JD(1),IC(1)),                                 000000
00135     20*         3            (DD(1),XC(1,1)),    (E(1),GA),                                     000000
00135     21*         4            (JF(1),IW),         (G(1),FI(1,1,1)),                              000000
00135     22*         5            (H(1),SV(1,1)),     (JPQ(1),NG)                                    000000
00136     23*          WRITE(JUNIT) IM,TD,TF,(IS(K),K=1,5),TT,ITS                                      000000
00146     24*          WRITE(JUNIT) (A(K),K=1,KSTAT)                                                   000014
00151     25*          WRITE(JUNIT) (JB(K),K=1,KMASS1),(BB(L),L=1,KMASS2)                              000023
00155     26*          WRITE(JUNIT) (JC(K),K=1,KELEM1),(CC(L),L=1,KELEM2)                              000035
00161     27*          WRITE(JUNIT) (JD(K),K=1,KCONN1),(DD(L),L=1,KCONN2)                              000047
00165     28*          WRITE(JUNIT) (E(K),K=1,KGRAV)                                                   000061
00170     29*          WRITE(JUNIT) (JF(K),K=1,KWRIT)                                                  000070
00173     30*          WRITE(JUNIT) (G(K),K=1,KFORC)                                                   000077
00176     31*          WRITE(JUNIT) (H(K),K=1,KSAVE)                                                   000106
00201     32*          WRITE(JUNIT) (JPQ(K),K=1,KPAIR)                                                 000115
00204     33*          RETURN                                                                          000124
00205     34*          END                                                                             000140

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS .HEAD
FOR 8E3B-06/02/77-14:32:49 (.0)

   SUBROUTINE HEAD        ENTRY POINT 000610

   STORAGE USED: CODE(1) 000630; DATA(0) 000421; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   WRITE  000305

   EXTERNAL REFERENCES (BLOCK, NAME)

    0004   NWDUS
    0005   NIO2$
    0006   NERR2$
    0007   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000366 1L       0001    000320 10L      0001    000564 100L     0001    000071 160G     0001    000107 165G
    0001    000442 2L       0001    000154 204G     0001    000166 210G     0001    000232 226G     0001    000244 232G
    0001    000377 274G     0001    000506 3L       0001    000414 301G     0001    000450 317G     0001    000462 323G
    0001    000514 340G     0001    000526 344G     0001    000144 51L      0001    000222 52L      0001    000351 88L
    0001    000325 89L      0000    000352 9F       0001    000043 99L      0000    000355 999F     0000 R 000001 AWORD
    0000 R 000004 BWORD     0000 I 000000 CON       0000 R 000331 CWORD     0000 R 000007 DWORD     0000 R 000221 FMC
    0000 R 000261 FMCC      0000 R 000155 FMD       0000 R 000215 FMDD      0000 R 000045 FMI       0000 R 000105 FMII
    0000 R 000111 FMJ       0000 R 000151 FMJJ      0000 R 000265 FMM       0000 R 000325 FMMM      0000 R 000012 FMT
    0000 R 000042 FMTT      0000 I 000340 I         0000    000361 INJP$    0003    000000 IW       0003    000003 IWW
    0003    000004 IW1      0000 I 000336 JA        0000 I 000337 JH        0000 I 000346 JC        0000 I 000347 JD
    0003 I 000137 JDW       0000 I 000350 JE        0000 I 000332 JF        0000 I 000351 JG        0003 I 000005 JI
    0003 I 000043 JJ        0000 I 000344 JM        0003 I 000101 JTY       0000 I 000335 K         0003 I 000263 KDW
    0003 I 000175 KE        0003 I 000217 KI        0003 I 000241 KJ        0000 I 000341 KK        0000 I 000342 MA
    0003 I 000002 NF        0003 I 000001 NM        0000 I 000333 N1        0000 I 000334 N2        0000 I 000343 N3
    0000 I 000345 N4


00101       1*           COMPILER(XM=1),(ADR=IND)                                                       000027
00103       2*           SUBROUTINE HEAD                                                                000027
00105       3*           INCLUDE PARM                                                                   000027
00110       4*           INCLUDE WRITE                                                                  000027
00112       5*           INTEGER CON                                                                    000027
00113       6*           DIMENSION AWORD(3),BWORD(3),DWORD(3)                                           000027
00114       7*           DIMENSION FMT(3,8),FMTT(3),FMI(4,8),FMII(4),FMJ(4,8),FMJJ(4)                   000027
00115       8*           DIMENSION FMD(4,8),FMDD(4),FMC(4,8),FMCC(4),FMM(4,8),FMMM(4)                   000027
00116       9*           DATA AWORD/'X    ','Y    ','ANG '/                                             000027
00120      10*           DATA BWORD/'DISP','VEL ','ACCL'/                                               000027
00122      11*           DATA CWORD/'  TIME'/                                                           000027
00124      12*           DATA DWORD/'FX','FY','MZ'/                                                     000027
00126      13*           DATA FMT/'(1H ,  7X,2A4)                                                       000027
00126      14*          1         '(1H+, 22X,2A4)                                                       000027
```

```
00126         1          '(1H+, 37X,2A4)      ',          000027
00126         1          '(1H+, 52X,2A4)      ',          000027
00126         1          '(1H+, 67X,2A4)      ',          000027
00126         1          '(1H+, 82X,2A4)      ',          000027
00126         1          '(1H+, 97X,2A4)      ',          000027
00126         1          '(1H+,112X,2A4)      '/          000027
00130          DATA FMI/'(1H ,  7X,6HOF PT ,I2)',         000027
00130         2          '(1H+, 22X,6HOF PT ,I2)',        000027
00130         2          '(1H+, 37X,6HOF PT ,I2)',        000027
00130         2          '(1H+, 52X,6HOF PT ,I2)',        000027
00130         2          '(1H+, 67X,6HOF PT ,I2)',        000027
00130         2          '(1H+, 82X,6HOF PT ,I2)',        000027
00130         2          '(1H+, 97X,6HOF PT ,I2)',        000027
00130         2          '(1H+,112X,6HOF PT ,I2)'/        000027
00132          DATA FMJ/'(1H ,  7X,6HON M  ,I2)',         000027
00132         3          '(1H+, 22X,6HON M  ,I2)',        000027
00132         3          '(1H+, 37X,6HON M  ,I2)',        000027
00132         3          '(1H+, 52X,6HON M  ,I2)',        000027
00132         3          '(1H+, 67X,6HON M  ,I2)',        000027
00132         3          '(1H+, 82X,6HON M  ,I2)',        000027
00132         3          '(1H+, 97X,6HON M  ,I2)',        000027
00132         3          '(1H+,112X,6HON M  ,I2)'/        000027
00134          DATA FMC/'(1H ,  7X,6HAT PT ,I2)  ',       000027
00134         4          '(1H+, 22X,6HAT PT ,I2)  ',      000027
00134         4          '(1H+, 37X,6HAT PT ,I2)  ',      000027
00134         4          '(1H+, 52X,6HAT PT ,I2)  ',      000027
00134         4          '(1H+, 67X,6HAT PT ,I2)  ',      000027
00134         4          '(1H+, 82X,6HAT PT ,I2)  ',      000027
00134         4          '(1H+, 97X,6HAT PT ,I2)  ',      000027
00134         4          '(1H+,112X,6HAT PT ,I2)  '/      000027
00136          DATA FMD/'(1H ,  7X,A2,4H ON ,I2) ',       000027
00136         5          '(1H+, 22X,A2,4H ON ,I2) ',      000027
00136         5          '(1H+, 37X,A2,4H ON ,I2) ',      000027
00136         5          '(1H+, 52X,A2,4H ON ,I2) ',      000027
00136         5          '(1H+, 67X,A2,4H ON ,I2) ',      000027
00136         5          '(1H+, 82X,A2,4H ON ,I2) ',      000027
00136         5          '(1H+, 97X,A2,4H ON ,I2) ',      000027
00136         5          '(1H+,112X,A2,4H ON ,I2) '/      000027
00140          DATA FMM/'(1H ,  7X,6HBY M  ,I2)  ',       000027
00140         6          '(1H+, 22X,6HBY M  ,I2)  ',      000027
00140         6          '(1H+, 37X,6HBY M  ,I2)  ',      000027
00140         6          '(1H+, 52X,6HBY M  ,I2)  ',      000027
00140         6          '(1H+, 67X,6HBY M  ,I2)  ',      000027
00140         6          '(1H+, 82X,6HBY M  ,I2)  ',      000027
00140         6          '(1H+, 97X,6HBY M  ,I2)  ',      000027
00140         6          '(1H+,112X,6HBY M  ,I2)  '/      000027
00142       9 FORMAT(1H1,5X,A6//)                         000027
00143     999 FORMAT(//)                                  000027
00144         WRITE(6,9) CWORD                            000027
00147         JF=1                                        000035
00150         IF(NM.EQ.0) GO TO 10                        000037
00152         N1=1                                        000041
00153      99 N2=N1+7                                     000043
00154         JF=1                                        000045
00155         IF(N2.GT.NM) N2=NM                          000047
00157         DO 111 K=N1,N2                              000063
00162         JA=JDR(K)                                   000074
```

```
00163     72*          JB=JTY(K)                                          000077
00164     73*          DO 11 I=1,3                                        000101
00167     74*       11 FMTT(I)=FMT(I,JF)                                  000107
00171     75*          WRITE(6,FMTT) AWORD(JA),BWORD(JB)                  000111
00175     76*      111 JF=JF+1                                            000120
00177     77*          KK=1                                               000126
00200     78*          IF(JF.LT.9.AND.NF.GT.0) GO TO 89                   000130
00202     79*       51 JF=1                                               000144
00203     80*          DO 229 K=N1,N2                                     000145
00206     81*          CON=JJ(K)                                          000157
00207     82*          DO 22 I=1,4                                        000166
00212     83*       22 FMII(I)=FMI(I,JF)                                  000166
00214     84*          WRITE(6,FMII) CON                                  000170
00217     85*      229 JF=JF+1                                            000176
00221     86*          KK=2                                               000203
00222     87*          IF(JF.LT.9.AND.NF.GT.0) GO TO 89                   000205
00224     88*       52 JF=1                                               000222
00225     89*          DO 339 K=N1,N2                                     000223
00230     90*          MA=JI(K)                                           000235
00231     91*          DO 33 I=1,4                                        000244
00234     92*       33 FMJJ(I)=FMJ(I,JF)                                  000244
00236     93*          WRITE(6,FMJJ) MA                                   000246
00241     94*      339 JF=JF+1                                            000254
00243     95*          KK=3                                               000261
00244     96*          IF(JF.LT.9.AND.NF.GT.0) GO TO 89                   000263
00246     97*          N1=N2+1                                            000277
00247     98*          IF(JF.GT.8) WRITE(6,999)                           000302
00252     99*          IF(N1.LE.NM) GO TO 99                              000313
00254    100*       10 IF(NF.EQ.0) GO TO 100                              000320
00256    101*          N3=1                                               000321
00257    102*          GO TO 88                                           000323
00260    103*       89 JM=JF                                              000325
00261    104*          N3=1                                               000326
00262    105*          N4=9-JM                                            000330
00263    106*          IF(N4.GT.NF) N4=NF                                 000332
00265    107*          GO TO(1,2,3),KK                                    000340
00266    108*       88 JM=1                                               000351
00267    109*          KK=10                                              000352
00270    110*          N4=N3+7                                            000354
00271    111*          IF(N4.GT.NF) N4=NF                                 000357
00273    112*        1 DO 222 K=N3,N4                                     000366
00276    113*          JC=KDR(K)                                          000402
00277    114*          JD=KE(K)                                           000405
00300    115*          DO 13 I=1,4                                        000414
00303    116*       13 FMDD(I)=FMD(I,JM)                                  000414
00305    117*          WRITE(6,FMDD) DWORD(JC),JD                         000416
00311    118*      222 JM=JM+1                                            000425
00313    119*          IF(KK.LT.10) GO TO 51                              000433
00315    120*          JM=1                                               000437
00316    121*        2 DO 333 K=N3,N4                                     000442
00321    122*          JE=KJ(K)                                           000453
00322    123*          DO 23 I=1,4                                        000462
00325    124*       23 FMCC(I)=FMC(I,JM)                                  000462
00327    125*          WRITE(6,FMCC) JF                                   000464
00332    126*      333 JM=JM+1                                            000472
00334    127*          IF(KK.LT.10) GO TO 52                              000477
00336    128*          JM=1                                               000503
```

```
00337     129*          3 DO 444 K=N3,N4                                  000506
00342     130*            JG=KI(K)                                        000517
00343     131*            DO 43 I=1,4                                     000526
00346     132*           43 FMMM(I)=FMM(I,JM)                             000526
00350     133*            WRITE(6,FMMM) JG                                000530
00353     134*          444 JM=JM+1                                       000536
00355     135*            N3=N4+1                                         000543
00356     136*            IF(JM.GT.A) WRITE(6,999)                        000546
00361     137*            IF(N3.LE.NF) GO TO 88                           000557
00363     138*          100 CONTINUE                                      000564
00364     139*            RETURN                                          000564
00365     140*            END                                             000627

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOM*IS *COMOD
FOR SE38=06/02/77-14:33:00 (*0)

   SUBROUTINE COMOD     ENTRY POINT 000227

   STORAGE USED: CODE(1) 000301; DATA(0) 000043; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NERR35

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000066 1L        0001   000127 2L        0001   000170 3L        0001   000206 4L        0000 R 000000 F11
    0000 R 000001 F12       0000 R 000002 F21       0000 R 000003 F22       0000   000004 INJPS
```

```
00101      1*            COMPILER(XM=1),(ADR=IND)                                          000000
00102      2*            SUBROUTINE COMOD (E11,E12,D11,D12,E21,E22,D21,D22,E1,E2,E3,       000000
00103      3*           1 D1,D2,D3,F1,F2,F3                    )                          000000
00105      4*            F11=E11*D11                                                       000000
00106      5*            F12=F11+E12*(D12-D11)                                             000002
00107      6*            F21=E21*D21                                                       000007
00110      7*            F22=F21+E22*(D22-D21)                                             000012
00111      8*            IF (F11.GE.F21) GO TO 1                                           000017
00113      9*            F1=F21                                                            000023
00114     10*            F2=F22                                                            000025
00115     11*            D1=D21                                                            000026
00116     12*            D2=D22                                                            000030
00117     13*            E1=E21                                                            000032
00120     14*            E2=E22                                                            000034
00121     15*            F21=F11                                                           000036
00122     16*            F22=F12                                                           000040
00123     17*            D21=D11                                                           000041
00124     18*            D22=D12                                                           000043
00125     19*            E21=E11                                                           000045
00126     20*            E22=E12                                                           000047
00127     21*            F11=F1                                                            000051
00130     22*            F12=F2                                                            000053
00131     23*            D11=D1                                                            000055
00132     24*            D12=D2                                                            000057
00133     25*            E11=E1                                                            000061
00134     26*            E12=E2                                                            000063
00135     27*           1 CONTINUE                                                         000066
00136     28*            F1=F21                                                            000066
00137     29*            D1=D21+F21/E11                                                    000067
00140     30*            E1=E11*E21/(E11+E21)                                              000072
00141     31*            IF (F11.LT.F22 ) GO TO 2                                          000100
00143     32*            F2=F22                                                            000104
00144     33*            D2=D22+F22/F11                                                    000106
00145     34*            E3=E11*F22/(E11+E22)                                              000111
```

```
00146     35*            F3=F2                                                  000117
00147     36*            D3=D2                                                  000121
00150     37*            E2=E3                                                  000122
00151     38*            RETURN                                                 000123
00152     39*          2 CONTINUE                                               000127
00153     40*            F2=F11                                                 000127
00154     41*            D2=D11+D21+(F11-F21)/E22                               000130
00155     42*            E2=E11*E22/(E11+E22)                                   000136
00156     43*            IF (F22.GT.F12) GO TO 3                                000144
00160     44*            F3=F22                                                 000150
00161     45*            E3=E12*E22/(E12+E22)                                   000152
00162     46*            D3=D22+D11+(F22-F11)/E12                               000160
00163     47*            GO TO 4                                                000166
00164     48*          3 CONTINUE                                               000170
00165     49*            F3=F12                                                 000170
00166     50*            D3=D12+D21+(F12-F21)/F22                               000171
00167     51*            E3=(E12*E22)/(E12+E22)                                 000177
00170     52*          4 IF (D11.EQ.D21) E2=F1                                  000206
00172     53*            RETURN                                                 000212
00173     54*            END                                                    000300

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR:IS .EULR
FOR SE38-06/02/77-14:33:16 (:0)

   SUBROUTINE EULR      ENTRY POINT 000154

   STORAGE USED: CODE(1) 000176: DATA(0) 000043: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   TIME   000012
    0004   STAT   000660
    0005   MASS   001101
    0006   WRITE  000305

   EXTERNAL REFERENCES (BLOCK, NAME)

    0007   FINT
    0010   FEXT
    0011   ACCL
    0012   OUTP
    0013   NERR3S

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000022 1L        0001   000065 130G      0004 R 000600 AA        0004 R 000140 AP        0004 R 000360 AV
    0005   000521 FA        0005   000601 FF        0000 1 000000 I         0005   000061 IF        0005   000001 II
    0003   000000 IM        0005 I 000000 IN        0000   000002 INJPS     0003   000003 IS        0003 I 000011 ITS
    0006   000000 IW        0006   000003 IWW       0004   000004 IW1       0006   000137 JDR       0006   000005 JI
    0006   000043 JJ        0006   000101 JTY       0006   000263 KDR       0006   000175 KE        0006   000217 KI
    0006   000241 KJ        0006   000002 NF        0006   000001 NM        0005   000441 RI        0003 R 000001 TD
    0003 R 000002 TF        0003 R 000010 TT        0005   000361 WT        0004 R 000440 XA        0004 R 000000 XP
    0004 R 000220 XV        0004 R 000520 YA        0004 R 000060 YP        0004 R 000300 YV

00101      1*          COMPILER(XM=1),(ADR=IND)                                    000022
00103      2*          SUBROUTINE EULR                                             000022
00105      3*          INCLUDE PARM                                                000022
00110      4*          INCLUDE TIME                                                000022
00112      5*          INCLUDE STAT                                                000022
00114      6*          INCLUDE MASS                                                000022
00116      7*          INCLUDE WRITE                                               000022
00120      8*        1 CONTINUE                                                    000022
00121      9*          CALL FINT                                                   000022
00122     10*          CALL FEXT                                                   000023
00123     11*          CALL ACCL                                                   000025
00124     12*          CALL OUTP                                                   000027
00125     13*          TT=TT+TD                                                    000031
00126     14*          ITS=ITS+1                                                   000034
00127     15*          DO 2 I=1,IN                                                 000065
00132     16*          XV(1)=XV(1)+XA(1)*TD                                        000065
00133     17*          YV(1)=YV(1)+YA(1)*TD                                        000070
```

```
00134      18*            AV(I)=AV(I)+AA(I)*TD                                  000074
00135      19*            XP(I)=XP(I)+XV(I)*TD                                  000100
00136      20*            YP(I)=YP(I)+YV(I)*TD                                  000104
00137      21*            AP(I)=AP(I)+AV(I)*TD                                  000110
00140      22*          2 CONTINUE                                              000115
00142      23*            IF(TT .GT. TF) RETURN                                 000115
00144      24*            GO TO 1                                               000124
00145      25*            END                                                   000175

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
#FOR:IS .FINT
FOR SE38-06/02/77-14:33:29 (.0)

   SUBROUTINE FINT       ENTRY POINT 000344

   STORAGE USED: CODE(1) 000357: DATA(0) 000031: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   TIME   000012
    0004   MASS   001101
    0005   ELEM   013117
    0006   FORC   063140

   EXTERNAL REFERENCES (BLOCK, NAME)

    0007   LSPR
    0010   LDSM
    0011   TRSP
    0012   TMOP
    0013   BEAM
    0014   PINN
    0015   SLDR
    0016   SPIN
    0017   DSLD
    0020   RIGD
    0021   CPL1
    0022   CPL2
    0023   DGSP
    0024   END3
    0025   ACLMR1
    0026   NLTS
    0027   WRIN
    0030   NLSP
    0031   NLDS
    0032   SLSPR
    0033   TAPB
    0034   NERR2$
    0035   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000170 1L       0001   000234 10L      0001   000320 102L     0001   000114 103L     0001   000240 11L
    0001   000244 12L      0001   000023 121G     0001   000250 13L      0001   000254 14L      0001   000260 15L
    0001   000264 16L      0001   000270 17L      0001   000271 18L      0001   000275 19L      0001   000174 2L
    0001   000301 20L      0001   000305 21L      0001   000311 22L      0001   000315 23L      0001   000316 24L
    0001   000317 25L      0001   000200 3L       0001   000204 4L       0001   000210 5L       0001   000214 6L
    0001   000220 7L       0001   000224 8L       0001   000230 9L       0006   062720 F        0004   000521 FA
    0006   062500 FE       0004   000601 FF       0006   000000 FI       0000 I 000000 I        0005 I 000551 IA
    0000 I 000001 IA1      0000 I 000002 IA2      0000 I 000003 IA3      0005   000265 ID       0005 I 000000 IE
    0004   000061 IF       0004 I 000001 II       0003   000000 IM       0004   000000 IN       0000   000005 INJP$
    0005   001605 IP       0003 I 000003 IS       0005 I 000001 IT       0000 I 000004 ITI      0003 I 000011 ITS
    0005   004731 PP       0004   000441 MI       0003   000001 TD       0003   000002 TF       0003   000010 TT
```

```
00101      1*          COMPILER(XM=1),(ADR=IND)                                    000013
00103      2*          SUBROUTINE FINT                                             000013
00105      3*          INCLUDE TIME                                                000013
00107      4*          INCLUDE PARM                                                000013
00112      5*          INCLUDE MASS                                                000013
00114      6*          INCLUDE ELEM                                                000013
00116      7*          INCLUDE FORC                                                000013
00120      8*          DO 102 I=1,IE                                               000013
00123      9*          IA1=IA(I,1)                                                 000023
00124     10*          IA1=II(IA1)                                                 000027
00125     11*          IA1=IS(IA1)                                                 000033
00126     12*          IA1=MOD(ITS,IA1)                                            000035
00127     13*          IA2=IA(I,2)                                                 000041
00130     14*          IF(IA2.EQ.0) IA2=-1                                         000043
00132     15*          IF(IA2.EQ.-1) GO TO 103                                     000047
00134     16*          IA2=II(IA2)                                                 000052
00135     17*          IA2=IS(IA2)                                                 000060
00136     18*          IA2=MOD(ITS,IA2)                                            000062
00137     19*          IA3=IA(I,3)                                                 000066
00140     20*          IF(IA3.EQ.0) IA3=-1                                         000070
00142     21*          IF(IA3.EQ.-1) GO TO 103                                     000074
00144     22*          IA3=II(IA3)                                                 000077
00145     23*          IA3=IS(IA3)                                                 000105
00146     24*          IA3=MOD(ITS,IA3)                                            000107
00147     25*      103 CONTINUE                                                    000114
00150     26*          IF(IA1.NE.0.AND.IA2.NE.0.AND.IA3.NE.0) GO TO 102            000114
00152     27*          IT1=IT(I)                                                   000127
00153     28*          GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,  000131
00153     29*         C 23,24,25),IT1                                              000131
00154     30*        1 CALL LSPR(I)                                                000170
00155     31*          GO TO 102                                                   000172
00156     32*        2 CALL LDSH(I)                                                000174
00157     33*          GO TO 102                                                   000176
00160     34*        3 CALL TRSP(I)                                                000200
00161     35*          GO TO 102                                                   000202
00162     36*        4 CALL TRDP(I)                                                000204
00163     37*          GO TO 102                                                   000206
00164     38*        5 CALL BEAM(I)                                                000210
00165     39*          GO TO 102                                                   000212
00166     40*        6 CONTINUE                                                    000214
00167     41*          CALL PINN(I)                                                000214
00170     42*          GO TO 102                                                   000216
00171     43*        7 CONTINUE                                                    000220
00172     44*          CALL SLDR(I)                                                000220
00173     45*          GO TO 102                                                   000222
00174     46*        8 CONTINUE                                                    000224
00175     47*          CALL SPIN(I)                                                000224
00176     48*          GO TO 102                                                   000226
00177     49*        9 CONTINUE                                                    000230
00200     50*          CALL DBLD(I)                                                000230
00201     51*          GO TO 102                                                   000232
00202     52*       10 CONTINUE                                                    000234
```

```
00203     53*            CALL RIGD(I)                                    000234
00204     54*            GO TO 102                                       000236
00205     55*         11 CONTINUE                                        000240
00206     56*            CALL CPL1(I)                                    000240
00207     57*            GO TO 102                                       000242
00210     58*         12 CONTINUE                                        000244
00211     59*            CALL CPL2(I)                                    000244
00212     60*            GO TO 102                                       000246
00213     61*         13 CONTINUE                                        000250
00214     62*             CALL DGSP(I)                                   000250
00215     63*            GO TO 102                                       000252
00216     64*         14 CONTINUE                                        000254
00217     65*            CALL END3(I)                                    000254
00220     66*            GO TO 102                                       000256
00221     67*         15 CONTINUE                                        000260
00222     68*            CALL ACLMR1(I)                                  000260
00223     69*            GO TO 102                                       000262
00224     70*         16 CONTINUE                                        000264
00225     71*            CALL NLTS(I)                                    000264
00226     72*            GO TO 102                                       000266
00227     73*         17 CONTINUE                                        000270
00230     74*            GO TO 102                                       000270
00231     75*         18 CONTINUE                                        000271
00232     76*            CALL WRIN(I)                                    000271
00233     77*            GO TO 102                                       000273
00234     78*         19 CONTINUE                                        000275
00235     79*            CALL NLSP(I)                                    000275
00236     80*            GO TO 102                                       000277
00237     81*         20 CONTINUE                                        000301
00240     82*            CALL NLDB(I)                                    000301
00241     83*            GO TO 102                                       000303
00242     84*         21 CONTINUE                                        000305
00243     85*            CALL SLSPR(I)                                   000305
00244     86*            GO TO 102                                       000307
00245     87*         22 CONTINUE                                        000311
00246     88*            CALL TAPH(I)                                    000311
00247     89*            GO TO 102                                       000313
00250     90*         23 CONTINUE                                        000315
00251     91*            GO TO 102                                       000315
00252     92*         24 CONTINUE                                        000316
00253     93*            GO TO 102                                       000316
00254     94*         25 CONTINUE                                        000317
00255     95*            GO TO 102                                       000317
00256     96*        102 CONTINUE                                        000324
00260     97*            RETURN                                          000324
00261     98*            END                                             000356

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS .FEXT
FOR SE3B-06/02/77-14:34:05 (,0)

   SUBROUTINE FEXT        ENTRY POINT 000031

   STORAGE USED: CODE(1) 000037; DATA(0) 000014; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    MASS    001101
    0004    FORC    063140

   EXTERNAL REFERENCES (BLOCK, NAME)

    0005    NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000010 115G       0001   000011* 120G      0004    062720 F         0003   000521 FA        0004 R 062500 FE
    0003   000601 FF         0004   000000 FI         0000 I 000000 I          0003   000061 IF        0003   000001 II
    0003 I 000000 IN         0000   000003 INJP$      0000 I 000001 J          0003   000441 RI        0003   000361 WT


00101        1*          COMPILER(XM=1),(ADR=IND)                                                  000011
00103        2*          SUBROUTINE FEXT                                                           000011
00105        3*          INCLUDE PARM                                                              000011
00110        4*          INCLUDE MASS                                                              000011
00112        5*          INCLUDE FORC                                                              000011
00114        6*          DO 1 I=1,IN                                                               000011
00117        7*          DO 1 J=1,3                                                                000011
00122        8*          FE(I,J)=0.0                                                               000011
00123        9*        1 CONTINUE                                                                  000016
00126       10*          RETURN                                                                    000016
00127       11*          END                                                                       000036

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS .ACCL
FOR SE38-06/02/77-14:34:12 (,0)

   SUBROUTINE ACCL        ENTRY POINT 000255

   STORAGE USED: CODE(1) 003021; DATA(0) 000601; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    STAT    000660
    0004    ELEM    013117
    0005    MASS    001101
    0006    FORC    063140
    0007    PAIR    000503
    0010    GRAV    000222

   EXTERNAL REFERENCES (BLOCK, NAME)

    0011    GRUP
    0012    NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000101 125G    0001   000102 130G    0001   000105 134G    0001   000160 150G    0001   000207 2L
    0003 R 000600 AA      0003   000140 AP      0003   000360 AV      0006 R 062720 F       0005   000521 FA
    0006 R 062500 FE      0005 R 000601 FF      0010 R 000002 FG      0006 R 000000 FI      0010   000000 GA
    0010 R 000001 GG      0000 I 000000 I       0004   000551 IA      0004   000265 ID      0004 I 000000 IE
    0005   000061 IF      0005   000001 II      0005 I 000000 IN      0000   000005 INJP$   0004   001605 IP
    0004   000001 IT      0000 I 000001 J       0000 I 000002 K       0007 I 000026 MG      0007   000173 MI
    0007   000217 MJ      0007   000337 MK      0007   000363 ML      0007   000002 MP      0007 I 000000 NG
    0007   000166 NMG     0007   000001 NP      0004   004731 PP      0005 R 000441 RI      0005 R 000361 WT
    0003 R 000440 XA      0003   000000 XP      0003   000220 XV      0003 R 000520 YA      0003   000060 YP
    0003   000300 YV


00101      1*          COMPILER(XM=1),(ADR=IND)                                      000056
00103      2*          SUBROUTINE ACCL                                               000056
00105      3*          INCLUDE PARM                                                  000056
00110      4*          INCLUDE STAT                                                  000056
00112      5*          INCLUDE ELEM                                                  000056
00114      6*          INCLUDE MASS                                                  000056
00116      7*          INCLUDE FORC                                                  000056
00120      8*          INCLUDE PAIR                                                  000056
00122      9*          INCLUDE GRAV                                                  000056
00124     10*          DO 1 I=1,IN                                                   000056
00127     11*          DO 1 J=1,3                                                    000102
00132     12*          F(I,J)=FE(I,J)                                                000102
00133     13*          DO 3 K=1,IE                                                   000105
00136     14*          F(I,J)=F(I,J)-FI(I,K,J)                                       000105
00137     15*        3 CONTINUE                                                      000110
00141     16*          F(I,J)=F(I,J)+FG(I,J)                                         000110
```

```
00142     17*         1 CONTINUE                                                    000135
00145     18*           IF(NG.NE.0) CALL GRUP                                       000135
00147     19*           DO 2 I=1,IN                                                 000141
00152     20*           IF(MG(I,1).NE.0) GO TO 2                                    000160
00154     21*           XA(I)=(F(I,1)*FF(I,1)+F(I,2)*FF(I,4))*GG/WT(I)              000161
00155     22*           YA(I)=(F(I,1)*FF(I,4)+F(I,2)*FF(I,2))*GG/WT(I)              000172
00156     23*           AA(I)=F(I,3)*FF(I,3)/RI(I)                                  000202
00157     24*         2 CONTINUE                                                    000223
00161     25*           RETURN                                                      000223
00162     26*           END                                                         000301

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR,IS .OUTP
FOR BE38-06/02/77-14134119 (,0)

   SUBROUTINE OUTP       ENTRY POINT 000656

   STORAGE USED: CODE(1) 000676: DATA(0) 000502: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    TIME    000012
    0004    STAT    000660
    0005    MASS    001101
    0006    ELEM    013117
    0007    CONN    002720
    0010    GRAV    000222
    0011    FORC    063140
    0012    WRITE   000305
    0013    FUSS    062500

   EXTERNAL REFERENCES (BLOCK, NAME)

    0014    RKIN
    0015    NWDUS
    0016    NIO2S
    0017    SIN
    0020    COS
    0021    NIO1S
    0022    NERR3S

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000077 11L      0001    000176 170G     0001    000627 22L      0001    000324 230G     0001    000537 267G
    0001    000570 301G     0001    000613 312G     0000    000415 333F     0001    000412 35L      0001    000414 36L
    0001    000467 37L      0001    000544 44L      0000    000420 444F     0001    000161 505L     0001    000313 55L
    0000    000423 555F     0001    000504 65L      0001    000506 66L      0001    000617 67L      0001    000602 68L
    0001    000225 7L       0001    000227 77L      0000    000413 9F       0000    000411 99F      0000 R  000000 A
    0004 R  000600 AA       0007    001760 AC       0004 R  000140 AP       0000 R  000404 APUS     0004 R  000360 AV
    0000 R  000364 AX       0000 R  000365 AY       0000 R  000011 B        0000 R  000366 BX       0000 R  000367 BY
    0011    062720 F        0005    000521 FA       0011    062500 FE       0005    000601 FF       0010    000002 FG
    0011 R  000000 FI       0013 R  000000 FT       0010    000000 GA       0010    000001 GG       0006    000551 IA
    0007    000000 IC       0006    000265 ID       0006    000000 IE       0005    000061 IF       0005    000001 II
    0000 I  000370 IJKLM    0003    000000 IM       0005    000000 IN       0000    000427 INJPS    0006    001605 IP
    0003    000003 IS       0006    000001 IT       0003    000011 ITS      0012 I  000000 IW       0012 I  000003 IWW
    0012 I  000004 IW1      0000 I  000362 J        0000 I  000360 JA       0000 I  000361 JB       0012 I  000137 JDW
    0012 I  000005 JI       0012 I  000043 JJ       0012 I  000101 JJY      0000 I  000357 K        0000 I  000403 KDD
    0012 I  000263 KDR      0012 I  000175 KE       0000 I  000400 KEE      0000 I  000377 KF       0012 I  000217 KI
    0000 I  000401 KII      0012 I  000241 KJ       0000 I  000402 KJJ      0000 I  000363 L        0012 I  000002 NF
    0012 I  000001 NM       0000 I  000355 NO       0000 I  000356 NB       0000 I  000407 N1       0000 I  000410 N2
    0006    004731 PP       0005    000441 RI       0003    000001 ID       0003    000002 IF       0003 R  000010 IT
    0005    000361 WT       0004    000440 XA       0007 R  000060 XC       0000 R  000375 XDDL     0000 R  000373 XDL
    0000 R  000405 XG       0000 R  000371 XL       0004    000000 XP       0004    000220 XV       0004    000520 YA
    0007 R  001020 YC       0000 R  000376 YDDL     0000 R  000374 YDL      0000 R  000406 YG       0000 R  000372 YL
    0004    000060 YP       0004    000300 YV
```

```
00101      1*          COMPILER(XM=1),(ADR=IND)                                    000055
00103      2*          SUBROUTINE OUTP                                             000055
00105      3*          INCLUDE PARM                                                000055
00110      4*          INCLUDE TIME                                                000055
00112      5*          INCLUDE STAT                                                000055
00114      6*          INCLUDE MASS                                                000055
00116      7*          INCLUDE ELEM                                                000055
00120      8*          INCLUDE CONN                                                000055
00122      9*          INCLUDE GRAV                                                000055
00124     10*          INCLUDE FORC                                                000055
00126     11*          INCLUDE WRITE                                               000055
00130     12*          DIMENSION A(3,3),B(NOUT)                                    000055
00131     13*          COMMON/FUSS/FT(NUMM,NELE,3)                                 000055
00132     14*       99 FORMAT(//1X,E15.5)                                          000055
00133     15*        9 FORMAT( 1X,E15.5)                                           000055
00134     16*      333 FORMAT(1X,8(E15.5))                                         000055
00135     17*      444 FORMAT( 1X,8(E15.5))                                        000055
00136     18*      555 FORMAT(/)                                                   000055
00137     19*          IF(IW.NE.0) GO TO 22                                        000055
00141     20*          NO=NM+NF                                                    000057
00142     21*          NS=NM+1                                                     000062
00143     22*          IF(IW1.EQ.0) GO TO 11                                       000065
00145     23*          WRITE(6,9) TT                                               000067
00150     24*          GO TO 505                                                   000075
00151     25*       11 IF(NO.LE.8) IW1=5                                           000077
00153     26*          IF(NO.GT.8.AND.NO.LE.16) IW1=3                              000104
00155     27*          IF(NO.GT.16.AND.NO.LE.32) IW1=2                             000124
00157     28*          IF(NO.GT.32) IW1=1                                          000144
00161     29*          WRITE(6,99) TT                                              000152
00164     30*      505 IW1=IW1-1                                                   000161
00165     31*          IF(NM.EQ.0) GO TO 55                                        000163
00167     32*          DO 33 K=1,NM                                                000165
00172     33*          JA=JDR(K)                                                   000176
00173     34*          JB=JTY(K)                                                   000177
00174     35*          J=JI(K)                                                     000201
00175     36*          L=JJ(K)                                                     000203
00176     37*          AX=0.0                                                      000205
00177     38*          AY=0.0                                                      000206
00200     39*          IF(L.EQ.0) GO TO 7                                          000207
00202     40*          BX=XC(J,L)                                                  000211
00203     41*          BY=YC(J,L)                                                  000221
00204     42*          GO TO 77                                                    000223
00205     43*        7 BX=0.0                                                      000225
00206     44*          BY=0.0                                                      000225
00207     45*       77 A(1,3)=AP(J)                                                000227
00210     46*          A(2,3)=AV(J)                                                000241
00211     47*          A(3,3)=AA(J)                                                000243
00212     48*          IJKLM = JK                                                  000245
00213     49*          CALL RKIN(AX,AY,BX,BY,J,IJKLM,XL,YL,XDL,YDL,XDDL,YDDL)      000247
00214     50*          A(1,1) = XL                                                 000266
00215     51*          A(1,2) = YL                                                 000270
00216     52*          A(2,1) = XDL                                                000272
00217     53*          A(2,2) = YDL                                                000274
00220     54*          A(3,1) = XDDL                                               000276
00221     55*          A(3,2) = YDDL                                               000300
00222     56*       33 B(K)=A(JB,JA)                                               000302
```

```
00224     57*       55 IF(NF.EQ.0) GO TO 65                                         000313
00226     58*          KF=1                                                         000314
00227     59*          DO 30 K=NS,NO                                                000324
00232     60*          KFE=KF(KF)                                                   000337
00233     61*          KII=KI(KF)                                                   000342
00234     62*          KJJ=KJ(KF)                                                   000347
00235     63*          KDD=KDR(KF)                                                  000352
00236     64*          APOS=AP(KII)                                                 000365
00237     65*          FT(KII,KEE,KDD)=FI(KII,KEE,KDD)                              000367
00240     66*          IF(KDD.NE.3) GO TO 37                                        000371
00242     67*          IF(KJJ.EQ.0) GO TO 35                                        000374
00244     68*          XL=XC(KII,KJJ)                                               000376
00245     69*          YL=YC(KII,KJJ)                                               000406
00246     70*          GO TO 36                                                     000410
00247     71*       35 XL=0.0                                                       000412
00250     72*          YL=0.0                                                       000412
00251     73*       36 XG=XL*COS(APOS)-YL*SIN(APOS)                                 000414
00252     74*          YG=XL*SIN(APOS)+YL*COS(APOS)                                 000445
00253     75*          FT(KII,KEE,KDD)=FI(KII,KEE,KDD)-XG*FI(KII,KEE,2)+YG*FI(KII,KEE,1)   000460
00254     76*       37 B(K)=FT(KII,KEE,KDD)                                         000467
00255     77*       30 KF=KF+1                                                      000477
00257     78*       65 N1=1                                                         000504
00260     79*       66 N2=N1+7                                                      000506
00261     80*          IF(N2.GT.NO) GO TO 44                                        000510
00263     81*          IF(IW1.EQ.0.AND.N2.EQ.NO) GO TO 68                           000513
00265     82*          WRITE(6,444) (B(K),K=N1,N2)                                  000525
00273     83*          GO TO 67                                                     000542
00274     84*       44 N2=NO                                                        000544
00275     85*          IF(IW1.EQ.0.AND.N2.EQ.NO) GO TO 68                           000545
00277     86*          WRITE(6,444) (B(K),K=N1,N2)                                  000556
00305     87*          WRITE(6,555)                                                 000573
00307     88*          GO TO 67                                                     000600
00310     89*       68 WRITE(6,333) (B(K),K=N1,N2)                                  000602
00316     90*       67 N1=N2+1                                                      000617
00317     91*          IF(N1.LE.NO) GO TO 66                                        000621
00321     92*          IW=IWW                                                       000624
00322     93*       22 CONTINUE                                                     000627
00323     94*          IW=IW-1                                                      000627
00324     95*          RETURN                                                       000631
00325     96*          END                                                          000675

END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR:I8 .GRUP
FOR SE3B-06/02/77-14134131 (:0)

   SUBROUTINE GRUP       ENTRY POINT 002022

   STORAGE USED: CODE(1) 002047; DATA(0) 001617; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    MASS    001101
    0004    FORC    063140
    0005    CONN    002720
    0006    GRAV    000222
    0007    PAIR    000503
    0010    TIME    000012
    0011    STAT    000660
    0012    SAVE    030470
    0013    ELEM    013117

   EXTERNAL REFERENCES (BLOCK, NAME)

    0014    MALG
    0015    LEQS
    0016    NERR2S
    0017    COS
    0020    SIN
    0021    SQRT
    0022    NERR3S

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000551 10L      0001    000740 11L      0001    001172 12L      0001    001231 13L      0001    000132 135G
    0001    000174 142G     0001    000175 145G     0001    001761 15L      0001    000200 153G     0001    001626 16L
    0001    000221 163G     0001    001662 17L      0001    001706 18L      0001    001731 19L      0001    001411 2L
    0001    001743 20L      0001    001755 21L      0001    000257 3L       0001    001422 376G     0001    001424 401G
    0001    001517 412G     0001    001570 422G     0001    001601 430G     0001    001624 442G     0001    000354 9L
    0011 R  000600 AA       0005 R  001760 AC       0000 R  001364 AF       0000 R  001465 AK       0000 R  001456 AL
    0000 R  000537 AM       0011 R  000140 AP       0011 R  000360 AV       0000 R  000000 A11      0000 R  000220 A12
    0000 R  001023 A2       0000 R  001466 CAK      0000 R  001460 CAL      0000 R  001455 CTA      0000 R  001444 CTK
    0000 R  001446 CTL      0000 R  001473 DET      0004 R  062720 F        0003    000521 FA       0004    062500 FE
    0003    000601 FF       0000 R  001403 FFF      0006    000002 FG       0004    000000 FI       0000 R  000504 F1
    0000 R  000520 F2       0006    000000 GA       0006 R  000001 GG       0000 I  001423 I        0013    000551 IA
    0005    000000 IC       0013 I  000265 ID       0000 I  001441 IDP      0013    000000 IE       0003    000061 IF
    0003    000001 II       0000 I  001425 IL       0010    000000 IM       0003    000000 IN       0000    001514 INJPS
    0013 I  001605 IP       0000 I  001432 IR1      0000 I  001433 IR2      0000 I  001434 IR3      0000 I  001435 IR4
    0000 I  001436 IR5      0000 I  001437 IR6      0010    000003 I8       0000 I  001474 I8CALE   0013 I  000001 IT
    0010    000011 IT8      0000 I  001420 J        0000 I  001424 K        0000 I  001421 L        0000 I  001440 L8
    0000 I  001422 M        0000 I  001431 ME       0000 I  001442 ME1      0000 I  001443 ME2      0007    000026 MG
    0007 I  000173 MI       0007 I  000217 MJ       0007 I  000337 MK       0007 I  000363 ML       0000 I  001427 MM
    0007    000002 MP       0000 I  001426 M1       0000 I  001430 M2       0000 I  001470 N        0000 I  001472 NCC
    0000 I  001417 NCOL     0007 I  000000 NG       0007 I  000166 NMG      0007    000001 NP       0000 I  001471 NRC
    0013 R  004731 PP       0003 R  000441 RI       0000 R  001467 SAK      0000 R  001457 SAL      0000 R  001454 STA
    0000 R  001445 STK      0000 R  001447 STL      0012 R  000000 SV       0010    000001 TD       0010    000002 TF
```

```
0010   000010 IT       0003 R 000361 WT       0011 R 000440 XA       0005 R 000060 XC       0000 R 001461 XKM
0000 R 001450 XMK      0000 R 001463 XML      0000 R 001452 XNL      0011 R 000000 XP       0011 R 000220 XV
0011 R 000520 YA       0005 R 001020 YC       0000 R 001462 YKM      0000 R 001451 YMK      0000 R 001464 YML
0000 R 001453 YNL      0011 R 000060 YP       0011 R 000300 YV
```

```
00101      1*          COMPILER(XM=1),(ADR=IND)                                                    000113
00103      2*          SUBROUTINE GRUP                                                             000113
00105      3*          INCLUDE PARM                                                                000113
00110      4*          INCLUDE MASS                                                                000113
00112      5*          INCLUDE FORC                                                                000113
00114      6*          INCLUDE CONN                                                                000113
00116      7*          INCLUDE GRAV                                                                000113
00120      8*          INCLUDE PAIR                                                                000113
00122      9*          INCLUDE TIME                                                                000113
00124     10*          INCLUDE STAT                                                                000113
00126     11*          INCLUDE SAVE                                                                000113
00130     12*          INCLUDE ELEM                                                                000113
00132     13*          DIMENSION A11(NROWF1,NROWF1),A12(NROWF1,NROWF2),F1(NROWF1),F2(NROW          000113
00132     14*         1F2),AM(NROWF2,NROWF1),A2(NROWF2,NROWF2),AF(NROWF2),FFF(NROWF1)              000113
00133     15*          NCOL=NROWF2                                                                 000113
00134     16*          DO 1 J=1,NG                                                                 000115
00137     17*          L=0                                                                         000146
00140     18*          M=3*NMG(J)                                                                  000147
00141     19*          DO 4 I=1,M                                                                  000153
00144     20*          DO 14 K=1,NCOL                                                              000175
00147     21*          A12(I,K)=0.0                                                                000175
00150     22*       14 CONTINUE                                                                    000200
00152     23*          DO 4 K=1,M                                                                  000200
00155     24*          A11(I,K)=0.0                                                                000200
00156     25*        4 CONTINUE                                                                    000210
00161     26*          IL=NMG(J)                                                                   000210
00162     27*          DO 2 I=1,IL                                                                 000212
00165     28*          K=3*I                                                                       000221
00166     29*          M1=MI(I,J)                                                                  000224
00167     30*          A11(K-2,K-2)=GG/WT(M1)                                                      000226
00170     31*          A11(K-1,K-1)=GG/WT(M1)                                                      000234
00171     32*          A11(K,K)=1./R1(M1)                                                          000235
00172     33*          F1(K-2)=F(M1,1)                                                             000242
00173     34*          F1(K-1)=F(M1,2)                                                             000247
00174     35*          F1(K)=F(M1,3)                                                               000253
00175     36*          K=0                                                                         000255
00176     37*        3 IF(K.EQ.MK(I,J)) GO TO 2                                                    000257
00200     38*          K=K+1                                                                       000261
00201     39*          MM=ML(I,J,K)                                                                000264
00202     40*          M2=MI(MM,J)                                                                 000274
00203     41*          ME=MJ(I,J,K)                                                                000301
00204     42*          IR1=3*I-2                                                                   000305
00205     43*          IR2=3*I-1                                                                   000312
00206     44*          IR3=3*I                                                                     000315
00207     45*          IR4=3*MM-2                                                                  000317
00210     46*          IR5=3*MM-1                                                                  000323
00211     47*          IR6=3*MM                                                                    000326
00212     48*          LS=IT(ME)-5                                                                 000330
00213     49*          IDP=ID(ME)                                                                  000335
```

```
00214     50*          GO TO (9,10,11,12,13),L8                                                    000341
00215     51*        9 CONTINUE                                                                    000354
00216     52*          ME1=IP(ME,1,1)                                                              000354
00217     53*          ME2=IP(ME,2,1)                                                              000357
00220     54*          CTK=COS(AP(M1))                                                             000363
00221     55*          STK=SIN(AP(M1))                                                             000373
00222     56*          CTL=COS(AP(M2))                                                             000401
00223     57*          STL=SIN(AP(M2))                                                             000411
00224     58*          XMK=XC(M1,ME1)                                                              000417
00225     59*          YMK=YC(M1,ME1)                                                              000426
00226     60*          XNL=XC(M2,ME2)                                                              000432
00227     61*          YNL=YC(M2,ME2)                                                              000441
00230     62*          A12(IR1,L+1)=1.0                                                            000445
00231     63*          A12(IR2,L+1)=0.0                                                            000453
00232     64*          A12(IR3,L+1)=-XMK*STK+YMK*CTK                                               000456
00233     65*          A12(IR4,L+1)=-1.0                                                           000465
00234     66*          A12(IR5,L+1)= 0.0                                                           000470
00235     67*          A12(IR6,L+1)= XNL*STL+YNL*CTL                                               000473
00236     68*          A12(IR1,L+2)= 0.0                                                           000501
00237     69*          A12(IR2,L+2)= 1.0                                                           000502
00240     70*          A12(IR3,L+2)= XMK*CTK-YMK*STK                                               000503
00241     71*          A12(IR4,L+2)= 0.0                                                           000511
00242     72*          A12(IR5,L+2)=-1.0                                                           000512
00243     73*          A12(IR6,L+2)=-XNL*CTL+YNL*STL                                               000513
00244     74*          F2(L+1)                                                                     000520
00244     75*        C      =AV(M1)**2*(XMK*CTK-YMK*STK)-AV(M2)**2*(XNL*CTL-YNL*STL)              000520
00245     76*          F2(L+2)=AV(M1)**2*(XMK*STK+YMK*CTK)-AV(M2)**2*(XNL*STL+YNL*CTL)             000537
00246     77*          L=L+2                                                                       000544
00247     78*          GO TO 3                                                                     000547
00250     79*       10 CONTINUE                                                                    000551
00251     80*          ME1=IP(ME,1,1)                                                              000551
00252     81*          STA=SIN(AP(M1)+.017453293*AC(M1,ME1))                                       000554
00253     82*          CTA=COS(AP(M1)+.017453293*AC(M1,ME1))                                       000571
00254     83*          A12(IR1,L+1)=-STA                                                           000575
00255     84*          A12(IR2,L+1)= CTA                                                           000603
00256     85*          A12(IR3,L+1)=SV(5,ME)                                                       000606
00257     86*          A12(IR4,L+1)= STA                                                           000616
00260     87*          A12(IR5,L+1)=-CTA                                                           000621
00261     88*          A12(IR6,L+1)                                                                000624
00261     89*        C           =SV(5,ME)-(XP(M2)-XP(M1))*CTA-(YP(M2)-YP(M1))*STA                 000624
00262     90*          A12(IR1,L+2)= 0.0                                                           000652
00263     91*          A12(IR2,L+2)= 0.0                                                           000653
00264     92*          A12(IR3,L+2)=-PP(1,IDP,7)/2.0                                               000654
00265     93*          A12(IR4,L+2)= 0.0                                                           000663
00266     94*          A12(IR5,L+2)= 0.0                                                           000665
00267     95*          A12(IR6,L+2)= PP(1,IDP,7)/2.0                                               000667
00270     96*          F2(L+1)                                                                     000670
00270     97*        C      =AV(M1)*(CTA*(2.*(XV(M1)-XV(M2))+AV(M1)*(YP(M1)-YP(M2)))+             000670
00270     98*        1 STA*(2.*(YV(M1)-YV(M2))-AV(M1)*(XP(M1)-XP(M2))))                            000670
00271     99*          F2(L+2)=0.0                                                                 000732
00272    100*          L=L+2                                                                       000733
00273    101*          GO TO 3                                                                     000736
00274    102*       11 CONTINUE                                                                    000740
00275    103*          ME1=IP(ME,1,1)                                                              000740
00276    104*          ME2=IP(ME,2,1)                                                              000743
00277    105*          AL=AP(M2)+AC(M2,ME2)*.017453293                                             000747
00300    106*          SAL=SIN(AL)                                                                 000760
```

```
00301    107*          CAL=COS(AL)                                                    000764
00302    108*          STK=SIN(AP(M1))                                                000770
00303    109*          CTK=COS(AP(M1))                                                001000
00304    110*          XKM=YC(M1,ME1)*STK-XC(M1,ME1)*CTK                              001006
00305    111*          YKM=-YC(M1,ME1)*CTK-XC(M1,ME1)*STK                             001022
00306    112*          XML=XP(M1)-XP(M2)-XKM                                          001030
00307    113*          YML=YP(M1)-YP(M2)-YKM                                          001040
00310    114*          A12(IR1,L+1)=-SAL                                              001050
00311    115*          A12(IR2,L+1)= CAL                                              001056
00312    116*          A12(IR3,L+1)=-YKM*SAL-XKM*CAL                                  001062
00313    117*          A12(IR4,L+1)= SAL                                              001070
00314    118*          A12(IR5,L+1)=-CAL                                              001074
00315    119*          A12(IR6,L+1)=-YML*SAL-XML*CAL                                  001100
00316    120*          F2(L+1)                                                        001106
00316    121*        C      =2.*AV(M2)*((YV(M1)-YV(M2))*SAL+(XV(M1)-XV(M2))*CAL)     001106
00316    122*        C +AV(M1)**2*(XKM*SAL-YKM*CAL)+AV(M2)**2*(YML*CAL-XML*SAL)     001106
00316    123*        C +2.*AV(M1)*AV(M2)*(YKM*CAL-XKM*SAL)                          001106
00317    124*          L=L+1                                                          001165
00320    125*          GO TO 3                                                        001170
00321    126*       12 CONTINUE                                                       001172
00322    127*          A12(IR1,L+1)= 0.0                                              001172
00323    128*          A12(IR2,L+1)= 0.0                                              001176
00324    129*          A12(IR3,L+1)= PP(1,IDP,9)/2.0                                  001201
00325    130*          A12(IR4,L+1)= 0.0                                              001211
00326    131*          A12(IR5,L+1)= 0.0                                              001214
00327    132*          A12(IR6,L+1)=-PP(1,IDP,9)/2.0                                  001217
00330    133*          F2(L+1)= 0.0                                                   001222
00331    134*          L=L+1                                                          001224
00332    135*          GO TO 3                                                        001227
00333    136*       13 CONTINUE                                                       001231
00334    137*          ME1=IP(ME,1,1)                                                 001231
00335    138*          ME2=IP(ME,2,1)                                                 001234
00336    139*          AK=AP(M1)+AC(M1,ME1)*.017453293                                001240
00337    140*          AL=AP(M2)+AC(M2,ME2)*.017453293                                001251
00340    141*          CAK=COS(AK)                                                    001262
00341    142*          CAL=COS(AL)                                                    001266
00342    143*          SAK=SIN(AK)                                                    001272
00343    144*          SAL=SIN(AL)                                                    001276
00344    145*          A12(IR1,L+1)=-CAK                                              001302
00345    146*          A12(IR2,L+1)=-SAK                                              001310
00346    147*          A12(IR3,L+1)=-SV(1,ME)                                         001314
00347    148*          A12(IR4,L+1)= CAL                                              001324
00350    149*          A12(IR5,L+1)= SAL                                              001330
00351    150*          A12(IR6,L+1)= SV(2,ME)                                         001333
00352    151*          A12(IR1,L+2)= SAK                                              001337
00353    152*          A12(IR2,L+2)=-CAK                                              001340
00354    153*          A12(IR3,L+2)=-SV(3,ME)                                         001341
00355    154*          A12(IR4,L+2)=-SAL                                              001343
00356    155*          A12(IR5,L+2)= CAL                                              001344
00357    156*          A12(IR6,L+2)= SV(4,ME)                                         001345
00360    157*          A12(IR1,L+3)= 0.0                                              001347
00361    158*          A12(IR2,L+3)= 0.0                                              001350
00362    159*          A12(IR3,L+3)=-1.0                                              001351
00363    160*          A12(IR4,L+3)= 0.0                                              001353
00364    161*          A12(IR5,L+3)= 0.0                                              001354
00365    162*          A12(IR6,L+3)= 1.0                                              001355
00366    163*          F2(L+1)                                                        001356
```

```
00366    164*          C      S=AV(M1)**2*SV(3,ME)+AV(M2)**2*SV(4,ME)                                  001356
00367    165*           F2(L+2)=-AV(M1)**2*SV(1,ME)+AV(M2)**2*SV(2,ME)                                  001375
00370    166*           F2(L+3)= 0.0                                                                    001403
00371    167*           L=L+3                                                                           001404
00372    168*           GO TO 3                                                                         001407
00373    169*         2 CONTINUE                                                                        001414
00375    170*           DO 5 K=1,L                                                                      001414
00400    171*           DO 5 N=1,M                                                                      001424
00403    172*           AM(K,N)=A12(N,K)*A11(N,N)                                                       001424
00404    173*         5 CONTINUE                                                                        001436
00407    174*           CALL MALG(1,AM,1.,0,L,M,NROWF2,NROWF1,A12,1.,0,M,L,NROWF1,NROWF2,               001436
00407    175*          C A2,NRC,NCC,NROWF2,NROWF2)                                                     001436
00410    176*           CALL MALG(1,AM,1.,0,L,M,NROWF2,NROWF1,F1,1.,0,M,1,NROWF1,1,AF,                  001464
00410    177*          C NRC,NCC,NROWF2,1)                                                             001464
00411    178*           DO 6 K=1,L                                                                      001512
00414    179*           F2(K)=AF(K)-F2(K)                                                               001517
00415    180*         6 CONTINUE                                                                        001522
00417    181*           CALL LEQS(A2,F2,L,1,NROWF2,1,DET,ISCALE)                                        001522
00420    182*           CALL MALG(1,A12,1.,0,M,L,NROWF1,NROWF2,F2,1.,0,L,1,NROWF2,1,FFF,                001534
00420    183*          C NRC,NCC,NROWF1,1)                                                             001534
00421    184*           DO 7 K=1,M                                                                      001562
00424    185*           FFF(K)=(F1(K)-FFF(K))*A11(K,K)                                                  001570
00425    186*         7 CONTINUE                                                                        001601
00427    187*           DO 8 I=1,IL                                                                     001601
00432    188*           M1=MI(I,J)                                                                      001601
00433    189*           XA(M1)=FFF(3*I-2)                                                               001602
00434    190*           YA(M1)=FFF(3*I-1)                                                               001606
00435    191*           AA(M1)=FFF(3*I)                                                                 001612
00436    192*         8 CONTINUE                                                                        001617
00440    193*           L=0                                                                             001617
00441    194*           DO 15 I=1,IL                                                                    001624
00444    195*           K=0                                                                             001624
00445    196*        16 IF (K.EQ.MK(I,J)) GO TO 15                                                      001626
00447    197*           K=K+1                                                                           001630
00450    198*           ME=MJ(I,J,K)                                                                    001640
00451    199*           LS=IT(ME)-5                                                                     001644
00452    200*           GO TO (17,18,19,20,21),LS                                                       001647
00453    201*        17 CONTINUE                                                                        001662
00454    202*           SV(1,ME)=SQRT(F2(L+1)**2+F2(L+2)**2)                                            001665
00455    203*           L=L+2                                                                           001701
00456    204*           GO TO 16                                                                        001704
00457    205*        18 CONTINUE                                                                        001706
00460    206*           SV(1,ME)=(ABS(F2(L+1)+F2(L+2))+ABS(F2(L+1)-F2(L+2)))/2.                         001711
00461    207*           SV(2,ME)=F2(L+1)                                                                001722
00462    208*           L=L+2                                                                           001724
00463    209*           GO TO 16                                                                        001727
00464    210*        19 CONTINUE                                                                        001731
00465    211*           SV(1,ME)=ABS(F2(L+1))                                                           001734
00466    212*           L=L+1                                                                           001736
00467    213*           GO TO 16                                                                        001741
00470    214*        20 CONTINUE                                                                        001743
00471    215*           SV(1,ME)=F2(L+1)                                                                001746
00472    216*           L=L+1                                                                           001750
00473    217*           GO TO 16                                                                        001753
00474    218*        21 CONTINUE                                                                        001755
00475    219*           L=L+3                                                                           001755
00476    220*           GO TO 16                                                                        001757
```

```
00477     221*        15 CONTINUE                                   001770
00501     222*         1 CONTINUF                                   001770
00503     223*           RETURN                                     001770
00504     224*           END                                        002046

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS ,LSPR
FOR SE38=06/02/77-14:34:55 (,0)

   SUBROUTINE LSPR       ENTRY POINT 000356

   STORAGE USED: CODE(1) 0004021 DATA(0) 0001021 BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    ELEM    013117
    0004    STAT    000660
    0005    FORC    063140
    0006    CONN    002720

   EXTERNAL REFERENCES (BLOCK, NAME)

    0007    RKIN
    0010    SQRT
    0011    NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000224 1L       0001    000307 2L       0004    000600 AA       0006    001760 AC       0004    000140 AP
    0004    000360 AV       0000 R  000007 AX       0000 R  000010 AY       0000 R  000011 BX       0000 R  000012 BY
    0000 R  000034 CA       0000 R  000032 DL       0000 R  000030 DX       0000 R  000031 DY       0005    062720 F
    0000 R  000035 FA       0000 R  000036 FD       0005    062500 FE       0005 R  000000 FI       0000 R  000002 FL
    0003 I  000551 IA       0006    000000 IC       0003 I  000265 ID       0003    000000 IE       0000 I  000013 IJKLM
    0000    000040 INJPS    0003 I  001605 IP       0003    000001 IT       0000 I  000000 J        0000 I  000003 K
    0000 I  000004 L        0000 I  000005 M        0000 I  000006 N        0003 R  004731 PP       0000 R  000033 SA
    0000 R  000001 SK       0004    000440 XA       0006 R  000040 XC       0000 R  000020 XDDM     0000 R  000026 XDDN
    0000 R  000016 XDM      0000 R  000024 XDN      0000 R  000014 XM       0000 R  000022 XN       0004 R  000000 XP
    0004    000220 XV       0004    000520 YA       0006 R  001020 YC       0000 R  000021 YDDM     0000 R  000027 YDDN
    0000 R  000017 YDM      0000 R  000025 YDN      0000 R  000015 YM       0000 R  000023 YN       0004 R  000060 YP
    0004    000300 YV


00101       1*          COMPILER(XM=1),(ADR=IND)                                                    000044
00103       2*          SUBROUTINE LSPR(I)                                                          000044
00105       3*          INCLUDE PARM                                                                000044
00110       4*          INCLUDE ELEM                                                                000044
00112       5*          INCLUDE STAT                                                                000044
00114       6*          INCLUDE FORC                                                                000044
00116       7*          INCLUDE CONN                                                                000044
00120       8*          J=ID(I)                                                                     000044
00121       9*          SK=PP(1,J,1)                                                                000050
00122      10*          FL=PP(2,J,1)                                                                000052
00123      11*          K=IA(I,1)                                                                   000054
00124      12*          L=IA(I,2)                                                                   000056
00125      13*          M=IP(I,1,1)                                                                 000060
00126      14*          N=IP(I,2,1)                                                                 000062
00127      15*          AX=0.0                                                                      000064
```

```
00130     16*          AY=0.0                                                                    000065
00131     17*          BX=XC(K,M)                                                                000066
00132     18*          BY=YC(K,M)                                                                000074
00133     19*          IJKLM = 4                                                                 000100
00134     20*          CALL RKIN(AX,AY,BX,BY,K,IJKLM,XM,YM,XDM,YDM,XDDM,YDDM)          K + M     000102
00135     21*          BX=XC(L,N)                                                                000121
00136     22*          BY=YC(L,N)                                                                000130
00137     23*          CALL RKIN(AX,AY,BX,BY,L,IJKLM,XN,YN,XDN,YDN,XDDN,YDDN)          L + N     000134
00140     24*          DX=XN-XM                                                                  000152
00141     25*          DY=YN-YM                                                                  000155
00142     26*          DL=SQRT(DX*DX+DY*DY)                                                      000160
00143     27*          SA=DY/DL                                                                  000170
00144     28*          CA=DX/DL                                                                  000173
00145     29*          FA=(DL-FL)*SK                                                             000176
00146     30*          FD = PP(4,J,1)*((XDN-XDM)*CA + (YDN-YDM)*SA)                              000201
00147     31*          IF(ABS(PP(3,J,1)) .LT. 0.0001) GO TO 1                                    000213
00151     32*          IF(ABS(FA).GE.PP(3,J,1 )) GO TO 2                                         000217
00153     33*        1 CONTINUE                                                                  000224
00154     34*          FA = FA + FD                                                              000224
00155     35*          FI(K,I,1)=-FA*CA                                                          000226
00156     36*          FI(K,I,2)=-FA*SA                                                          000234
00157     37*          FI(K,I,3)=-FI(K,I,1)*(YM-YP(K))+FI(K,I,2)*(XM-XP(K))                      000241
00160     38*          FI(L,I,1)=-FI(K,I,1)                                                      000255
00161     39*          FI(L,I,2)=-FI(K,I,2)                                                      000263
00162     40*          FI(L,I,3)=-FI(L,I,1)*(YN-YP(L))+FI(L,I,2)*(XN-XP(L))                      000267
00163     41*          RETURN                                                                    000303
00164     42*        2 CONTINUE                                                                  000307
00165     43*          FI(K,I,1)=0.0                                                             000307
00166     44*          FI(K,I,2)=0.0                                                             000313
00167     45*          FI(K,I,3)=0.0                                                             000316
00170     46*          FI(L,I,1)=0.0                                                             000317
00171     47*          FI(L,I,2)=0.0                                                             000326
00172     48*          FI(L,I,3)=0.0                                                             000327
00173     49*          RETURN                                                                    000330
00174     50*          END                                                                       000401

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS ,LDSH
FOR SE3B-06/02/77-14:35:30 (,0)

   SUBROUTINE LDSH       ENTRY POINT 000306

   STORAGE USED: CODE(1) 000332; DATA(0) 000076; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   ELEM   013117
    0004   STAT   000660
    0005   FORC   063140
    0006   CONN   002720

   EXTERNAL REFERENCES (BLOCK, NAME)

    0007   RKIN
    0010   SQRT
    0011   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0004   000600 AA        0006   001760 AC        0004   000140 AP        0004   000360 AV        0000 R 000006 AX
    0000 R 000007 AY        0000 R 000010 BX        0000 R 000011 BY        0000 R 000033 CA        0000 R 000001 DK
    0000 R 000031 DL        0000 R 000027 DX        0000 R 000030 DY        0005   062720 F         0000 R 000034 FA
    0005   062500 FE        0005 R 000000 FI        0003 I 000551 IA        0006   000000 IC        0003 I 000265 ID
    0003   000000 IE        0000 I 000012 IJKLM     0000   000035 INJP$     0003 I 001605 IP        0003   000001 IT
    0000 I 000000 J         0000 I 000002 K         0000 I 000003 L         0000 I 000004 M         0000 I 000005 N
    0003 R 004731 PP        0000 R 000032 SA        0004   000440 XA        0006 R 000060 XC        0000 R 000017 XDDM
    0000 R 000025 XDDN      0000 R 000015 XDM       0000 R 000023 XDN       0000 R 000013 XM        0000 R 000021 XN
    0004 R 000000 XP        0004   000220 XV        0004   000520 YA        0006 R 001020 YC        0000 R 000020 YDDM
    0000 R 000026 YDDN      0000 R 000016 YDM       0000 R 000024 YDN       0000 R 000014 YM        0000 R 000022 YN
    0004 R 000060 YP        0004   000300 YV


00101      1*      COMPILER(XM=1),(ADR=IND)                               000043
00103      2*      SUBROUTINE LDSH (I)                                    000043
00105      3*      INCLUDE PARM                                           000043
00110      4*      INCLUDE ELEM                                           000043
00112      5*      INCLUDE STAT                                           000043
00114      6*      INCLUDE FORC                                           000043
00116      7*      INCLUDE CONN                                           000043
00120      8*      J=ID(I)                                                000043
00121      9*      DK=PP(1,J,2)                                           000045
00122     10*      K=IA(I,1)                                              000051
00123     11*      L=IA(I,2)                                              000053
00124     12*      M=IP(I,1,1)                                            000055
00125     13*      N=IP(I,2,1)                                            000057
00126     14*      AX=0.0                                                 000061
00127     15*      AY=0.0                                                 000062
00130     16*      BX=XC(K,M)                                             000063
```

```
00131     17*          BY=YC(K,M)                                                                    000071
00132     18*          IJKLM = 4                                                                     000075
00133     19*          CALL RKIN(AX,AY,BX,BY,K,IJKLM,XM,YM,XDM,YDM,XDDM,YDDM)          K + M         000077
00134     20*          BX=XC(L,N)                                                                    000115
00135     21*          BY=YC(L,N)                                                                    000124
00136     22*          CALL RKIN(AX,AY,BX,BY,L,IJKLM,XN,YN,XDN,YDN,XDDN,YDDN)          L + N         000130
00137     23*          DX=XN-XM                                                                      000146
00140     24*          DY=YN-YM                                                                      000151
00141     25*          DL=SQRT(DX*DX+DY*DY)                                                          000154
00142     26*          SA=DY/DL                                                                      000164
00143     27*          CA=DX/DL                                                                      000167
00144     28*          FA=DK*((XDN-XDM)*CA+(YDN-YDM)*SA)                                             000172
00145     29*          FI(K,I,1)=-FA*CA                                                              000203
00146     30*          FI(K,I,2)=-FA*SA                                                              000211
00147     31*          FI(K,I,3)=-FI(K,I,1)*(YM-YP(K))+FI(K,I,2)*(XM-XP(K))                          000216
00150     32*          FI(L,I,1)=-FI(K,I,1)                                                          000232
00151     33*          FI(L,I,2)=-FI(K,I,2)                                                          000240
00152     34*          FI(L,I,3)=-FI(L,I,1)*(YN-YP(L))+FI(L,I,2)*(XN-XP(L))                          000244
00153     35*          RETURN                                                                        000260
00154     36*          END                                                                           000331

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS .TRSP
FOR SE38=06/02/77-14:35:51 (.0)

   SUBROUTINE TRSP       ENTRY POINT 000112

   STORAGE USED: CODE(1) 000130: DATA(0) 000032: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    ELEM    013117
    0004    STAT    000660
    0005    FORC    063140
    0006    CONN    002720

   EXTERNAL REFERENCES (BLOCK, NAME)

    0007    NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0004    000600 AA       0006    001760 AC       0004 R 000140 AP       0004    000360 AV       0005    062720 F
    0005    062500 FE       0005 R 000000 FI       0000 R 000002 FL       0003 I 000551 IA       0006    000000 IC
    0003 I 000265 ID        0003    000000 IE       0000    000006 INJPS   0003    001605 IP       0003    000001 IT
    0000 I 000000 J         0000 I 000003 K         0000 I 000004 L        0003 R 004731 PP       0000 R 000001 SK
    0004    000440 XA       0006    000060 XC       0004    000000 XP      0004    000220 XV       0004    000520 YA
    0006    001020 YC       0004    000060 YP       0004    000300 YV

00101      1*           COMPILER(XM=1),(ADR=IND)                                                    000024
00103      2*           SUBROUTINE TRSP(I)                                                          000024
00105      3*           INCLUDE PARM                                                                000024
00110      4*           INCLUDE ELEM                                                                000024
00112      5*           INCLUDE STAT                                                                000024
00114      6*           INCLUDE FORC                                                                000024
00116      7*           INCLUDE CONN                                                                000024
00120      8*           J=ID(I)                                                                     000024
00121      9*           SK=PP(1,J,3)                                                                000026
00122     10*           FL=PP(2,J,3)*.017453293                                                     000032
00123     11*           K=IA(I,1)                                                                   000035
00124     12*           L=IA(I,2)                                                                   000037
00125     13*           FI(K,I,1)=0.0                                                               000041
00126     14*           FI(K,I,2)=0.0                                                               000045
00127     15*           FI(K,I,3)=-SK*(AP(L)-AP(K)-FL)                                              000050
00130     16*           FI(L,I,1)=0.0                                                               000061
00131     17*           FI(L,I,2)=0.0                                                               000067
00132     18*           FI(L,I,3)=-FI(K,I,3)                                                        000070
00133     19*           RETURN                                                                      000072
00134     20*           END                                                                         000127

       END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS .TROP
FOR SE38-06/02/77-14:35:56 (,0)

   SUBROUTINE TROP      ENTRY POINT 000106

   STORAGE USED: CODE(1) 000124; DATA(0) 000030; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   ELEM   013117
    0004   STAT   000660
    0005   FORC   063140
    0006   CONN   002720

   EXTERNAL REFERENCES (BLOCK, NAME)

    0007   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0004   000600 AA      0006   001760 AC      0004   000140 AP      0004 R 000360 AV      0000 R 000001 DK
    0005   062720 F       0005   062500 FE      0005 R 000000 FI      0003 I 000551 IA      0006   000000 IC
    0003 I 000265 ID      0003   000000 IE      0000   000004 INJP$   0003   001605 IP      0003   000001 IT
    0000 I 000000 J       0000 I 000002 K       0000 I 000003 L       0003 R 004731 PP      0004   000440 XA
    0006   000060 XC      0004   000000 XP      0004   000220 XV      0004   000520 YA      0006   001020 YC
    0004   000060 YP      0004   000300 YV


00101       1*        COMPILER(XM=1),(ADR=IND)                        000024
00103       2*        SUBROUTINE TROP(I)                              000024
00105       3*        INCLUDE PARM                                    000024
00110       4*        INCLUDE ELEM                                    000024
00112       5*        INCLUDE STAT                                    000024
00114       6*        INCLUDE FORC                                    000024
00116       7*        INCLUDE CONN                                    000024
00120       8*        J=ID(I)                                         000024
00121       9*        DK=PP(1,J,4)                                    000026
00122      10*        K=IA(I,1)                                       000032
00123      11*        L=IA(I,2)                                       000034
00124      12*        FI(K,I,1)=0.0                                   000036
00125      13*        FI(K,I,2)=0.0                                   000042
00126      14*        FI(K,I,3)=-DK*(AV(L)-AV(K))                     000045
00127      15*        FI(L,I,1)=0.0                                   000055
00130      16*        FI(L,I,2)=0.0                                   000063
00131      17*        FI(L,I,3)=-FI(K,I,3)                            000064
00132      18*        RETURN                                          000066
00133      19*        END                                             000123

    END OF COMPILATION:       NO DIAGNOSTICS.
```

```
@FOR.IS .MEAN
FOR SE38=06/02/77-14:36:03 (.0)

   SUBROUTINE BEAM       ENTRY POINT 001053

   STORAGE USED: CODE(1) 001102: DATA(0) 000240: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    ELEM    013117
    0004    STAT    000660
    0005    FORC    063140
    0006    CONN    002720
    0007    SAVE    030470

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010    RKIN
    0011    STRS
    0012    SQRT
    0013    ASIN
    0014    NERR33

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000401 165G      0001    000430 174G      0001    000554 222G      0001    000573 227G      0004    000600 AA
    0006    001760 AC        0004 R  000140 AP        0004 R  000340 AV        0000 R  000053 AX        0000 R  000054 AY
    0000 R  000122 BI        0000 R  000123 BJ        0000 R  000055 BX        0000 R  000056 BY        0000 R  000126 CA
    0000 R  000106 CI        0000 R  000107 CJ        0000 R  000124 DA        0000 R  000130 DAMP      0000 R  000132 DAMPA
    0000 R  000131 DAMPY     0000 R  000100 DD        0000 R  000036 DH        0000 R  000101 DL        0000 R  000076 DX
    0000 R  000077 DY        0000 R  000117 ET        0000 R  000120 EJ        0000 R  000105 EP        0005    062720 F
    0000 R  000133 FD        0000 R  000135 FDA       0000 R  000134 FDY       0005    062500 FE        0005 R  000000 FI
    0000 R  000110 HH        0003 I  000551 IA        0006    000000 IC        0003 I  000265 ID        0003    000000 IE
    0000 I  000057 IJKLM     0000    000146 INJPS     0003 I  001605 IF        0003    000001 II        0000 I  000046 J
    0000 I  000127 JDJ       0000 I  000112 JI        0000 I  000113 JJ        0000 I  000121 JK        0000 I  000111 JO
    0000 I  000114 JS        0000 I  000115 JSI       0000 I  000116 JY        0000 I  000047 K         0000 I  000050 L
    0000 I  000051 M         0000 I  000052 N         0000 R  000103 PI        0000 R  000104 PJ        0003 R  004731 PP
    0000 R  000125 SA        0000 R  000012 SI        0000 R  000024 SJ        0007 R  000000 SV        0000 R  000102 TA
    0000 R  000136 VI        0000 R  000137 VJ        0004    000440 XA        0006 R  000060 XC        0000 R  000074 XD
    0000 R  000064 XDDM      0000 R  000072 XDDN      0000 R  000062 XDM       0000 R  000070 XDN       0000 R  000060 XM
    0000 R  000066 XN        0004 R  000000 XP        0004    000220 XV        0000 R  000000 Y         0004    000520 YA
    0006 R  001020 YC        0000 R  000075 YD        0000 R  000065 YDDM      0000 R  000073 YDDN      0000 R  000063 YDM
    0000 R  000071 YDN       0000 R  000061 YM        0000 R  000067 YN        0004 R  000060 YP        0004    000300 YV


00101       1*          COMPILER(XM=1),(ADR=IND)                                                    000062
00103       2*          SUBROUTINE BEAM(I)                                                          000062
00105       3*          INCLUDE PARM                                                                000062
00110       4*          INCLUDE ELEM                                                                000062
00112       5*          INCLUDE STAT                                                                000062
00114       6*          INCLUDE FORC                                                                000062
```

```
00116          INCLUDE CONN                                                              000062
00120          INCLUDE SAVE                                                              000062
00122          DIMENSION Y(10),SI(10),SJ(10)                                             000062
00123          DIMENSION DH(8)                                                           000062
00124          J=ID(I)                                                                   000062
00125          K=IA(I,1)                                                                 000071
00126          L=IA(I,2)                                                                 000073
00127          M=IP(I,1,1)                                                               000075
00130          N=IP(I,2,1)                                                               000077
00131          AX=0.0                                                                    000101
00132          AY=0.0                                                                    000102
00133          BX=XC(K,M)                                                                000103
00134          BY=YC(K,M)                                                                000111
00135          IJKLM = 4                                                                 000115
00136          CALL RKIN(AX,AY,BX,BY,K,IJKLM,XM,YM,XDM,YDM,XDDM,YDDM)         K + M      000117
00137          BX=XC(L,N)                                                                000136
00140          BY=YC(L,N)                                                                000145
00141          CALL RKIN(AX,AY,BX,BY,L,IJKLM,XN,YN,XDN,YDN,XDDN,YDDN)         L + N      000151
00142          XD=XN-XM                                                                  000167
00143          YD=YN-YM                                                                  000172
00144          DX=XD-SV(1,I)                                                             000175
00145          DY=YD-SV(2,I)                                                             000200
00146          DD=2.*(SV(1,I)*DX+SV(2,I)*DY)+DX*DX+DY*DY                                 000202
00147          DL=SQRT(SV(5,I)**2+DD)                                                    000215
00150          TA=(SV(1,I)*DY-SV(2,I)*DX)/(SV(5,I)*DL)                                   000225
00151          IF(ABS(TA).GT..01) TA=ASIN(TA)                                            000236
00153          PI=AP(K)-SV(3,I)-TA                                                       000246
00154          PJ=AP(L)-SV(4,I)-TA                                                       000255
00155          EP=DD/((SV(5,I)+DL)*SV(5,I))                                              000263
00156          CI=-2.*(2.*PI+PJ)/SV(5,I)                                                 000271
00157          CJ=2.*(PI+2.*PJ)/SV(5,I)                                                  000276
00157    C     HH IS THE POSITION OF THE NEUTRAL AXIS -- FROM THE TOP OF THE BEAM        000276
00160          HH=SV(6,I)                                                                000304
00161          DX=0.0                                                                    000306
00162          JO=0                                                                      000307
00163          JI=IFIX(PP(5,J,5)+.1)                                                     000310
00164          DO 1 JJ=1,JI                                                              000364
00167          JS=IFIX(PP(5+3*JJ,J,5)+.1)                                                000401
00170          IF(JJ .EQ. JI) JS=JS+1                                                    000412
00170    C     DH IS THE HEIGHT OF A BLOCK DIVISION WITHIN A BLOCK SECTION               000412
00172          DH(JJ)=PP(3+3*JJ,J,5)/PP(5+3*JJ,J,5)                                      000420
00173          DO 5 JSI=1,JS                                                             000423
00176          JY=JO + JSI                                                               000430
00176    C*****Y IS THE DISTANCE FROM THE N.A. OF ANY DIVISION LINE BOUNDRY              000430
00177          Y(JY)=HH-FLOAT(JSI-1)*DH(JJ)-DX                                           000433
00177    C*****EI IS THE STRAIN AT END *I*                                               000433
00200          EI=EP-Y(JY)*CI                                                            000445
00200    C*****EJ IS THE STRAIN AT END *J*                                               000445
00201          EJ=EP-Y(JY)*CJ                                                            000450
00202          JK=1+6*JY                                                                 000454
00203          CALL STRS(EI,PP(1,J,5),SV(JK,I),SI(JY))                                   000457
00204          JK=JK+3                                                                   000500
00205          CALL STRS(EJ,PP(1,J,5),SV(JK,I),SJ(JY))                                   000503
00206        5 CONTINUE                                                                  000525
00210          DX=DX+PP(3+3*JJ,J,5)                                                      000525
00211          JO=JO+JS                                                                  000530
00212        1 CONTINUE                                                                  000541
```

```
00214          PI=0.0                                                                   000541
00215          PJ=0.0                                                                   000542
00216          BI=0.0                                                                   000543
00217          BJ=0.0                                                                   000544
00220          JO=0                                                                     000545
00221          DO 2 JJ=1,JI                                                             000554
00221    C*****DA IS THE AREA OF A BLOCK DIVISION                                       000554
00224          DA=DH(JJ)*PP(4+3*JJ,J,5)                                                 000554
00225          JS=IFIX(PP(5+3*JJ,J,5)+.1)                                               000556
00226          DO 6 JSI=1,JS                                                            000567
00231          JY=JO + JSI                                                              000573
00232          PI=PI+(SI(JY)+SI(JY+1))*DA                                               000576
00233          PJ=PJ+(SJ(JY)+SJ(JY+1))*DA                                               000604
00234          BI=BI+(SI(JY)*(2.*Y(JY)+Y(JY+1))+SI(JY+1)*(Y(JY)+2.*Y(JY+1)))*DA         000611
00235          BJ=BJ+(SJ(JY)*(2.*Y(JY)+Y(JY+1))+SJ(JY+1)*(Y(JY)+2.*Y(JY+1)))*DA         000627
00236        6 CONTINUE                                                                 000641
00240          JO=JO+JS                                                                 000641
00241        2 CONTINUE                                                                 000647
00243          SA = YD/DL                                                               000647
00244          CA = XD/DL                                                               000652
00245          JDJ = 6 + 3*JI                                                           000655
00246          DAMP = PP(JDJ,J,5)                                                       000661
00247          DAMPY = PP(JDJ+1,J,5)                                                    000667
00250          DAMPA = PP(JDJ+2,J,5)                                                    000671
00251          FD = DAMP*((XDN-XDM)*CA + (YDN-YDM)*SA)                                  000673
00252          FDY = DAMPY*((YDN-YDM)*CA-(XDN-XDM)*SA)                                  000700
00253          FDA = DAMPA*(AV(L)-AV(K))                                                000707
00253    C*****PJ IS THE FORCE AT END *J*                                               000707
00254          PJ=(PI+PJ)/4.                                                            000715
00255          PJ = PJ + FD                                                             000721
00255    C*****PI IS THE FORCE AT END *I*                                               000721
00256          PI=-PJ                                                                   000723
00256    C*****BI IS THE MOMENT AT END *I*                                              000723
00257          BI=BI/6.                                                                 000724
00257    C*****BJ IS THE MOMENT AT END *J*                                              000724
00260          BJ=-BJ/6.                                                                000727
00261          VI=(BI+BJ)/DL                                                            000731
00262          BI = BI - FDA                                                            000734
00263          BJ = BJ + FDA                                                            000736
00264          VI = VI - FDY                                                            000740
00265          VJ=-VI                                                                   000742
00266          FI(K,I,1)=PI*CA-VI*SA                                                    000743
00267          FI(K,I,2)=PI*SA+VI*CA                                                    000750
00270          FI(K,I,3)=BI-FI(K,I,1)*(YM-YP(K))+FI(K,I,2)*(XM-XP(K))                   000757
00271          FI(L,I,1)=PJ*CA-VJ*SA                                                    000772
00272          FI(L,I,2)=PJ*SA+VJ*CA                                                    001000
00273          FI(L,I,3)=BJ-FI(L,I,1)*(YN-YP(L))+FI(L,I,2)*(XN-XP(L))                   001007
00274          RETURN                                                                   001022
00275          END                                                                      001101

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS .STRS
FOR SE3B-06/02/77-14:36:22 (,0)

   SUBROUTINE STRS      ENTRY POINT 000106

   STORAGE USED: CODE(1) 000124; DATA(0) 000017; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000071 1L          0000 R 000001 ASIGN      0000 R 000002 ASIGY      0000 R 000000 DEPS       0000 R 000004 DEPSE
    0000 R 000003 DSIGE       0000   000005 INJP$


00101      1*          COMPILER(XM=1),(ADR=IND)                                                        000001
00103      2*          SUBROUTINE STRS(EE,PP,SV,SGNW)                                                  000001
00105      3*          DIMENSION PP(1),SV(1)                                                           000001
00106      4*          DEPS=EE-SV(1)                                                                   000001
00107      5*          SGNW=SV(2)+PP(1)*DEPS                                                           000004
00110      6*          IF(PP(3).EQ.0.) GO TO 1                                                         000007
00112      7*          ASIGN=ABS(SGNW)                                                                 000011
00113      8*          ASIGY=ABS(SV(3))                                                                000013
00114      9*          IF(ASIGN.LE.ASIGY) GO TO 1                                                      000015
00116     10*          DSIGE=ASIGY-ABS(SV(2))                                                          000020
00117     11*          IF(SGNW*SV(2).LT.0.) DSIGE=ASIGY+ABS(SV(2))                                     000023
00121     12*          DEPSE=SIGN(DSIGE/PP(1),DEPS)                                                    000032
00122     13*          SGNW=SIGN(ASIGY,SGNW)+PP(2)*(DEPS-DEPSE)                                        000040
00123     14*          IF(PP(4).NE.0..AND.ABS(SGNW).GT.PP(4)) SGNW=SIGN(PP(4),SGNW)                    000047
00125     15*          SV(3)=SGNW                                                                      000066
00126     16*        1 CONTINUE                                                                        000071
00127     17*          SV(1)=EE                                                                        000071
00130     18*          SV(2)=SGNW                                                                      000072
00131     19*          RETURN                                                                          000074
00132     20*          END                                                                             000123

     END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS .PINN
FOR SE38-06/02/77-14:36:27 (,0)

   SUBROUTINE PINN      ENTRY POINT 000134

   STORAGE USED: CODE(1) 000153; DATA(0) 000036; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    ELEM    013117
    0004    STAT    000660
    0005    FORC    063140
    0006    CONN    002720
    0007    SAVE    030470

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010    NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0004    000600 AA        0006   001760 AC        0004   000140 AP        0004 R 000360 AV        0000 R 000003 DV
    0005    062720 F         0005   062500 FE        0005 R 000000 FI        0000 R 000004 FM        0003 I 000551 IA
    0006    000000 IC        0003 I 000265 ID        0003   000000 IE        0000   000010 INJP$     0003   001605 IP
    0003    000001 IT        0000 I 000000 J         0000 I 000001 K         0000 I 000002 L         0003 R 004731 PP
    0007 R  000000 SV        0004   000440 XA        0006   000060 XC        0004   000000 XP        0004   000220 XV
    0004    000520 YA        0006   001020 YC        0004   000060 YP        0004   000300 YV


00101      1*          COMPILER(XM=1),(ADR=IND)                               000032
00103      2*          SUBROUTINE PINN(I)                                     000032
00105      3*          INCLUDE PARM                                           000032
00110      4*          INCLUDE ELEM                                           000032
00112      5*          INCLUDE STAT                                           000032
00114      6*          INCLUDE FORC                                           000032
00116      7*          INCLUDE CONN                                           000032
00120      8*          INCLUDE SAVE                                           000032
00122      9*          J=ID(I)                                                000032
00123     10*          K=IA(I,1)                                              000034
00124     11*          L=IA(I,2)                                              000036
00125     12*          DV=AV(L)-AV(K)                                         000040
00126     13*          FM=SIGN(1.0,DV)                                        000047
00127     14*          IF(ABS(DV).LT..01) FM=DV/.01                           000053
00131     15*          FI(K,I,1)=0.0                                          000062
00132     16*          FI(K,I,2)=0.0                                          000067
00133     17*          FI(K,I,3)=-PP(1,J,6)*SV(1,I)*FM                        000072
00134     18*          FI(L,I,1)=0.0                                          000101
00135     19*          FI(L,I,2)=0.0                                          000110
00136     20*          FI(L,I,3)=-FI(K,I,3)                                   000111
00137     21*          RETURN                                                 000113
00140     22*          END                                                    000152

     END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
#FOR.IS .SLDR
FOR SE38-06/02/77-14:36:34 (.0)

   SUBROUTINE SLDR      ENTRY POINT 000337

   STORAGE USED: CODE(1) 000364: DATA(0) 000073: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   CONN   002720
    0004   ELEM   013117
    0005   STAT   000660
    0006   SAVE   030470
    0007   FORC   063140

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010   SIN
    0011   COS
    0012   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0005   000600 AA        0003 R 001760 AC        0005 R 000140 AP        0005 R 000360 AV        0000 R 000006 C
    0000 R 000010 CF        0000 R 000011 CS        0007   062720 F         0007   062500 FE        0000 R 000012 FF
    0007 R 000000 FI        0000 R 000013 FM        0004 I 000551 IA        0003   000000 IC        0004 I 000265 ID
    0004   000000 IE        0000   000020 INJP$     0004 I 001605 IP        0004   000001 IT        0000 I 000000 J
    0000 I 000001 K         0000 I 000002 L         0000 I 000003 M         0000 I 000004 N         0004 R 004731 PP
    0000 R 000005 S         0006 R 000000 SV        0000 R 000007 V         0005   000440 XA        0003   000060 XC
    0005 R 000000 XP        0005 R 000220 XV        0005   000520 YA        0003   001020 YC        0005 R 000060 YP
    0005 R 000300 YV

00101      1*          COMPILER(XM=1),(ADR=IND)                                                    000062
00103      2*          SUBROUTINE SLDR(I)                                                          000062
00105      3*          INCLUDE PARM                                                                000062
00110      4*          INCLUDE CONN                                                                000062
00112      5*          INCLUDE ELEM                                                                000062
00114      6*          INCLUDE STAT                                                                000062
00116      7*          INCLUDE SAVE                                                                000062
00120      8*          INCLUDE FORC                                                                000062
00122      9*          J=ID(I)                                                                     000062
00123     10*          K=IA(I,1)                                                                   000066
00124     11*          L=IA(I,2)                                                                   000070
00125     12*          M=IP(1,1,1)                                                                 000072
00126     13*          N=IP(1,2,1)                                                                 000074
00127     14*          S=SIN(AP(K)+.01745329*AC(K,M))                                              000076
00130     15*          C=COS(AP(K)+.01745329*AC(K,M))                                              000114
00131     16*          V=-(XV(L)-XV(K))*C+(XP(L)-XP(K))*AV(K)*S-(YV(L)-YV(K))*S-(YP(L)-YP          000120
00131     17*         1(K))*AV(K)*C                                                                000120
00132     18*          CF=PP(3,J,7)                                                                000200
```

```
00133      19*            CS=CF*2.                                                    000203
00134      20*            IF(CS.LE. 1.0) CS = 1.                                      000205
00136      21*            IF(ABS(V/CS) .LT. 1.0) CF = CF*ABS(V/CS)                    000213
00140      22*            V=SIGN(1.0,V)                                               000246
00141      23*            FF=CF*SV(1,I)*V                                             000252
00142      24*            FM=CF*SV(2,I)*PP(2,J,7)*V/2.0                               000256
00143      25*            FI(K,I,1)=+FF*C                                             000265
00144      26*            FI(K,I,2)=+FF*S                                             000270
00145      27*            FI(K,I,3)=-FM+FF*SV(3,I)                                    000274
00146      28*            FI(L,I,1)=-FF*C                                             000300
00147      29*            FI(L,I,2)=-FF*S                                             000302
00150      30*            FI(L,I,3)=+FM-FF*SV(4,I)                                    000304
00151      31*            RETURN                                                      000310
00152      32*            END                                                         000363

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR,IS ,SPIN
FOR SE3B=06/02/77-14:36:41 (.0)

   SUBROUTINE SPIN       ENTRY POINT 000352

   STORAGE USED: CODE(1) 000375: DATA(0) 000073: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   CONN   002720
    0004   ELEM   013117
    0005   STAT   000660
    0006   SAVE   030470
    0007   FORC   063140

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010   SIN
    0011   COS
    0012   NERR3S

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0005   000600 AA        0003 R 001760 AC        0005 R 000140 AP        0005 R 000360 AV        0000 R 000006 CK
    0000 R 000010 CL        0007   062720 F         0007   062500 FE        0000 R 000022 FF        0007 R 000000 FI
    0004 I 000551 IA        0003   000000 IC        0004 I 000265 ID        0004   000000 IE        0000   000025 INJPS
    0004 I 001605 IP        0004   000001 IT        0000 I 000000 J         0000 I 000001 K         0000 I 000002 L
    0000 I 000003 M         0000 I 000004 N         0004 R 004731 PP        0000 R 000005 SK        0000 R 000007 SL
    0006 R 000000 SV        0000 R 000021 V         0005   000440 XA        0003 R 000060 XC        0000 R 000011 XKM
    0000 R 000015 XKMD      0000 R 000013 XML       0000 R 000017 XMLD      0005 R 000000 XP        0005 R 000220 XV
    0005   000520 YA        0003 R 001020 YC        0000 R 000012 YKM       0000 R 000016 YKMD      0000 R 000014 YML
    0000 R 000020 YMLD      0005 R 000060 YP        0005 R 000300 YV

00101      1*          COMPILER(XM=1),(ADR=IND)                                    000070
00103      2*          SUBROUTINE SPIN(I)                                          000070
00105      3*          INCLUDE PARM                                                000070
00110      4*          INCLUDE CONN                                                000070
00112      5*          INCLUDE ELEM                                                000070
00114      6*          INCLUDE STAT                                                000070
00116      7*          INCLUDE SAVE                                                000070
00120      8*          INCLUDE FORC                                                000070
00122      9*          J=ID(I)                                                     000070
00123     10*          K=IA(I,1)                                                   000072
00124     11*          L=IA(I,2)                                                   000074
00125     12*          M=IP(I,1,1)                                                 000076
00126     13*          N=IP(I,2,1)                                                 000100
00127     14*          SK=SIN(AP(K))                                               000102
00130     15*          CK=COS(AP(K))                                               000112
00131     16*          SL=SIN(AP(L)+.01745329*AC(L,N))                             000120
00132     17*          CL=COS(AP(L)+.01745329*AC(L,N))                             000130
```

```
00133      18*            XKM=YC(K,M)*SK-XC(K,M)*CK                                    000142
00134      19*            YKM=-YC(K,M)*CK-XC(K,M)*SK                                   000156
00135      20*            XML=XP(K)-XP(L)-XKM                                          000164
00136      21*            YML=YP(K)-YP(L)-YKM                                          000175
00137      22*            XKMD=AV(K)*(YC(K,M)*CK+XC(K,M)*SK)                           000206
00140      23*            YKMD=AV(K)*(YC(K,M)*SK-XC(K,M)*CK)                           000213
00141      24*            XMLD=XV(K)-XV(L)-XKMD                                        000216
00142      25*            YMLD=YV(K)-YV(L)-YKMD                                        000226
00143      26*            V=-YMLD*SL-YML*AV(L)*CL-XMLD*CL+XML*AV(L)*SL                 000236
00144      27*            FF=PP(1,J,8)*SV(1,I)*SIGN(1.0,V)                             000252
00145      28*            FI(K,I,1)=-FF*CL                                             000264
00146      29*            FI(K,I,2)=-FF*SL                                             000274
00147      30*            FI(K,I,3)=-FF*(YKM*CL-XKM*SL)                                000277
00150      31*            FI(L,I,1)=+FF*CL                                             000306
00151      32*            FI(L,I,2)=+FF*SL                                             000315
00152      33*            FI(L,I,3)=-FF*(YML*CL-XML*SL)                                000316
00153      34*            RETURN                                                       000325
00154      35*            END                                                          000374

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
#FOR.IS .DSLD
FOR SE38-06/02/77-14:36:48 (.0)

   SUBROUTINE DSLD       ENTRY POINT 001016

   STORAGE USED: CODE(1) 001044: DATA(0) 000161: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   CONN   002720
    0004   ELEM   013117
    0005   SAVE   030470
    0006   FORC   063140
    0007   STAT   000660

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010   RKIN
    0011   SIN
    0012   COS
    0013   NWDUS
    0014   NIO2S
    0015   NSTOPS
    0016   NERR3S

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000527 1L        0001    000345 161G      0000    000066 3F        0001    000575 4L        0007    000600 AA
    0003 R  001760 AC        0007 R  000140 AP        0007    000360 AV        0000 R  000013 AX        0000 R  000014 AY
    0000 R  000015 BX        0000 R  000016 BY        0000 R  000035 CA        0000 R  000037 CAK       0000 R  000041 CAL
    0000 R  000005 DK        0000 R  000010 DL        0000 R  000047 D1        0000 R  000050 D2        0000 R  000051 D3
    0000 R  000052 D4        0006    062720 F         0006    062500 FE        0006 R  000000 F1        0000 R  000007 FK
    0000 R  000012 FL        0000 R  000064 FM1       0000 R  000065 FM2       0000 R  000053 FN1       0000 R  000054 FN2
    0000 R  000055 FN3       0000 R  000056 FN4       0000 R  000062 F1        0000 R  000063 F2        0004 I  000551 IA
    0003    000000 IC        0004 I  000265 ID        0004    000000 IE        0000 I  000046 II        0000 I  000017 IJKLM
    0000    000100 INJPS     0004 I  001605 IP        0004    000001 IT        0000 I  000057 I1        0000 I  000060 I2
    0000 I  000061 I3        0000 I  000000 J         0000 I  000001 K         0000 I  000002 L         0000 I  000003 M
    0000 I  000004 N         0004 R  004731 PP        0000 R  000034 SA        0000 R  000036 SAK       0000 R  000040 SAL
    0000 R  000044 SK        0000 R  000045 SL        0005 R  000000 SV        0000 R  000042 VK        0000 R  000043 VL
    0000 R  000006 WK        0000 R  000011 WL        0007    000440 XA        0003 R  000060 XC        0000 R  000024 XDDM
    0000 R  000032 XDDN      0000 R  000022 XDM       0000 R  000030 XDN       0000 R  000020 XM        0000 R  000026 XN
    0007 R  000000 XP        0007    000220 XV        0007    000520 YA        0003 R  001020 YC        0000 R  000025 YDDM
    0000 R  000033 YDDN      0000 R  000023 YDM       0000 R  000031 YDN       0000 R  000021 YM        0000 R  000027 YN
    0007 R  000060 YP        0007    000300 YV

00101       1*          COMPILER(XM=1),(ADR=IND)                                                    000057
00103       2*          SUBROUTINE DSLD(I)                                                          000057
00105       3*          INCLUDE PARM                                                                000057
00110       4*          INCLUDE CONN                                                                000057
00112       5*          INCLUDE ELEM                                                                000057
```

```
00114      6*          INCLUDE SAVE                                                                    000057
00116      7*          INCLUDE FORC                                                                    000057
00120      8*          INCLUDE STAT                                                                    000057
00122      9*          J=ID(I)                                                                         000057
00123     10*          K=IA(1,1)                                                                       000063
00124     11*          L=IA(I,2)                                                                       000065
00125     12*          M=IP(I,1,1)                                                                     000067
00126     13*          N=IP(I,2,1)                                                                     000071
00127     14*          DK=PP(1,J,9)                                                                    000073
00130     15*          WK=PP(2,J,9)                                                                    000075
00131     16*          FK=PP(3,J,9)                                                                    000077
00132     17*          DL=PP(4,J,9)                                                                    000101
00133     18*          WL=PP(5,J,9)                                                                    000103
00134     19*          FL=PP(6,J,9)                                                                    000105
00135     20*          AX=0.0                                                                          000107
00136     21*          AY=0.0                                                                          000110
00137     22*          BX=XC(K,M)                                                                      000111
00140     23*          BY=YC(K,M)                                                                      000121
00141     24*          IJKLM = 4                                                                       000125
00142     25*          CALL RKIN(AX,AY,BX,BY,K,IJKLM,XM,YM,XDM,YDM,XDDM,YDDM)                          000127
00143     26*          BX=XC(L,N)                                                                      000145
00144     27*          BY=YC(L,N)                                                                      000154
00145     28*          CALL RKIN(AX,AY,BX,BY,L,IJKLM,XN,YN,XDN,YDN,XDDN,YDDN)                          000167
00146     29*          SA=SIN(AP(K)+.01745329*AC(K,M))                                                 000206
00147     30*          CA=COS(AP(K)+.01745329*AC(K,M))                                                 000216
00150     31*          SAK=SIN(AP(K))                                                                  000222
00151     32*          CAK=COS(AP(K))                                                                  000230
00152     33*          SAL=SIN(AP(L))                                                                  000236
00153     34*          CAL=COS(AP(L))                                                                  000246
00154     35*          VK=(XDN-XDM)*CA+(YDN-YDM)*SA                                                    000260
00155     36*          VL=-(XDN-XDM)*SA+(YDN-YDM)*CA                                                   000266
00156     37*          SK=SIGN(1.0,VK)                                                                 000274
00157     38*          SL=SIGN(1.0,VL)                                                                 000300
00160     39*          DO 2 II=1,16                                                                    000313
00163     40*          D1=2.*DL+FL*SL*((SV(4,I)-SV(5,I))*(DK-FK*SK*WL-FK*FL*SK*SL*WK*SV(3              000367
          1,I))-(SV(4,I)+SV(5,I))*FK*SK*DL*SV(3,I))                                                 000367
00164     42*          D2=2.*DL+FL*SL*((SV(4,I)-SV(5,I))*(-DK-FK*SK*WL-FK*FL*SK*SL*WK*SV(              000401
          12,I))-(SV(4,I)+SV(5,I))*FK*SK*DL*SV(2,I))                                                000401
00165     44*          D3=-2.*DL+FK*SK*((SV(2,I)-SV(3,I))*(-DL-FL*WK*SL-FK*FL*SK*SL*WL*SV              000413
          1(5,I))+(SV(2,I)+SV(3,I))*FL*SL*DK*SV(5,I))                                               000413
00166     46*          D4=-2.*DL+FK*SK*((SV(2,I)-SV(3,I))*(DL-FL*WK*SL-FK*FL*SK*SL*WL*SV(              000425
          14,I))+(SV(2,I)+SV(3,I))*FL*SL*DK*SV(4,I))                                                000425
00167     48*          FN1= SV(1,1)*D1/(D1+D2)                                                         000437
00170     49*          FN2=-SV(1,1)*D2/(D1+D2)                                                         000444
00171     50*          FN3= SV(1,I)*D3/(D1+D2)                                                         000451
00172     51*          FN4=-SV(1,I)*D4/(D1+D2)                                                         000456
00173     52*          IF(SV(2,I).NE.SIGN(1.0,FN1).OR.SV(3,I).NE.SIGN(1.0,FN2).OR.SV(4,I)              000463
00173     53*         1.NE.SIGN(1.0,FN3).OR.SV(5,I).NE.SIGN(1.0,FN4)) GO TO 1                          000463
00175     54*          GO TO 4                                                                         000525
00176     55*        1 CONTINUE                                                                        000527
00177     56*          I1=MOD(II,8)                                                                    000527
00200     57*          I2=MOD(II,4)                                                                    000532
00201     58*          I3=MOD(II,2)                                                                    000536
00202     59*          IF(I1.EQ.1) SV(2,I)=-SV(2,I)                                                    000542
00204     60*          IF(I2.EQ.1) SV(3,I)=-SV(3,I)                                                    000547
00206     61*          IF(I3.EQ.1) SV(4,I)=-SV(4,I)                                                    000554
00210     62*          SV(5,I)=-SV(5,I)                                                                000561
```

```
00211     63*          2 CONTINUE                                                                    000565
00213     64*            WRITE(6,3)                                                                  000565
00215     65*          3 FORMAT(14H1ERROR IN DSLD)                                                   000572
00216     66*            STOP                                                                        000572
00217     67*          4 CONTINUE                                                                    000575
00220     68*            F1=-FN3-FN4                                                                 000575
00221     69*            F2=-FN1-FN2                                                                 000577
00222     70*            FM1=-F2*FK*WL*SK/2.0                                                        000602
00223     71*            FM2=-F1*FL*WK*SL/2.0                                                        000607
00224     72*            FI(K,I,1)=-F1*CA-F2*SA                                                      000614
00225     73*            FI(K,I,2)=-F1*SA+F2*CA                                                      000626
00226     74*            FI(K,I,3)=FM1-F1*(.017453293*                                               000636
00226     75*           1                    AC(K,M)*SV(6,I)-YC(K,M)*SV(7,I))+F2*(((XN-XP(K))        000636
00226     76*           1*CAK+(YN-YP(K))*SAK)*SV(7,I)+(-(XN-XP(K))*SAK+(YN-YP(K))*CAK)*SV(6          000636
00226     77*           2,I))                                                                        000636
00227     78*            FI(L,I,1)=F1*CA+F2*SA                                                       000707
00230     79*            FI(L,I,2)=F1*SA-F2*CA                                                       000714
00231     80*            FI(L,I,3)=-FM1-F1*(((XM-XP(L))*CAL+(YM-YP(L))*SAL)*SV(9,I)+(-(XM-X          000717
00231     81*           1P(L))*SAL+(YM-YP(L))*CAL)*SV(8,I))+F2*(.017453293*                          000717
00231     82*           1                                             AC(L,N)*SV(8,I)+YC(L,N)*SV(    000717
00231     83*           29,I))                                                                       000717
00232     84*            RETURN                                                                      000766
00233     85*            END                                                                         001043

     END OF COMPILATION:         NO  DIAGNOSTICS.
```

```
@FOR,IS .RIGD
FOR SE3B-06/02/77-14:37:01 (,0)

   SUBROUTINE RIGD      ENTRY POINT 000053

   STORAGE USED: CODE(1) 000064; DATA(0) 000015; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   ELEM    013117
    0004   FORC    063140

   EXTERNAL REFERENCES (BLOCK, NAME)

    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0004   062720 F        0004   062500 FE       0004 R 000000 FI        0003 I 000551 IA        0003   000265 ID
    0003   000000 IE       0000   000002 INJP$    0003   001605 IP        0003   000001 II       0000 I 000000 K
    0000 I 000001 L        0003   004731 PP


00101       1*             COMPILER(XM=1),(ADR=IND)                                                   000014
00103       2*             SUBROUTINE RIGD(I)                                                         000014
00105       3*             INCLUDE PARM                                                               000014
00110       4*             INCLUDE ELEM                                                               000014
00112       5*             INCLUDE FORC                                                               000014
00114       6*             K=IA(I,1)                                                                  000014
00115       7*             L=IA(I,2)                                                                  000016
00116       8*             FI(K,I,1)=0.0                                                              000020
00117       9*             FI(K,I,2)=0.0                                                              000026
00120      10*             FI(K,I,3)=0.0                                                              000027
00121      11*             FI(L,I,1)=0.0                                                              000030
00122      12*             FI(L,I,2)=0.0                                                              000036
00123      13*             FI(L,I,3)=0.0                                                              000037
00124      14*             RETURN                                                                     000040
00125      15*             END                                                                        000063

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
00125     12*          IF(SV(5,I).GT.SV(15,I).OR.SV(5,I).LT.-SV(15,I)) GO TO 5                                  000071
00127     13*          M=IP(I,1,1)                                                                               000107
00130     14*          N=IP(I,2,1)                                                                               000111
00131     15*          AX=0.0                                                                                    000113
00132     16*          AY=0.0                                                                                    000114
00133     17*          BX=SV(1,I)                                                                                000115
00134     18*          BY=SV(2,I)                                                                                000117
00135     19*          IJKLM = 4                                                                                 000121
00136     20*          CALL RKIN(AX,AY,BX,BY,K,IJKLM,XK,YK,XDK,YDK,XDDK,YDDK)                    K + K           000123
00137     21*          BX=SV(3,I)                                                                                000141
00140     22*          BY=SV(4,I)                                                                                000143
00141     23*          CALL RKIN(AX,AY,BX,BY,L,IJKLM,XL,YL,XDL,YDL,XDDL,YDDL)                    L + L           000145
00142     24*          AN=(AP(K)+AP(L)+.01745329*(AC(K,M)+AC(L,N)-180.))/2.                                      000165
00143     25*          CAN=COS(AN)                                                                               000210
00144     26*          SAN=SIN(AN)                                                                               000214
00145     27*          DH=(XL-XK)*CAN+(YL-YK)*SAN                                                                000220
00146     28*          DV=(YL-YK)*CAN-(XL-XK)*SAN                                                                000231
00147     29*          VH=(XDL-XDK)*CAN+(YDL-YDK)*SAN+(AV(K)+AV(L))*.5*(-(XL-XK)*SAN+(YL-                        000236
00147     30*         1YK)*CAN)                                                                                  000236
00150     31*          IF(DH.GE.0.0) GO TO 1                                                                     000257
00152     32*          IF(VH.GT.0.0) GO TO 2                                                                     000261
00154     33*          FH=SV(6,I)*DH                                                                             000263
00155     34*          IF(DH.LT.SV(7,I)) FH=FH+(SV(8,I)-SV(6,I))*(DH-SV(7,I))                                    000266
00157     35*          IF(DH.LT.SV(9,I)) FH=FH+(SV(10,I)-SV(8,I))*(DH-SV(9,I))                                   000301
00161     36*          FHA=SV(18,I)+(DH-SV(19,I))*SV(10,I)                                                       000314
00162     37*          IF(FHA.GT.FH) FH=FHA                                                                      000321
00164     38*          SV(16,I)=FH                                                                               000327
00165     39*          SV(17,I)=DH                                                                               000331
00166     40*          GO TO 3                                                                                   000333
00167     41*        1 CONTINUE                                                                                  000335
00170     42*          FH=0.0                                                                                    000335
00171     43*          IF(DH.LT.PP(18,J,11)) GO TO 3                                                             000335
00173     44*          DH=DH-PP(18,J,11)                                                                         000344
00174     45*          IF(VH.LT.0.0) GO TO 6                                                                     000347
00176     46*          FH=SV(6,I)*DH                                                                             000352
00177     47*          IF(DH.GT.-SV(7,I)) FH=FH+(SV(8,I)-SV(6,I))*(DH+SV(7,I))                                   000354
00201     48*          IF(DH.GT.-SV(9,I)) FH=FH+(SV(10,I)-SV(8,I))*(DH+SV(9,I))                                  000367
00203     49*          FHA=SV(18,I)+(DH-SV(19,I))*SV(10,I)                                                       000402
00204     50*          IF(FHA.LT.FH) FH=FHA                                                                      000407
00206     51*          SV(16,I)=FH                                                                               000415
00207     52*          SV(17,I)=DH                                                                               000417
00210     53*          GO TO 3                                                                                   000421
00211     54*        6 CONTINUE                                                                                  000423
00212     55*          FH=SV(14,I)*DH                                                                            000423
00213     56*          IF(DH.GT.-SV(13,I)) FH=FH+(SV(12,I)-SV(14,I))*(DH+SV(13,I))                               000425
00215     57*          IF(DH.GT.-SV(11,I)) FH=FH+(SV(10,I)-SV(12,I))*(DH+SV(11,I))                               000440
00217     58*          FHA=SV(16,I)-(SV(17,I)-DH)*SV(10,I)                                                       000453
00220     59*          IF(FHA.GT.FH) FH=FHA                                                                      000460
00222     60*          SV(18,I)=FH                                                                               000466
00223     61*          SV(19,I)=DH                                                                               000470
00224     62*          GO TO 3                                                                                   000472
00225     63*        2 CONTINUE                                                                                  000474
00226     64*          FH=SV(14,I)*DH                                                                            000474
00227     65*          IF(DH.LT.SV(13,I)) FH=FH+(SV(12,I)-SV(14,I))*(DH-SV(13,I))                                000476
00231     66*          IF(DH.LT.SV(11,I)) FH=FH+(SV(10,I)-SV(12,I))*(DH-SV(11,I))                                000511
00233     67*          FHA=SV(16,I)-(SV(17,I)-DH)*SV(10,I)                                                       000524
00234     68*          IF(FHA.LT.FH) FH=FHA                                                                      000531
```

```
@FOR:IS .CPL1
FOR 5E38=06/02/77-14:37:07 (:0)

   SUBROUTINE CPL1       ENTRY POINT 001072

   STORAGE USED: CODE(1) 0011171 DATA(0) 0001141 BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   ELEM   013117
    0004   FORC   063140
    0005   STAT   000660
    0006   CONN   002720
    0007   SAVE   030470

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010   RKIN
    0011   COS
    0012   SIN
    0013   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000335 1L        0001    000474 2L        0001    000544 3L        0001    001013 313L      0001    000637 4L
    0001    001002 414L      0001    001021 5L        0001    000423 6L        0005    000600 AA        0006 R  001760 AC
    0000 R  000026 AN        0005 R  000140 AP        0005 R  000360 AV        0000 R  000005 AX        0000 R  000006 AY
    0000 R  000042 A1        0000 R  000007 BX        0000 R  000010 BY        0000 R  000043 CA        0000 R  000027 CAN
    0000 R  000031 DH        0000 R  000032 DV        0004    062720 F         0004    062500 FE        0000 R  000040 FF
    0000 R  000041 FFV       0000 R  000034 FH        0000 R  000035 FHA       0004 R  000000 FI        0000 R  000037 FV
    0003 I  000551 IA        0006    000000 IC        0003 I  000265 ID        0003    000000 IE        0000 I  000011 IJKLM
    0000    000051 INJP$     0003 I  001605 IP        0003    000001 IT        0000 I  000000 J         0000 I  000001 K
    0000 I  000045 KICK      0000 I  000002 L         0000 I  000003 M         0000 I  000004 N         0003 R  004731 PP
    0000 R  000044 SA        0000 R  000030 SAN       0007 R  000000 SV        0000 R  000033 VH        0000 R  000036 VBD
    0005    000440 XA        0006    000060 XC        0000 R  000016 XDDK      0000 R  000024 XDDL      0000 R  000014 XDK
    0000 R  000022 XDL       0000 R  000012 XK        0000 R  000020 XL        0005 R  000000 XP        0005    000220 XV
    0005    000520 YA        0006    001020 YC        0000 R  000017 YDDK      0000 R  000025 YDDL      0000 R  000015 YDK
    0000 R  000023 YDL       0000 R  000013 YK        0000 R  000021 YL        0005 R  000060 YP        0005    000300 YV


00101       1*          COMPILER(XM=1),(ADR=IND)                                                    000063
00103       2*          SUBROUTINE CPL1(I)                                                          000063
00105       3*          INCLUDE PARM                                                                000063
00110       4*          INCLUDE ELEM                                                                000063
00112       5*          INCLUDE FORC                                                                000063
00114       6*          INCLUDE STAT                                                                000063
00116       7*          INCLUDE CONN                                                                000063
00120       8*          INCLUDE SAVE                                                                000063
00122       9*          J=ID(I)                                                                     000063
00123      10*          K=IA(I,1)                                                                   000065
00124      11*          L=IA(I,2)                                                                   000067
```

```
00236     69*            SV(18,I)=FH                                                                    000537
00237     70*            SV(19,I)=DH                                                                    000541
00240     71*          3 CONTINUE                                                                       000544
00241     72*            VSD=DV-SV(5,I)                                                                 000544
00242     73*            FV=0.0                                                                         000551
00243     74*            IF(VSD.GT.PP(6,J,11)) FV=(VSD-PP(6,J,11))*(PP(5,J,11)*PP(13,J,11))            000553
00243     75*           1/(PP(5,J,11)+PP(13,J,11))                                                      000553
00245     76*            IF(VSD.LT.-PP(14,J,11)) FV=(VSD+PP(14,J,11))*(PP(5,J,11)*PP(13,J,1             000571
00245     77*           11))/(PP(5,J,11)+PP(13,J,11))                                                   000571
00247     78*            FF=ABS(FH)*PP(17,J,11)                                                         000607
00250     79*            IF(FF.GT.ABS(FV)) GO TO 4                                                      000613
00252     80*            FFV=FV                                                                         000617
00253     81*            FV=SIGN(FF,FV)                                                                 000621
00254     82*            SV(5,I)=SV(5,1)+(FFV-FV)*(PP(5,J,11)+PP(13,J,11))/(PP(5,J,11)*PP(1             000625
00254     83*           13,J,11))                                                                       000625
00255     84*          4 CONTINUE                                                                       000637
00256     85*            A1=AP(K)+.01745329*AC(K,M)                                                     000637
00257     86*            CA=COS(A1)                                                                     000650
00260     87*            SA=SIN(A1)                                                                     000654
00261     88*            FI(K,I,1)=-FH*CA+FV*SA                                                         000660
00262     89*            FI(K,I,2)=-FH*SA-FV*CA                                                         000671
00263     90*            FI(K,I,3)=FI(K,I,1)*(YP(K)-YK)+FI(K,I,2)*(XK-XP(K))                            000701
00264     91*            A1=.01745329*(AC(L,N)-180.0)+AP(L)                                             000717
00265     92*            CA=COS(A1)                                                                     000732
00266     93*            SA=SIN(A1)                                                                     000736
00267     94*            FI(L,I,1)=FH*CA-FV*SA                                                          000742
00270     95*            FI(L,I,2)=FH*SA+FV*CA                                                          000753
00271     96*            FI(L,I,3)=FI(L,1,1)*(YP(L)-YL)+FI(L,I,2)*(XL-XP(L))                            000763
00272     97*        414 CONTINUE                                                                       001002
00273     98*            IF(KICK.GT.-2) GO TO 313                                                       001002
00273     99*      C     WRITE(6,797)I,K,L,(FI(K,I,JIJ),JIJ=1,3),(FI(L,I,JIJ),JIJ=1,3),SV(5,I)          001002
00273    100*      C    1,I)                                                                            001002
00273    101*      C 797 FORMAT(3I5,6F15.0,F15.5)                                                       001002
00273    102*      C     WRITE(6,415) DH,DV,VH,FH,FFV,FF,FV                                             001002
00273    103*      C 415 FORMAT(3F15.5,4F15.0)                                                          001002
00275    104*            IF(KICK.EQ.-2) KICK=58                                                         001005
00277    105*        313 CONTINUE                                                                       001013
00300    106*            KICK=KICK-1                                                                    001013
00301    107*            RETURN                                                                         001015
00302    108*          5 CONTINUE                                                                       001021
00303    109*            FI(K,I,1)=0.0                                                                  001021
00304    110*            FI(K,I,2)=0.0                                                                  001025
00305    111*            FI(K,I,3)=0.0                                                                  001030
00306    112*            FI(L,I,1)=0.0                                                                  001031
00307    113*            FI(L,I,2)=0.0                                                                  001040
00310    114*            FI(L,I,3)=0.0                                                                  001041
00311    115*            GO TO 414                                                                      001042
00312    116*            END                                                                            001116

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR*IS *CPL2
FOR SE39=06/02/77-14137:22 (.0)

   SUBROUTINE CPL2       ENTRY POINT 000762

   STORAGE USED: CODE(1) 0010071 DATA(0) 0001211 BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   ELEM   013117
    0004   FORC   063140
    0005   STAT   000660
    0006   CONN   002720
    0007   SAVE   030470

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010   RKIN
    0011   COS
    0012   SIN
    0013   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000307 1L        0001   000402 2L        0001   000433 3L        0001   000703 313L      0001   000533 4L
    0001   000672 414L      0000   000050 415F      0001   000711 5L        0001   000350 6L        0000   000045 797F
    0005   000600 AA        0006 R 001760 AC        0000 R 000026 AN        0005 R 000140 AP        0005 R 000360 AV
    0000 R 000005 AX        0000 R 000006 AY        0000 R 000041 A1        0000 R 000007 BX        0000 R 000010 BY
    0000 R 000042 CA        0000 R 000027 CAN       0000 R 000031 DH        0000 R 000032 DV        0004   062720 F
    0004   062500 FE        0000 R 000037 FF        0000 R 000040 FFV       0000 R 000034 FH        0004 R 000000 FI
    0000 R 000036 FV        0003 I 000551 IA        0006   000000 IC        0003 I 000265 ID        0003   000000 IE
    0000 I 000011 IJKLM     0000   000056 INJP$     0003 I 001605 IP        0003   000001 IT        0000 I 000000 J
    0000 I 000001 K         0000 I 000044 KICK      0000 I 000002 L         0000 I 000003 M         0000 I 000004 N
    0003 R 004731 PP        0000 R 000043 SA        0000 R 000030 SAN       0007 R 000000 SV        0000 R 000033 VH
    0000 R 000035 VSD       0005   000440 XA        0006   000060 XC        0000 R 000016 XDDK      0000 R 000024 XDDL
    0000 R 000014 XDK       0000 R 000022 XDL       0000 R 000012 XK        0000 R 000020 XL        0005 R 000000 XP
    0005   000220 XV        0005   000520 YA        0006   001020 YC        0000 R 000017 YDDK      0000 R 000025 YDDL
    0000 R 000015 YDK       0000 R 000023 YDL       0000 R 000013 YK        0000 R 000021 YL        0005 R 000060 YP
    0005   000300 YV

00101       1*            COMPILER(XM=1),(ADR=IND)                                                  000054
00103       2*            SUBROUTINE CPL2(I)                                                        000054
00105       3*            INCLUDE PARM                                                              000054
00110       4*            INCLUDE ELEM                                                              000054
00112       5*            INCLUDE FORC                                                              000054
00114       6*            INCLUDE STAT                                                              000054
00116       7*            INCLUDE CONN                                                              000054
00120       8*            INCLUDE SAVE                                                              000054
00122       9*            J=ID(I)                                                                   000054
00123      10*            K=IA(I+1)                                                                 000056
```

```
00124          L=IA(I,2)                                                                000060
00125          IF(SV(5,I).GT.SV(15,I).OR.SV(5,I).LT.-SV(15,I)) GO TO 5                  000062
00127          M=IP(I,1,1)                                                              000100
00130          N=IP(I,2,1)                                                              000102
00131          AX=0.0                                                                   000104
00132          AY=0.0                                                                   000105
00133          BX=SV(1,I)                                                               000106
00134          BY=SV(2,I)                                                               000110
00135          IJKLM = 4                                                                000112
00136          CALL RKIN(AX,AY,BX,BY,K,IJKLM,XK,YK,XDK,YDK,XDDK,YDDK)         K + K     000114
00137          BX=SV(3,I)                                                               000132
00140          BY=SV(4,I)                                                               000134
00141          CALL RKIN(AX,AY,BX,BY,L,IJKLM,XL,YL,XDL,YDL,XDDL,YDDL)         L + L     000136
00142          AN=(AP(K)+AP(L)+.017453293*(AC(K,M)+AC(L,N)-180.))/2.                    000154
00143          CAN=COS(AN)                                                              000201
00144          SAN=SIN(AN)                                                              000205
00145          DH=(XL-XK)*CAN+(YL-YK)*SAN                                               000211
00146          DV=(YL-YK)*CAN-(XL-XK)*SAN                                               000222
00147          VH=(XDL-XDK)*CAN+(YDL-YDK)*SAN+(AV(K)+AV(L))*.5*(-(XL-XK)*SAN+(YL-       000227
00147         1YK)*CAN)                                                                 000227
00150          IF(DH.GE.0.0) GO TO 1                                                    000250
00152          IF(VH.GT.0.0) GO TO 2                                                    000252
00154          FH=SV(6,I)*DH                                                            000254
00155          IF(DH.LT.SV(7,I)) FH=FH+(SV(8,I)-SV(6,I))*(DH-SV(7,I))                   000257
00157          IF(DH.LT.SV(9,I)) FH=FH+(SV(10,I)-SV(8,I))*(DH-SV(9,I))                  000272
00161          GO TO 3                                                                  000305
00162        1 CONTINUE                                                                 000307
00163          FH=0.0                                                                   000307
00164          IF(DH.LT. 0.0) GO TO 3                                                   000307
00166          IF(VH.LT.0.0) GO TO 6                                                    000312
00170          FH=SV(6,I)*DH                                                            000315
00171          IF(DH.GT.-SV(7,I)) FH=FH+(SV(8,I)-SV(6,I))*(DH+SV(7,I))                  000320
00173          IF(DH.GT.-SV(9,I)) FH=FH+(SV(10,I)-SV(8,I))*(DH+SV(9,I))                 000333
00175          GO TO 3                                                                  000346
00176        6 CONTINUE                                                                 000350
00177          FH=SV(14,I)*DH                                                           000350
00200          IF(DH.GT.-SV(13,I)) FH=FH+(SV(12,I)-SV(14,I))*(DH+SV(13,I))              000352
00202          IF(DH.GT.-SV(11,I)) FH=FH+(SV(10,I)-SV(12,I))*(DH+SV(11,I))              000365
00204          GO TO 3                                                                  000400
00205        2 CONTINUE                                                                 000402
00206          FH=SV(14,I)*DH                                                           000402
00207          IF(DH.LT.SV(13,I)) FH=FH+(SV(12,I)-SV(14,I))*(DH-SV(13,I))               000404
00211          IF(DH.LT.SV(11,I)) FH=FH+(SV(10,I)-SV(12,I))*(DH-SV(11,I))               000417
00213        3 CONTINUE                                                                 000433
00214          VSD=DV-SV(5,I)                                                           000433
00215          FV=0.0                                                                   000440
00216          IF(VSD.GT.PP(6,J,12)) FV=(VSD-PP(6,J,12))*(PP(5,J,12)*PP(13,J,12))       000442
00216         1/(PP(5,J,12)+PP(13,J,12))                                                000442
00220          IF(VSD.LT.-PP(14,J,12)) FV=(VSD+PP(14,J,12))*(PP(5,J,12)*PP(13,J,1       000460
00220         12))/(PP(5,J,12)+PP(13,J,12))                                             000460
00222          IF (ABS(FV).LT.PP(18,J,12)) GO TO 4                                      000476
00224          PP(18,J,12)=0.0                                                          000503
00225          FF=ABS(FH)*PP(17,J,12)                                                   000504
00226          IF(FF.GT.ABS(FV)) GO TO 4                                                000507
00230          FFV=FV                                                                   000513
00231          FV=SIGN(FF,FV)                                                           000515
00232          SV(5,I)=SV(5,I)+(FFV-FV)*(PP(5,J,12)+PP(13,J,12))/(PP(5,J,12)*PP(1       000521
```

```
00232      68*            13,J,12))                                                          000521
00233      69*          4 CONTINUE                                                           000533
00234      70*            A1=AP(K)+.017453293*AC(K,M)                                        000533
00235      71*            CA=COS(A1)                                                         000544
00236      72*            SA=SIN(A1)                                                         000550
00237      73*            FI(K,I,1)=-FH*CA+FV*SA                                             000554
00240      74*            FI(K,I,2)=-FH*SA-FV*CA                                             000565
00241      75*            FI(K,I,3)=FI(K,I,1)*(YP(K)-YK)+FI(K,I,2)*(XK-XP(K))                000575
00242      76*            A1=.017453293*(AC(L,N)-180.0)+AP(L)                                000611
00243      77*            CA=COS(A1)                                                         000624
00244      78*            SA=SIN(A1)                                                         000630
00245      79*            FI(L,I,1)=FH*CA-FV*SA                                              000634
00246      80*            FI(L,I,2)=FH*SA+FV*CA                                              000645
00247      81*            FI(L,I,3)=FI(L,I,1)*(YP(L)-YL)+FI(L,I,2)*(XL-XP(L))                000655
00250      82*        414 CONTINUE                                                           000672
00251      83*            IF(KICK.GT.-2) GO TO 313                                           000672
00251      84*     C      WRITE(6,797)I,K,L,(FI(K,I,JIJ),JIJ=1,3),(FI(L,I,JIJ),JIJ=1,3),SV(5,I)   000672
00251      85*     C     1,I)                                                                000672
00253      86*        797 FORMAT(3I5,6F15.0,F15.5)                                           000675
00253      87*     C      WRITE(6,415) DH,DV,VH,FH,FFV,FF,FV                                 000675
00254      88*        415 FORMAT(3F15.5,4F15.0)                                              000675
00255      89*            IF(KICK.EQ.-2) KICK=58                                             000675
00257      90*        313 CONTINUE                                                           000703
00260      91*            KICK=KICK-1                                                        000703
00261      92*            RETURN                                                             000705
00262      93*          5 CONTINUE                                                           000711
00263      94*            FI(K,I,1)=0.0                                                      000711
00264      95*            FI(K,I,2)=0.0                                                      000715
00265      96*            FI(K,I,3)=0.0                                                      000720
00266      97*            FI(L,I,1)=0.0                                                      000721
00267      98*            FI(L,I,2)=0.0                                                      000730
00270      99*            FI(L,I,3)=0.0                                                      000731
00271     100*            GO TO 414                                                          000732
00272     101*            END                                                                001006

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR*IS *DGSP
FOR SE38-06/02/77-14:38:54 (*0)

   SUBROUTINE DGSP       ENTRY POINT 000505

   STORAGE USED: CODE(1) 000533: DATA(0) 000110: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    ELEM    013117
    0004    STAT    000660
    0005    FORC    063140
    0006    CONN    002720
    0007    SAVE    030470

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010    RKIN
    0011    SQRT
    0012    NERR33

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000310 1L       0001    000245 11L      0001    000335 13L      0001    000302 172G     0001    000354 2L
    0001    000265 8L       0001    000357 9L       0004    000600 AA       0006    001760 AC       0004    000140 AP
    0004 R  000360 AV       0000 R  000005 AX       0000 R  000006 AY       0000 R  000007 BX       0000 R  000010 BY
    0000 R  000032 CA       0000 R  000033 D        0000 R  000030 DL       0000 R  000037 DRAG     0000 R  000026 DX
    0000 R  000027 DY       0005    062720 F        0000 R  000040 FD       0005    062500 FE       0005 R  000000 FI
    0003 I  000551 IA       0006    000000 IC       0003 I  000265 ID       0003    000000 IE       0000 I  000036 II
    0000 I  000011 IJKLM    0000    000044 INJPS    0003 I  001605 IP       0003    000001 IT       0000 I  000000 J
    0000 I  000001 K        0000 I  000002 L        0000 I  000003 M        0000 I  000004 N        0003 R  004731 PP
    0000 R  000035 Q        0000 R  000031 SA       0007 R  000000 SV       0000 R  000034 VD       0004    000440 XA
    0006    000060 XC       0000 R  000016 XDDK     0000 R  000024 XDDL     0000 R  000014 XDK      0000 R  000022 XDL
    0000 R  000012 XK       0000 R  000020 XL       0004 R  000000 XP       0004    000220 XV       0004    000520 YA
    0006    001020 YC       0000 R  000017 YDDK     0000 R  000025 YDDL     0000 R  000015 YDK      0000 R  000023 YDL
    0000 R  000013 YK       0000 R  000021 YL       0004 R  000060 YP       0004    000300 YV


00101       1*          COMPILER(XM=1),(ADR=IND)                                    000046
00103       2*          SUBROUTINE DGSP(I)                                          000046
00105       3*          INCLUDE PARM                                                000046
00110       4*          INCLUDE ELEM                                                000046
00112       5*          INCLUDE STAT                                                000046
00114       6*          INCLUDE FORC                                                000046
00116       7*          INCLUDE CONN                                                000046
00120       8*          INCLUDE SAVE                                                000046
00122       9*          J=ID(I)                                                     000046
00123      10*          K=IA(I,1)                                                   000050
00124      11*          L=IA(I,2)                                                   000052
00125      12*          M=IP(I,1,1)                                                 000054
00126      13*          N=IP(I,2,1)                                                 000056
```

```
00127     14*           AX=0.0                                                                                    000060
00130     15*           AY=0.0                                                                                    000061
00131     16*           BX=SV(1,I)                                                                                000062
00132     17*           BY=SV(2,I)                                                                                000064
00133     18*           IJKLM = 4                                                                                 000066
00134     19*           CALL RKIN(AX,AY,BX,BY,K,IJKLM,XK,YK,XDK,YDK,XDDK,YDDK)                  K + K            000070
00135     20*           BX=SV(3,I)                                                                                000106
00136     21*           BY=SV(4,I)                                                                                000110
00137     22*           CALL RKIN(AX,AY,BX,BY,L,IJKLM,XL,YL,XDL,YDL,XDDL,YDDL)                  L + L            000112
00140     23*           DX=XL-XK                                                                                  000130
00141     24*           DY=YL-YK                                                                                  000133
00142     25*           DL=SQRT(DX*DX+DY*DY)                                                                      000136
00143     26*           SA=DY/DL                                                                                  000150
00144     27*           CA=DX/DL                                                                                  000153
00145     28*           D=DL-SV(18,I)                                                                             000156
00146     29*           VD=(XDL-XDK)*CA+(YDL-YDK)*SA+(AV(K)+AV(L))*.5*(-(XL-XK)*SA+(YL-YK)                        000160
00146     30*          1*CA)                                                                                      000160
00147     31*           IF(D.GT.0.0) GO TO 8                                                                      000204
00151     32*           IF (D. GE . SV(19,I)) GO TO 1                                                             000206
00153     33*           IF(D.GE.SV(10,I)) GO TO 8                                                                 000212
00155     34*           IF(D.LT.SV(20,I)) GO TO 8                                                                 000216
00157     35*           IF(VD.GT.0.0) GO TO 11                                                                    000222
00161     36*           Q = SV (11,I) * (D - SV(10,I))                                                            000224
00162     37*           IF(D.LT.SV(12,I)) Q=Q+(SV(13,I)-SV(11,I))*(D-SV(12,I))                                    000230
00164     38*           GO TO 9                                                                                   000243
00165     39*          11 Q = SV(15,I) * (D - SV(10,I))                                                           000245
00166     40*           IF(D.LT.SV(14,I)) Q=Q+(SV(13,I)-SV(15,I))*(D-SV(14,I))                                    000250
00170     41*           GO TO 9                                                                                   000263
00171     42*            8 DO 12 II=1,3                                                                           000265
00174     43*           FI(K,I,II)=0.0                                                                            000302
00175     44*          12 FI(L,I,II)=0.0                                                                          000302
00177     45*           RETURN                                                                                    000304
00200     46*            1 IF(D.EQ.SV(6,I)) GO TO 2                                                               000310
00202     47*           IF(VD.GT.0.0) GO TO 13                                                                    000312
00204     48*           Q=SV(5,I)*D                                                                               000315
00205     49*           IF(D.LT.SV(6,I)) Q=Q+(SV(7,I)-SV(5,I))*(D-SV(6,I))                                        000320
00207     50*           GO TO 9                                                                                   000333
00210     51*          13 Q=SV(9,I)*D                                                                             000335
00211     52*           IF(D.LT.SV(8,I)) Q=Q+(SV(7,I)-SV(9,I))*(D-SV(8,I))                                        000337
00213     53*           GO TO 9                                                                                   000352
00214     54*            2 Q = SV (5,I) * D                                                                       000354
00215     55*            9 CONTINUE                                                                               000357
00216     56*           DRAG = PP(8,J,13)                                                                         000357
00217     57*           FD = DRAG*VD                                                                              000363
00220     58*           IF(ABS(VD) .GT. 1.) FD = SIGN(DRAG,VD)                                                    000365
00222     59*           Q = Q + FD                                                                                000375
00223     60*           FI(K,I,1)=-Q*CA                                                                           000400
00224     61*           FI(K,I,2)=-Q*SA                                                                           000406
00225     62*           FI(K,I,3)=-FI(K,I,1)*(YK-YP(K))+FI(K,I,2)*(XK-XP(K))                                      000413
00226     63*           FI(L,I,1)=-FI(K,I,1)                                                                      000427
00227     64*           FI(L,I,2)=-FI(K,I,2)                                                                      000435
00230     65*           FI(L,I,3)=-FI(L,I,1)*(YL-YP(L))+FI(L,I,2)*(XL-XP(L))                                      000441
00231     66*           RETURN                                                                                    000455
00232     67*           END                                                                                       000532

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR,I8 .END3
FOR 8E38-06/02/77-14139:32 (,0)

   SUBROUTINE END3       ENTRY POINT 000550

   STORAGE USED: CODE(1) 000575: DATA(0) 000113: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   ELEM   013117
    0004   STAT   000460
    0005   FORC   063140
    0006   CONN   002720
    0007   SAVE   030470

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010   RKIN
    0011   COS
    0012   SIN
    0013   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000471 313L      0001   000321 4L        0001   000460 414L      0001   000477 5L        0004   000600 AA
    0006 R 001760 AC        0000 R 000026 AN        0004 R 000140 AP        0004   000360 AV        0000 R 000005 AX
    0000 R 000006 AY        0000 R 000040 A1        0000 R 000007 BX        0000 R 000010 BY        0000 R 000041 CA
    0000 R 000027 CAN       0000 R 000031 DH        0000 R 000032 DV        0005   062720 F         0005   062500 FE
    0000 R 000036 FF        0000 R 000037 FFV       0000 R 000033 FH        0005 R 000000 FI        0000 R 000035 FV
    0003 I 000551 IA        0006   000000 IC        0003 I 000265 ID        0003   000000 IE        0000 I 000011 IJKLM
    0000   000047 INJP$     0003 I 001605 IP        0003   000001 IT        0000 I 000000 J         0000 I 000001 K
    0000 I 000043 KICK      0000 I 000002 L         0000 I 000003 M         0000 I 000004 N         0003 R 004731 PP
    0000 R 000042 SA        0000 R 000030 SAN       0007 R 000000 SV        0000 R 000034 V8D       0004   000440 XA
    0006   000060 XC        0000 R 000016 XDDK      0000 R 000024 XDDL      0000 R 000014 XDK       0000 R 000022 XDL
    0000 R 000012 XK        0000 R 000020 XL        0004 R 000000 XP        0004   000220 XV        0004   000520 YA
    0006   001020 YC        0000 R 000017 YDDK      0000 R 000025 YDDL      0000 R 000015 YDK       0000 R 000023 YDL
    0000 R 000013 YK        0000 R 000021 YL        0004 R 000060 YP        0004   000300 YV

00101      1*           COMPILER(XM=1),(ADR=IND)                                          000051
00103      2*           SUBROUTINE END3(I)                                                000051
00105      3*           INCLUDE PARM                                                      000051
00110      4*           INCLUDE ELEM                                                      000051
00112      5*           INCLUDE STAT                                                      000051
00114      6*           INCLUDE FORC                                                      000051
00116      7*           INCLUDE CONN                                                      000051
00120      8*           INCLUDE SAVE                                                      000051
00122      9*           J=ID(I)                                                           000051
00123     10*           K=IA(I,1)                                                         000053
00124     11*           L=IA(I,2)                                                         000055
00125     12*           IF(SV(5,I).GT.SV( 6,I).OR.SV(5,I).LT.-SV( 6,I)) GO TO 5           000057
```

```
00127     13*          M=IP(I,1,1)                                                                    000077
00130     14*          N=IP(I,2,1)                                                                    000101
00131     15*          AX=0.0                                                                         000103
00132     16*          AY=0.0                                                                         000104
00133     17*          BX=SV(1,I)                                                                     000105
00134     18*          BY=SV(2,I)                                                                     000107
00135     19*          IJKLM = 4                                                                      000111
00136     20*          CALL RKIN(AX,AY,BX,BY,K,IJKLM,XK,YK,XDK,YDK,XDDK,YDDK)          K + K          000113
00137     21*          BX=SV(3,I)                                                                     000132
00140     22*          BY=SV(4,I)                                                                     000134
00141     23*          CALL RKIN(AX,AY,BX,BY,L,IJKLM,XL,YL,XDL,YDL,XDDL,YDDL)          L + L          000136
00142     24*          AN=(AP(K)+AP(L)+.01745329*(AC(K,M)+AC(L,N)-180.))/2.                           000154
00143     25*          CAN=COS(AN)                                                                    000205
00144     26*          SAN=SIN(AN)                                                                    000213
00145     27*          DH=(XL-XK)*CAN+(YL-YK)*SAN                                                     000217
00146     28*          DV=(YL-YK)*CAN-(XL-XK)*SAN                                                     000224
00147     29*          IF(DH.GE.0.0) GO TO 5                                                          000232
00151     30*          FH=SV(8,I)*DH                                                                  000234
00152     31*          IF(DH .GT. SV(11,I)) FH = SV(10,I)*DH                                          000237
00154     32*          SV(11,I) = DH                                                                  000246
00155     33*          VSD=DV-SV(5,I)                                                                 000250
00156     34*          FV=0.0                                                                         000253
00157     35*          IF(VSD .GT. PP(5,J,14)) FV=(VSD-PP(5,J,14))*SV(7,I)                            000254
00161     36*          IF(VSD .LT. PP(5,J,14)) FV=(VSD+PP(5,J,14))*SV(7,I)                            000265
00163     37*          FF=ABS(FH)*PP(11,J,14)                                                         000276
00164     38*          IF(FF.GT.ABS(FV)) GO TO 4                                                      000302
00166     39*          FFV=FV                                                                         000306
00167     40*          FV=SIGN(FF,FV)                                                                 000310
00170     41*          SV(5,I)=SV(5,I)+(FFV-FV)/SV(7,I)                                               000314
00171     42*        4 CONTINUE                                                                       000321
00172     43*          A1=AP(K)+.01745329*AC(K,M)                                                     000321
00173     44*          CA=COS(A1)                                                                     000332
00174     45*          SA=SIN(A1)                                                                     000336
00175     46*          FI(K,I,1)=-FH*CA+FV*SA                                                         000342
00176     47*          FI(K,I,2)=-FH*SA-FV*CA                                                         000353
00177     48*          FI(K,I,3)=FI(K,I,1)*(YP(K)-YK)+FI(K,I,2)*(XK-XP(K))                            000363
00200     49*          A1=.01745329*(AC(L,N)-180.0)+AP(L)                                             000377
00201     50*          CA=COS(A1)                                                                     000412
00202     51*          SA=SIN(A1)                                                                     000416
00203     52*          FI(L,I,1)=FH*CA-FV*SA                                                          000422
00204     53*          FI(L,I,2)=FH*SA+FV*CA                                                          000433
00205     54*          FI(L,I,3)=FI(L,I,1)*(YP(L)-YL)+FI(L,I,2)*(XL-XP(L))                            000443
00206     55*      414 CONTINUE                                                                       000460
00207     56*          IF(KICK.GT.-2) GO TO 313                                                       000460
00207     57*    C     WRITE(6,797)I,K,L,(FI(K,I,JIJ),JIJ=1,3),(FI(L,I,JIJ),JIJ=1,3),SV(5,I)         000460
00207     58*    C    1,I)                                                                            000460
00207     59*    C797 FORMAT(3I5,6F15.0,F15,5)                                                       000460
00207     60*    C     WRITE(6,415) DH,DV,VH,FH,FFV,FF,FV                                             000460
00207     61*    C415 FORMAT(3F15.5,4F15.0)                                                          000460
00211     62*          IF(KICK.EQ.-2) KICK=58                                                         000463
00213     63*      313 CONTINUE                                                                       000471
00214     64*          KICK=KICK-1                                                                    000471
00215     65*          RETURN                                                                         000473
00216     66*        5 CONTINUE                                                                       000477
00217     67*          FI(K,I,1)=0.0                                                                  000477
00220     68*          FI(K,I,2)=0.0                                                                  000503
00221     69*          FI(K,I,3)=0.0                                                                  000506
```

```
00222      70*          FI(L,I,1)=0.0                                    000507
00223      71*          FI(L,I,2)=0.0                                    000516
00224      72*          FI(L,I,3)=0.0                                    000517
00225      73*          GO TO 414                                        000520
00226      74*          END                                              000574

      END OF COMPILATION:          NO  DIAGNOSTICS.
```

```
@FOR.IS .ACLMR1
FOR 8E38-06/02/77-14:40:55 (.0)

   SUBROUTINE ACLMR1     ENTRY POINT 001033

   STORAGE USED: CODE(1) 001051: DATA(0) 000233: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    ELEM    013117
    0004    STAT    000660
    0005    FORC    063140
    0006    CONN    002720
    0007    SAVE    030470

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010    RKIN
    0011    TRILN
    0012    COS
    0013    SIN
    0014    NWDUS
    0015    NIO2S
    0016    NERR3S

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000766 1L          0001    000527 2L          0000    000046 200F        0000    000065 201F        0001    000761 3L
    0001    000751 313L        0001    000300 4L          0001    000740 414L        0001    000562 5L          0001    000542 500L
    0004    000600 AA          0006 R  001760 AC          0000 R  000026 AN          0004 R  000140 AP          0004    000360 AV
    0000 R  000005 AX          0000 R  000006 AY          0000 R  000042 A1          0000 R  000007 BX          0000 R  000010 BY
    0000 R  000043 CA          0000 R  000027 CAN         0000 R  000033 DA          0000 R  000031 DM          0000 R  000035 DMU
    0000 R  000032 DV          0005    062720 F           0005    062500 FE          0000 R  000037 FM          0005 R  000000 FI
    0000 R  000041 FM          0000 R  000040 FV          0003 I  000551 IA          0006    000000 IC          0003 I  000265 ID
    0003    000000 IE          0000 I  000011 IJKLM       0000    000171 INJP$       0000 I  000036 IO          0003 I  001605 IP
    0003    000001 IT          0000 I  000000 J           0000 I  000001 K           0000 I  000045 KICK        0000 I  000002 L
    0000 I  000003 M           0000 I  000004 N           0003    004731 PP          0000 R  000044 SA          0000 R  000030 SAN
    0007 R  000000 SV          0000 R  000034 VSD         0004    000440 XA          0006    000060 XC          0000 R  000016 XDDK
    0000 R  000024 XDDL        0000 R  000014 XDK         0000 R  000022 XDL         0000 R  000012 XK          0000 R  000020 XL
    0004 R  000000 XP          0004    000220 XV          0004    000520 YA          0006    001020 YC          0000 R  000017 YDDK
    0000 R  000025 YDDL        0000 R  000015 YDK         0000 R  000023 YDL         0000 R  000013 YK          0000 R  000021 YL
    0004 R  000060 YP          0004    000300 YY

00101       1*          COMPILER(XH=1),(ADR=IND)                                                    000035
00103       2*          SUBROUTINE ACLMR1(I)                                                        000035
00105       3*          INCLUDE PARM                                                                000035
00110       4*          INCLUDE ELEM                                                                000035
00112       5*          INCLUDE STAT                                                                000035
00114       6*          INCLUDE FORC                                                                000035
00116       7*          INCLUDE CONN                                                                000035
```

```
00120      8*          INCLUDE SAVE                                                                    000035
00122      9*          J=ID(I)                                                                         000035
00123     10*          K=IA(I,1)                                                                       000041
00124     11*          L=IA(I,2)                                                                       000045
00125     12*          M=IP(I,1,1)                                                                     000051
00126     13*          N=IP(I,2,1)                                                                     000055
00127     14*          AX=0.0                                                                          000061
00130     15*          AY=0.0                                                                          000062
00131     16*          BX=SV(1,I)                                                                      000063
00132     17*          BY=SV(2,I)                                                                      000071
00133     18*          IJKLM = 4                                                                       000073
00134     19*          CALL RKIN(AX,AY,BX,BY,K,IJKLM,XK,YK,XDK,YDK,XDDK,YDDK)            K + K         000075
00135     20*          BX=SV(3,I)                                                                      000113
00136     21*          BY=SV(4,I)                                                                      000115
00137     22*          CALL RKIN(AX,AY,BX,BY,L,IJKLM,XL,YL,XDL,YDL,XDDL,YDDL)            L + L         000117
00140     23*          AN=(AP(K)+AP(L)+.017453293*(AC(K,M)+AC(L,N)-180.))/2.                           000135
00141     24*          CAN=COS(AN)                                                                     000162
00142     25*          SAN=SIN(AN)                                                                     000170
00143     26*          DH=(XL-XK)*CAN+(YL-YK)*SAN                                                      000174
00144     27*          DV=(YL-YK)*CAN-(XL-XK)*SAN                                                      000205
00145     28*          DA=0.017453293*(AC(L,N)-AC(K,M)-180.)+AP(L)-AP(K)                               000212
00146     29*          VSD=DV-SV(65,I)                                                                 000223
00147     30*          SV(16,I)=VSD                                                                    000225
00150     31*          SV(33,I)=DH                                                                     000226
00151     32*          SV(50,I)=DA                                                                     000227
00151     33*     C    WRITE (6,100) SV(16,I),SV(33,I),SV(50,I)                                        000227
00151     34*     C100 FORMAT (1X,' DV,DH,DA ',6E15.5)                                                 000227
00152     35*          IF((DH-SV(26,I)).GE. 0.0) GO TO 1                                               000230
00154     36*          IF(ABS(VSD) .GE. SV(64,I) .AND. DHO .GT. 0.0 .AND. DH .LE. 0.0)                 000233
00154     37*         1  GO TO 2                                                                      000233
00156     38*          IF( IO .EQ. 1) GO TO  500                                                       000256
00160     39*          IF(ABS(DH) .GE. SV(36,I) .OR. ABS(DA) .GE. SV(53,I)) GO TO 3                    000261
00162     40*        4 IF(ABS(VSD) .GE. SV(19,I)) GO TO 3                                              000300
00164     41*          VSD=SV(24,I)+SV(6,I)*(SV(16,I)-SV(17,I))                                        000306
00165     42*          IF(VSD .GE. 0.0) CALL TRILN(6,7,8,10,11,12,13,14,15,16,17,18,19,                000313
00165     43*         1   20,21,22,23,24,25,9,I)                                                      000313
00167     44*          IF(VSD .LT. 0.0) CALL TRILN(6,7,8,13,14,15,10,11,12,16,17,18,19,                000345
00167     45*         1    20,21,22,23,24,25,9,I)                                                     000345
00171     46*          IF(IO .EQ. 1) GO TO 5                                                           000377
00173     47*          CALL TRILN(30,31,32,66,67,68,27,28,29,33,34,35,36,37,38,39,40,41,               000402
00173     48*         1    42,26,I)                                                                   000402
00174     49*          DA=SV(58,I)+SV(60,I)*(SV(50,I)-SV(51,I))                                        000431
00175     50*          IF(DA .GE. 0.0) CALL TRILN(60,61,62,44,45,46,47,48,49,50,51,52,53,              000441
00175     51*         1    54,55,56,57,58,59,43,I)                                                    000441
00177     52*          IF(DA .LT. 0.0) CALL TRILN(60,61,62,47,48,49,44,45,46,50,51,52,53,              000473
00177     53*         1    54,55,56,57,58,59,43,I)                                                    000473
00201     54*          GO TO 5                                                                         000525
00202     55*        2 IO = 1                                                                          000527
00203     56*          WRITE(6,200) K,L,VSD,DH                                                         000530
00211     57*      200 FORMAT( ' OVER-RIDE BETWEEN MASSES ',I2,' AND ',I2,'     VSD=',               000542
00211     58*         1  F10.4,'   DH=',F10.4)                                                        000542
00212     59*      500 CONTINUE                                                                        000542
00213     60*          IF(IO .EQ. 1 .AND. ABS(VSD) .LT. SV(64,I)) GO TO 4                              000542
00215     61*        5 FH = SV(40,I)                                                                   000562
00216     62*          IF(FH .GT. 0.0) GO TO 1                                                         000566
00220     63*          FV=SV(23,I)                                                                     000570
00221     64*          FM=SV(57,I)                                                                     000572
```

```
00221     C      WRITE (6,101)                                                          000572
00221     C     1 SV(23,I),SV(40,I),SV(57,I)                                            000572
00221     C 101 FORMAT (1X, 'FV,FH,FM ', 6E15.5)                                        000572
00222            A1=AP(K)+.017453293*AC(K,M)                                            000574
00223            CA=COS(A1)                                                             000606
00224            SA=SIN(A1)                                                             000612
00225            FI(K,I,1)=-FH*CA+FV*SA                                                 000616
00226            FI(K,I,2)=-FH*SA-FV*CA                                                 000631
00227            FI(K,I,3)=FI(K,I,1)*(YP(K)-YK)+FI(K,I,2)*(XK-XP(K))-FM                 000641
00230            A1=.017453293*(AC(L,N)-180,0)+AP(L)                                    000656
00231            CA=COS(A1)                                                             000671
00232            SA=SIN(A1)                                                             000675
00233            FI(L,I,1)=FH*CA-FV*SA                                                  000701
00234            FI(L,I,2)=FH*SA+FV*CA                                                  000712
00235            FI(L,I,3)=FI(L,I,1)*(YP(L)-YL)+FI(L,I,2)*(XL-XP(L))+FM                 000722
00236         414 CONTINUE                                                              000740
00237            IF(KICK.GT.-2) GO TO 313                                               000740
00237     C      WRITE(6,797)I,K,L,(FI(K,I,JIJ),JIJ=1,3),(FI(L,I,JIJ),JIJ=1,3),SV(5,I)  000740
00237     C     1,I)                                                                    000740
00237     C 797 FORMAT(3I5,6F15.0,F15.5)                                                000740
00237     C      WRITE(6,415) DH,DV,VH,FH,FFV,FF,FV                                     000740
00237     C 415 FORMAT(3F15.5,4F15.0)                                                   000740
00241            IF(KICK.EQ.-2) KICK=58                                                 000743
00243         313 CONTINUE                                                              000751
00244            KICK=KICK-1                                                            000751
00245            DHO=DH                                                                 000753
00246            RETURN                                                                 000755
00247           3 WRITE(6,201)                                                          000761
00251         201 FORMAT( ' ANTICLIMBER FAILURE')                                       000766
00252           1 CONTINUE                                                              000766
00253            FI(K,I,1)=0.0                                                          000766
00254            FI(K,I,2)=0.0                                                          000774
00255            FI(K,I,3)=0.0                                                          000777
00256            FI(L,I,1)=0,0                                                          001000
00257            FI(L,I,2)=0.0                                                          001007
00260            FI(L,I,3)=0,0                                                          001010
00261            GO TO 414                                                              001011
00262            END                                                                    001050

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR*IS *TRILN
FOR SE3B-06/02/77-14:41:12 (*0)

   SUBROUTINE TRILN     ENTRY POINT 001350

   STORAGE USED: CODE(1) 001471: DATA(0) 000142: BLANK COMMON(2) 000000

    COMMON BLOCK81

    0003   SAVE   030470

   EXTERNAL REFERENCES (BLOCK, NAME)

    0004   NERR2>
    0005   NERR3>

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000172 1L        0001   000220 3L        0001   000630 31L       0001   000636 32L       0001   000644 33L
    0001   000662 34L       0001   000700 35L       0001   000644 36L       0001   000662 37L       0001   001115 4L
    0001   000701 40L       0001   001035 41L       0001   001223 5L        0001   001320 6L        0000 R 000006 D8
    0000 R 000001 DC        0000 R 000032 DD        0000 R 000033 DD0       0000 R 000034 DD4       0000 R 000035 DD5
    0000 R 000036 DD6       0000 R 000002 DL        0000 R 000010 DY        0000 R 000021 DYY       0000 R 000025 D0
    0000 R 000013 D1        0000 R 000014 D2        0000 R 000015 D3        0000 R 000016 D4        0000 R 000017 D5
    0000 R 000020 D6        0000 R 000027 F         0000 R 000007 FB        0000 R 000024 FC        0000 R 000004 FL
    0000 R 000005 FY        0000 R 000022 F1        0000 R 000023 F2        0000 R 000003 F3        0000 R 000031 G
    0000 I 000030 I         0000 I 000026 ID        0000   000040 INJPS     0000 R 000000 SK1       0000 R 000011 SK2
    0000 R 000012 SK3       0003 R 000000 SV

00101      1*            COMPILER(XM=1),(ADR=IND1                                                    000123
00103      2*            SUBROUTINE TRILN(IK1,IK2,IK3,ID1,ID2,ID3,ID4,ID5,ID6,IDC,IDL,IDY,           000123
00103      3*           1    IDYY,IF1,IF2,IF3,IFC,IFL,IFY,ID0,IJ)                                     000123
00105      4*            INCLUDE PARM                                                                000123
00110      5*            INCLUDE SAVE,LIST                                                           000123
00110      5*       SAVE  PROC                                                                       000123
00111      5*            COMMON/SAVE/SV(70,NFLE)                                                     000123
00111      5*             END                                                                        000123
00112      6*            SK1 = SV(IK1,IJ)                                                            000123
00113      7*            DC = SV(IDC,IJ)                                                             000126
00114      8*            DL = SV(IDL,IJ)                                                             000131
00115      9*            F3 = SV(IF3,IJ)                                                             000134
00116     10*            FL = SV(IFL,IJ)                                                             000137
00117     11*            FY = SV(IFY,IJ)                                                             000142
00117     12*       C    WRITE (6,1001     SK1,SK2,SK3,D1,D2,D3,D4,D5,D6,DC,DL,DY,DYY,F1,            000142
00117     13*       C   * F2,F3,FC,FL,FY,D0                                                          000142
00117     14*       C 100 FORMAT (' TRILN'/(12E10,4))                                               000142
00120     15*            IF(FY,GE,F3) GO TO 6                                                        000145
00122     16*            DB=DC-DL                                                                    000150
00123     17*            FB=FL+SK1*DB                                                                000152
00124     18*            IF(ABS(FB),GE,FY) GO TO 1                                                   000155
```

```
00126     19*          SV(IFC+IJ) = FB                                                    000161
00127     20*          SV(IFL+IJ) = FB                                                    000164
00130     21*          SV(IDL+IJ) = DC                                                    000165
00131     22*          RETURN                                                             000166
00132     23*        1 CONTINUE                                                           000172
00133     24*          IF (ABS(FB).NE.ABS(FY)) GO TO 3                                    000172
00135     25*          DY = SV(IDY+IJ)                                                    000175
00136     26*          SV(IFL+IJ) = SIGN(FY,FB)                                           000200
00137     27*          SV(IDL+IJ) = DY                                                    000205
00140     28*          SV(IFC+IJ) = FL                                                    000207
00141     29*          SV(IDC+IJ) = DY                                                    000212
00142     30*          RETURN                                                             000214
00143     31*        3 CONTINUE                                                           000220
00144     32*          SK2 = SV(IK2+IJ)                                                   000220
00145     33*          SK3 = SV(IK3+IJ)                                                   000222
00146     34*          D1 = SV(ID1+IJ)                                                    000225
00147     35*          D2 = SV(ID2+IJ)                                                    000230
00150     36*          D3 = SV(ID3+IJ)                                                    000233
00151     37*          D4 = SV(ID4+IJ)                                                    000236
00152     38*          D5 = SV(ID5+IJ)                                                    000241
00153     39*          D6 = SV(ID6+IJ)                                                    000244
00154     40*          DY = SV(IDY+IJ)                                                    000247
00155     41*          DYY = SV(IDYY+IJ)                                                  000252
00156     42*          F1 = SV(IF1+IJ)                                                    000255
00157     43*          F2 = SV(IF2+IJ)                                                    000260
00160     44*          FC = SV(IFC+IJ)                                                    000263
00161     45*          D0 = SV(ID0+IJ)                                                    000266
00162     46*          IF (D1.EQ.D2) ID=8                                                 000271
00164     47*          IF (D1.EQ.D2) GO TO 34                                             000276
00166     48*          IF((ABS(DL).GE.ABS(D1).AND.ABS(DL).LE.ABS(D2)).AND.(ABS(DC).GE.    000301
00166     49*         1 ABS(D1).AND.ABS(DC).LE.ABS(D2)))                    ID= 1        000301
00170     50*          IF (ABS(DL).GE.ABS(D2).AND.ABS(DC).GE.ABS(D2))          ID= 2      000344
00172     51*          IF((ABS(DL).GE.ABS(D0).AND.ABS(DL).LT.ABS(D1))                     000366
00172     52*         *                      .AND.(ABS(DC).GE.ABS(D1).AND.ABS(DC).LT.ABS(D  000366
00172     53*          12)))                ID= 3                                        000366
00174     54*          IF((ABS(DL).GE.ABS(D1).AND.ABS(DL).LT.ABS(D2)).AND.ABS(DC).GE.ABS(  000431
00174     55*          1D2))                  ID= 4                                     000431
00176     56*          IF((ABS(DL).GE.ABS(D0).AND.ABS(DL).LT.ABS(D1))                     000463
00176     57*         *                      .AND.ABS(DC).GE.ABS(D2))          ID= 5      000463
00200     58*          IF((ABS(DL).GE.ABS(D0).AND.ABS(DL).LE.ABS(D4))                     000515
00200     59*         * .AND.(ABS(DC).GE.ABS(D1).AND.ABS(DC).LE.ABS(D2)) ) ID=6          000515
00202     60*          IF((ABS(DL).GE.ABS(D0).AND.ABS(DL).LE.ABS(D4))                     000560
00202     61*         * .AND.(ABS(DC).GE.ABS(D2)))  ID=7                                  000560
00204     62*          GO TO (31,32,33,34,35,36,37) ,ID                                   000613
00205     63*       31 FC=FL+SK2*(DC-DL)                                                  000630
00206     64*          GO TO 40                                                           000634
00207     65*       32 FC=FL+SK3*(DC-DL)                                                  000636
00210     66*          GO TO 40                                                           000642
00211     67*       36 CONTINUE                                                          000644
00212     68*       33 DY=D1                                                             000644
00213     69*          FY=F1                                                              000645
00214     70*          DL=DY                                                              000647
00215     71*          FL = SIGN(FY,FB)                                                   000650
00216     72*          FC=FL+SK2*(DC-DL)                                                  000654
00217     73*          GO TO 40                                                           000660
00220     74*       37 CONTINUE                                                          000662
00221     75*       34 DY=D2                                                             000662
```

```
00222         FY=F2                                                000663
00223         DL=DY                                                000665
00224         FL = SIGN(FY,FB)                                     000666
00225         FC=FL+SK3*(DC-DL)                                    000672
00226         GO TO 40                                             000676
00227      35 GO TO 34                                             000700
00230      40 CONTINUE                                             000701
00231         F=ABS(FC)                                            000701
00232         IF(ABS(DC).GE.ABS(D1).AND.ABS(DC).LT.ABS(D2)) GO TO 4  000702
00234         I=11                                                 000722
00235         IF(ABS(DC) .GT. ABS(D2)) GO TO 41                    000724
00237         G=F2                                                 000733
00240         DD=ABS(DL-D1)                                        000735
00241         DD0=DD-G/SK1+ABS(D1-D0)                              000741
00242         DD4=DD+ABS(D1-D0)-2.0*G/SK1+ABS(D4-D0)               000747
00243         DD5=DD4              +ABS(D5-D4)                     000762
00244         DD6=DD5+ABS(D6-D5)-(F3-F2)/SK3                       000767
00245         D0=D0+SIGN(DD0,FL)                                   001000
00246         D4=D4+SIGN(DD4,FL)                                   001005
00247         D5=D5+SIGN(DD5,FL)                                   001012
00250          D6= D6+SIGN(DD6,FL)                                 001017
00251         DL=SIGN(D1,DL)                                       001024
00252         FL=SIGN(F1,FL)                                       001030
00253      41 CONTINUE                                             001035
00254         DD0=ABS(DC-D2)-(F-F2)/SK1                            001035
00255         DD5=DD0-F/SK1+ABS(D0-D5)                             001044
00256         DD6=ABS(D6-D5)+(F-F3)/SK3+DD5                        001054
00257         D0=D0+SIGN(DD0,FC)                                   001065
00260         D5=D5+SIGN(DD5,FC)                                   001072
00261          D6= D6+SIGN(DD6,FC)                                 001077
00262         D4=D5                                                001104
00263         D1=DC                                                001105
00264         D2=DC                                                001107
00265         F1=ABS(FC)                                           001110
00266          F2=F1                                               001112
00267         GO TO 5                                              001113
00270       4 CONTINUE                                             001115
00271         I=2?                                                 001115
00272         DD=ABS(DC-D1)                                        001116
00273         DD0=DD-F/SK1+ABS(D1-D0)                              001122
00274         DD4=DD+ABS(D1-D0)-2.0*F/SK1+ABS(D4-D0)               001133
00275         DD5=DD4-F2/SK2+F/SK2+ABS(D5-D4)                      001146
00276         DD6=DD5+ABS(D6-D5)-(F3-F2)/SK3                       001161
00277         D0=D0+SIGN(DD0,FC)                                   001172
00300         D4=D4+SIGN(DD4,FC)                                   001177
00301         D5=D5+SIGN(DD5,FC)                                   001204
00302          D6= D6+SIGN(DD6,FC)                                 001211
00303         F1=ABS(FC)                                           001216
00304         D1=DC                                                001220
00305       5 CONTINUE                                             001223
00306         FL=FC                                                001223
00307          DL=DC                                               001224
00310         FY=ABS(FC)                                           001226
00311         DY=DC                                                001230
00312         SV(ID1,IJ) = D1                                      001231
00313         SV(ID2,IJ) = D2                                      001234
00314         SV(ID3,IJ) = D3                                      001237
```

```
00315    133*            SV(ID4,IJ) = D4                                  001242
00316    134*            SV(ID5,IJ) = D5                                  001245
00317    135*            SV(ID6,IJ) = D6                                  001250
00320    136*            SV(IDC,IJ) = DC                                  001253
00321    137*            SV(IDL,IJ) = DL                                  001255
00322    138*            SV(IDY,IJ) = DY                                  001257
00323    139*            SV(IDYY,IJ) = DYY                                001261
00324    140*            SV(IF1,IJ) = F1                                  001264
00325    141*            SV(IF2,IJ) = F2                                  001267
00326    142*            SV(IF3,IJ) = F3                                  001272
00327    143*            SV(IFC,IJ ) = FC                                 001275
00330    144*            SV(IFL,IJ) = FL                                  001300
00331    145*            SV(IFY,IJ) = FY                                  001303
00332    146*            SV(IDD,IJ) = DD                                  001306
00333    147*            IF(FY.GE.F3) GO TO 6                             001311
00335    148*            RETURN                                           001314
00336    149*          6 SV(IFC,IJ) = 0.0                                 001320
00337    150*            RETURN                                           001321
00340    151*            END                                              001470

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS .NLTS
FOR SE38-06/02/77-14:41:54 (,0)

   SUBROUTINE NLTS       ENTRY POINT 000216

   STORAGE USED: CODE(1) 002341 DATA(0) 000037; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   ELEM   013117
    0004   STAT   000660
    0005   FORC   063140
    0006   CONN   002720

   EXTERNAL REFERENCES (BLOCK, NAME)

    0007   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000167 100L     0001    000101 20L      0001    000106 30L      0001    000121 40L      0001    000135 60L
    0001    000155 80L      0004    000600 AA       0006    001760 AC       0004 R  000140 AP       0004    000360 AV
    0005    062720 F        0005    062500 FE       0005 R  000000 FI       0000 R  000007 FL       0003 I  000551 IA
    0006    000000 IC       0003 I  000265 ID       0003    000000 IF       0000    000012 INJP$    0003    001605 IP
    0003    000001 IT       0000 I  000004 J        0000 I  000005 K        0000 R  000000 K1       0000 R  000001 K2
    0000 I  000006 L        0000 R  000002 L1       0000 R  000003 L2       0003 R  004731 PP       0000 R  000010 THETA
    0004    000440 XA       0006    000060 XC       0004    000000 XP       0004    000220 XV       0004    000520 YA
    0006    001020 YC       0004    000060 YP       0004    000300 YV


00101      1*         COMPILER(XM=1),(ADR=IND)                              000024
00103      2*         SUBROUTINE NLTS(I)                                    000024
00105      3*         INCLUDE PARM                                          000024
00110      4*         INCLUDE ELEM                                          000024
00112      5*         INCLUDE STAT                                          000024
00114      6*         INCLUDE FORC                                          000024
00116      7*         INCLUDE CONN                                          000024
00120      8*         REAL K1,K2,L1,L2                                      000024
00121      9*         J=ID(I)                                               000024
00122     10*         K=IA(I,1)                                             000026
00123     11*         L=IA(I,2)                                             000030
00124     12*         FI(K,I,1) = 0.0                                       000032
00125     13*         FI(K,I,2) = 0.0                                       000036
00126     14*         FI(L,I,1) = 0.0                                       000041
00127     15*         FI(L,I,2) = 0.0                                       000045
00130     16*         FL=PP(9,J,16)*.017453293                              000050
00131     17*         THETA = AP(L)-AP(K)-FL                                000055
00132     18*         IF(THETA) 10,20,30                                    000065
00135     19*      10 K1=-PP(5,J,16)                                        000067
00136     20*         K2=-PP(7,J,16)                                        000071
00137     21*         L1= PP(6,J,16)                                        000073
```

```
00140     22*            L2= PP(8,J,16)                                    000075
00141     23*            GO TO 40                                          000077
00142     24*         20 FI(K,I,3) = 0.0                                   000101
00143     25*            GO TO 100                                         000104
00144     26*         30 K1=-PP(1,J,16)                                    000106
00145     27*            K2=-PP(3,J,16)                                    000112
00146     28*            L1=-PP(2,J,16)                                    000114
00147     29*            L2=-PP(4,J,16)                                    000116
00150     30*         40 IF(ABS(THETA) .GT. ABS(L1)) GO TO 60              000121
00152     31*            FI(K,I,3)=K1*THETA                                000125
00153     32*            GO TO 100                                         000133
00154     33*         60 IF(ABS(THETA) .GT. ABS(L2)) GO TO 80              000135
00156     34*            FI(K,I,3)=-K1*L1+K2*(THETA+L1)                    000141
00157     35*            GO TO 100                                         000153
00160     36*         80 FI(K,I,3)=-K1*L1-K2*(L2-L1)                       000155
00161     37*        100 FI(L,I,3)=-FI(K,I,3)                             000167
00162     38*            RETURN                                            000176
00163     39*            END                                               000233

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
#FOR.IB ,*RIN
FOR BE38-06/02/77-14142100 (.0)

   SUBROUTINE *RIN       ENTRY POINT 000334

   STORAGE USED: CODE(1) 000357: DATA(0) 000100: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    ELEM    013117
    0004    FORC    063140
    0005    STAT    000660
    0006    CONN    002720
    0007    SAVE    030470

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010    RKIN
    0011    NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000277 1L        0001    000174 2L        0001    000177 3L        0005    000600 AA        0006    001760 AC
    0005    000140 AP        0005    000360 AV        0000 R  000012 AX        0000 R  000013 AY        0000 R  000002 BK
    0000 R  000014 BX        0000 R  000015 BY        0000 R  000030 DD        0000 R  000005 DK        0004    062720 F
    0000 R  000034 FA        0000 R  000033 FAA       0000 R  000031 FA1       0000 R  000032 FA2       0004    062500 FE
    0000 R  000035 FF        0004 R  000000 FI        0000 R  000006 FK        0000 R  000037 FM        0000 R  000036 FP
    0003 I  000551 IA        0006    000000 IC        0003 I  000265 ID        0003    000000 IE        0000 I  000016 IJKLM
    0000    000041 INJP$     0003 I  001605 IP        0003    000001 IT        0000 I  000000 J         0000 I  000010 K
    0000 I  000011 L         0003 R  004731 PP        0000 R  000004 RR        0000 R  000007 RY        0000 R  000001 SK
    0000 R  000003 ST        0007 R  000000 SV        0000 R  000017 X         0005    000440 XA        0006 R  000060 XC
    0000 R  000021 XD        0000 R  000023 XDD       0005 R  000000 XP        0000 R  000027 XPD       0005    000220 XV
    0000 R  000020 Y         0005    000520 YA        0006 R  001020 YC        0000 R  000022 YD        0000 R  000024 YDD
    0005 R  000060 YP        0000 R  000026 YPD       0000 R  000025 YQ        0005    000300 YV

00101       1*          COMPILER(XM=1),(ADR=IND)                                                    000043
00103       2*          SUBROUTINE *RIN(I)                                                          000043
00105       3*          INCLUDE PARM                                                                000043
00110       4*          INCLUDE ELEM                                                                000043
00112       5*          INCLUDE FORC                                                                000043
00114       6*          INCLUDE STAT                                                                000043
00116       7*          INCLUDE CONN                                                                000043
00120       8*          INCLUDE SAVE                                                                000043
00122       9*          J=ID(I)                                                                     000043
00123      10*          SK=PP(1,J,18)                                                               000047
00124      11*          BK=PP(2,J,18)                                                               000051
00125      12*          ST=PP(3,J,18)                                                               000053
00126      13*          RR=PP(4,J,18)                                                               000055
00127      14*          DK=PP(5,J,18)                                                               000057
00130      15*          FK=PP(6,J,18)                                                               000061
```

```
00131     16*            RY=PP(7,J,18)                                                    000063
00132     17*            K=IA(I,1)                                                        000065
00133     18*            L=IP(I,1,1)                                                      000067
00134     19*            AX=0.0                                                           000071
00135     20*            AY=0.0                                                           000072
00136     21*            BX=XC(K,L)                                                       000073
00137     22*            BY=YC(K,L)                                                       000101
00140     23*            IJKLM = 4                                                        000105
00141     24*            CALL RKIN(AX,AY,BX,BY,K,IJKLM,X,Y,XD,YD,XDD,YDD)                 000107
00142     25*            YQ=-X*SV(1,I)+(Y-RY)*SV(2,I)                                     000125
00143     26*            IF(YQ.GT.RR) GO TO 1                                             000134
00145     27*            YPD=-XD*SV(1,I)+YD*SV(2,I)                                       000137
00146     28*            XPD= XD*SV(2,I)+YD*SV(1,I)                                       000145
00147     29*            DD=RR-YQ                                                         000153
00150     30*            IF(DD.LT.ST) GO TO 2                                             000156
00152     31*            FA1=SK*ST                                                        000161
00153     32*            FA2=BK*(DD-ST)                                                   000164
00154     33*            FAA=FA1+FA2                                                      000170
00155     34*            GO TO 3                                                          000172
00156     35*          2 FAA=SK*DD                                                        000174
00157     36*          3 FA=FAA-DK*YPD                                                    000177
00160     37*            FF=SIGN(1.0,XPD)                                                 000202
00161     38*            IF(XPD.LT.1.0.AND.XPD.GT.-1.0) FF=XPD                            000206
00163     39*            FP=FK*FA*FF                                                      000226
00164     40*            FM=RR*FP                                                         000232
00165     41*            FI(K,I,1)=FA*SV(1,I)+FP*SV(2,I)                                  000234
00166     42*            FI(K,I,2)=-FA*SV(2,I)+FP*SV(1,I)                                 000246
00167     43*            FI(K,I,3)=FM-FI(K,I,1)*(Y-YP(K))+FI(K,I,2)*(X-XP(K))             000256
00170     44*            RETURN                                                           000273
00171     45*          1 FI(K,I,1)=0.0                                                    000277
00172     46*            FI(K,I,2)=0.0                                                    000305
00173     47*            FI(K,I,3)=0.0                                                    000306
00174     48*            RETURN                                                           000307
00175     49*            END                                                              000356

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR,I8 .NLSP
FOR SE38=06/02/77-14:42:14 (.0)

   SUBROUTINE NLSP       ENTRY POINT 000370

   STORAGE USED: CODE(1) 000414: DATA(0) 000107: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   ELEM   013117
    0004   FORC   063140
    0005   STAT   000660
    0006   CONN   002720

   EXTERNAL REFERENCES (BLOCK, NAME)

    0007   RRIN
    0010   SQRT
    0011   NEWR3S

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000235 1L        0001    000253 2L        0005    000600 AA        0006    001760 AC        0005    000140 AP
    0005    000360 AV        0000 R  000015 AX        0000 R  000016 AY        0000 R  000017 BX        0000 R  000020 BY
    0000 R  000042 CA        0000 R  000014 D         0000 R  000040 DL        0000 R  000036 DX        0000 R  000037 DY
    0004    062720 F         0000 R  000044 FA        0000 R  000045 FD        0004    062500 FE        0004 R  000000 FI
    0003 I  000551 IA        0006    000000 IC        0003 I  000265 ID        0003    000000 IE        0000 I  000021 IJKLM
    0000    000046 INJPS     0003 I  001605 IP        0003    000001 IT        0000 I  000000 J         0000 I  000001 K
    0000 I  000002 L         0000 I  000003 M         0000 I  000004 N         0003 R  004731 PP        0000 R  000041 SA
    0000 R  000011 SC        0000 R  000043 SD        0000 R  000013 SL        0000 R  000012 ST        0000 R  000005 S1
    0000 R  000006 S2        0000 R  000007 S3        0000 R  000010 S4        0005    000440 XA        0006 R  000060 XC
    0000 R  000026 XDDM      0000 R  000034 XDDN      0000 R  000024 XDM       0000 R  000032 XDN       0000 R  000022 XM
    0000 R  000030 XN        0005 R  000000 XP        0005    000220 XV        0005    000520 YA        0006 R  001020 YC
    0000 R  000027 YDDM      0000 R  000035 YDDN      0000 R  000025 YDM       0000 R  000033 YDN       0000 R  000023 YM
    0000 R  000031 YN        0005 R  000060 YP        0005    000300 YV

00101       1*          COMPILER(XM=1),(ADR=IND)                                                    000043
00103       2*          SUBROUTINE NLSP(I)                                                          000043
00105       3*          INCLUDE PARM                                                                000043
00110       4*          INCLUDE ELEM                                                                000043
00112       5*          INCLUDE FORC                                                                000043
00114       6*          INCLUDE STAT                                                                000043
00116       7*          INCLUDE CONN                                                                000043
00120       8*          J=ID(I)                                                                     000043
00121       9*          K=IA(I,1)                                                                   000047
00122      10*          L=IA(I,2)                                                                   000051
00123      11*          M=IP(I,1,1)                                                                 000053
00124      12*          N=IP(I,2,1)                                                                 000055
00125      13*          S1=PP(1,J,19)                                                               000057
00126      14*          S2=PP(2,J,19)                                                               000061
```

```
00127     15*          S3=PP(3,J,19)                                                          000063
00130     16*          S4=PP(4,J,19)                                                          000065
00131     17*          SC=PP(5,J,19)                                                          000067
00132     18*          ST=PP(6,J,19)                                                          000071
00133     19*          SL=PP(7,J,19)                                                          000073
00134     20*          D = PP(8,J,19)                                                         000075
00135     21*          AX=0.0                                                                 000077
00136     22*          AY=0.0                                                                 000100
00137     23*          BX=XC(K,M)                                                             000101
00140     24*          BY=YC(K,M)                                                             000110
00141     25*          IJKLM = 4                                                              000114
00142     26*          CALL RKIN(AX,AY,BX,PY,K,IJKLM,XM,YM,XDM,YDM,XDDM,YDDM)        K + M    000116
00143     27*          BX=XC(L,N)                                                             000134
00144     28*          BY=YC(L,N)                                                             000143
00145     29*          CALL RKIN(AX,AY,BX,PY,L,IJKLM,XN,YN,XDN,YDN,XDDN,YDDN)        L + N    000147
00146     30*          DX=XN-XM                                                               000165
00147     31*          DY=YN-YM                                                               000170
00150     32*          DL=SQRT(DX*DX+DY*DY)                                                   000173
00151     33*          SA=DY/DL                                                               000203
00152     34*          CA=DX/DL                                                               000206
00153     35*          SD=DL-SL                                                               000211
00154     36*          IF(SD.LT.0.0) GO TO 1                                                  000213
00156     37*          FA=SD*S3                                                               000215
00157     38*          IF(SD.GT.ST) FA=FA+(S4-S3)*(SD-ST)                                     000220
00161     39*          GO TO 2                                                                000233
00162     40*        1 CONTINUE                                                               000235
00163     41*          FA=SD*S1                                                               000235
00164     42*          IF(SD.LT.-SC) FA=FA+(S2-S1)*(SD+SC)                                    000237
00166     43*        2 CONTINUE                                                               000253
00167     44*          FD = D*((XDN-XDM)*CA + (YDN-YDM)*SA)                                   000253
00170     45*          FA = FA + FD                                                           000263
00171     46*          FI(K,I,1)=-FA*CA                                                       000265
00172     47*          FI(K,I,2)=-FA*SA                                                       000273
00173     48*          FI(K,I,3)=-FI(K,I,1)*(YM-YP(K))+FI(K,I,2)*(XM-XP(K))                   000300
00174     49*          FI(L,I,1)=-FI(K,I,1)                                                   000314
00175     50*          FI(L,I,2)=-FI(K,I,2)                                                   000322
00176     51*          FI(L,I,3)=-FI(L,I,1)*(YN-YP(L))+FI(L,I,2)*(XN-XP(L))                   000326
00177     52*          RETURN                                                                 000342
00200     53*          END                                                                    000413

     END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
#FOR.IS .NLDS
FOR SE3B-06/02/77-14142153 (.0)


   SUBROUTINE NLDS       ENTRY POINT 000420


   STORAGE USED: CODE(1) 000445: DATA(0) 000114: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   ELEM   013117
    0004   STAT   000660
    0005   CONN   002720
    0006   FORC   063140


   EXTERNAL REFERENCES (BLOCK, NAME)

    0007   RKIN
    0010   SQRT
    0011   NERR3$


   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0004    000600 AA        0005    001760 AC        0004    000140 AP        0004    000360 AV        0000 R 000015 AX
    0000 R  000014 AY        0000 R  000015 BX        0000 R  000016 BY        0000 R  000040 CA        0000 R  000036 DL
    0000 R  000034 DX        0000 R  000035 DY        0000 R  000005 D1        0000 R  000006 D2        0000 R  000007 D3
    0000 R  000010 D4        0006    062720 F         0000 R  000042 FA        0006    062500 FE        0006 R  000000 FI
    0003 I  000551 IA        0005    000000 IC        0003 I  000265 ID        0003    000000 IE        0000 I  000017 IJKLM
    0000    000043 INJPS     0003 I  001605 IP        0003    000001 IT        0000 I  000000 J         0000 I  000001 K
    0000 I  000002 L         0000 I  000003 M         0000 1  000004 N         0003 R  004731 PP        0000 R  000037 SA
    0000 R  000041 VA        0000 R  000011 VN        0000 R  000012 VP        0004    000440 XA        0005 R  000060 XC
    0000 R  000024 XDDM      0000 R  000032 XDDN      0000 R  000022 XDM       0000 R  000030 XDN       0000 R  000020 XM
    0000 R  000026 XN        0004 R  000000 XP        0004    000220 XV        0004    000520 YA        0005 R  001020 YC
    0000 R  000025 YDDM      0000 R  000033 YDDN      0000 R  000023 YDM       0000 R  000031 YDN       0000 R  000021 YM
    0000 R  000027 YN        0004 R  000060 YP        0004    000300 YV


00101       1*          COMPILER(XM=1),(ADR=IND)                                                    000043
00103       2*          SUBROUTINE NLDS(1)                                                          000043
00105       3*          INCLUDE PARM                                                                000043
00110       4*          INCLUDE ELEM                                                                000043
00112       5*          INCLUDE STAT                                                                000043
00114       6*          INCLUDE CONN                                                                000043
00116       7*          INCLUDE FORC                                                                000043
00120       8*          J=ID(I)                                                                     000043
00121       9*          K=IA(I,1)                                                                   000047
00122      10*          L=IA(I,2)                                                                   000051
00123      11*          M=IP(1,1,1)                                                                 000053
00124      12*          N=IP(1,2,1)                                                                 000055
00125      13*          D1=PP(1,J,20)                                                               000057
00126      14*          D2=PP(2,J,20)                                                               000061
00127      15*          D3=PP(3,J,20)                                                               000063
```

```
00130          D4=PP(4,J,20)                                                                   000065
00131          VN=PP(5,J,20)                                                                   000067
00132          VP=PP(6,J,20)                                                                   000071
00133          AX=0.0                                                                          000073
00134          AY=0.0                                                                          000074
00135          BX=XC(K,M)                                                                      000075
00136          BY=YC(K,M)                                                                      000104
00137          IJKLM = 4                                                                       000110
00140          CALL RKIN(AX,AY,BX,BY,K,IJKLM,XM,YM,XDM,YDM,XDDM,YDDM)             K + M       000112
00141          BX=XC(L,N)                                                                      000130
00142          BY=YC(L,N)                                                                      000137
00143          CALL RKIN(AX,AY,BX,BY,L,IJKLM,XN,YN,XDN,YDN,XDDN,YDDN)             L + N       000143
00144          DX=XN-XM                                                                        000161
00145          DY=YN-YM                                                                        000164
00146          DL=SQRT(DX*DX+DY*DY)                                                            000167
00147          SA=DY/DL                                                                        000177
00150          CA=DX/DL                                                                        000202
00151          VA=(XDN-XDM)*CA+(YDN-YDM)*SA                                                    000205
00152          IF(VA.LT.-VN) FA=D1*VA                                                          000215
00154          IF(VA.LT.0.0.AND.VA.GE.-VN) FA=D2*VA                                            000224
00156          IF(VA.GT.0.0.AND.VA.LE.VP) FA=D3*VA                                             000300
00160          IF(VA.GT.VP) FA=D4*VA                                                           000320
00162          FI(K,I,1)=-FA*CA                                                                000327
00163          FI(K,I,2)=-FA*SA                                                                000333
00164          FI(K,I,3)=-FI(K,I,1)*(YM-YP(K))+FI(K,I,2)*(XM-XP(K))                            000337
00165          FI(L,I,1)=-FI(K,I,1)                                                            000351
00166          FI(L,I,2)=-FI(K,I,2)                                                            000354
00167          FI(L,I,3)=-FI(L,I,1)*(YN-YP(L))+FI(L,I,2)*(XN-XP(L))                            000357
00170          RETURN                                                                          000371
00171          END                                                                             000444

     END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR,IS .SLSPR
FOR 8E3B-06/02/77-14:44:43 (.0)

   SUBROUTINE SLSPR      ENTRY POINT 000364

   STORAGE USED: CODE(1) 000410: DATA(0) 000104: BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    STAT    000660
    0004    ELEM    013117
    0005    FORC    063140
    0006    CONN    002720

   EXTERNAL REFERENCES (BLOCK, NAME)

    0007    RKIN
    0010    SQRT
    0011    NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000230 1L       0001    000315 2L       0003    000600 AA       0006    001760 AC       0003    000140 AP
    0003    000360 AV       0000 R  000013 AX       0000 R  000014 AY       0000 R  000015 BX       0000 R  000016 BY
    0000 R  000040 CA       0000 R  000036 DL       0000 R  000034 DX       0000 R  000035 DY       0005    062720 F
    0000 R  000042 FA       0005    062500 FE       0005 R  000000 FI       0004 I  000551 IA       0006    000000 IC
    0004 I  000265 ID       0004    000000 IE       0000 I  000017 IJKLM     0000    000043 INJPS    0004 I  001605 IP
    0004    000001 IT       0000 I  000000 J        0000 I  000007 K        0000 I  000010 L        0000 I  000011 M
    0000 I  000012 N        0004 R  004731 PP       0000 R  000037 SA       0000 R  000003 SD       0000 R  000006 SF
    0000 R  000005 SFL      0000 R  000001 SKA      0000 R  000002 SKB      0000 R  000041 SL       0000 R  000004 SP
    0003    000440 XA       0006 R  000060 XC       0000 R  000024 XDOM     0000 R  000032 XDON     0000 R  000022 XDM
    0000 R  000030 XDN      0000 R  000020 XM       0000 R  000026 XN       0003 R  000000 XP       0003    000220 XV
    0003    000520 YA       0006 R  001020 YC       0000 R  000025 YDOM     0000 R  000033 YDON     0000 R  000023 YDM
    0000 R  000031 YDN      0000 R  000021 YM       0000 R  000027 YN       0003 R  000060 YP       0003    000300 YV


00101      1*           COMPILER(XM=1),(ADR=INO)                                                    000043
00103      2*           SUBROUTINE SLSPR(I)                                                         000043
00105      3*           INCLUDE PARM                                                                000043
00110      4*           INCLUDE STAT                                                                000043
00112      5*           INCLUDE ELEM                                                                000043
00114      6*           INCLUDE FORC                                                                000043
00116      7*           INCLUDE CONN                                                                000043
00120      8*           J=ID(1)                                                                     000043
00120      9*     C*****************************************************************************   000043
00120     10*     C          THIS SPRING IS ACTIVE ONLY IN COMPRESSION                              000043
00120     11*     C*****************************************************************************   000043
00121     12*          SKA=PP(1,J,21)                                                              000047
00122     13*          SKB=PP(2,J,21)                                                              000051
00123     14*          SD=PP(3,J,21)                                                               000053
00124     15*          SP=PP(4,J,21)                                                               000055
```

```
00125    16*          SFL=PP(5,J,21)                                                      000057
00126    17*          SF=PP(6,J,21)                                                       000061
00127    18*          K=IA(I,1)                                                           000063
00130    19*          L=IA(I,2)                                                           000065
00131    20*          M=IP(I,1,1)                                                         000067
00132    21*          N=IP(I,2,1)                                                         000071
00133    22*          AX=0.0                                                              000073
00134    23*          AY=0.0                                                              000074
00135    24*          BX=XC(K,M)                                                          000075
00136    25*          BY=YC(K,M)                                                          000103
00137    26*          IJKLM = 1                                                           000107
00140    27*          CALL RKIN(AX,AY,BX,BY,K,IJKLM,XM,YM,XDM,YDM,XDDM,YDDM)      K + M   000111
00141    28*          BX=XC(L,N)                                                          000127
00142    29*          BY=YC(L,N)                                                          000136
00143    30*          CALL RKIN(AX,AY,BX,BY,L,IJKLM,XN,YN,XDN,YDN,XDDN,YDDN)     L + N    000142
00144    31*          DX=XN-XM                                                            000160
00145    32*          DY=YN-YM                                                            000163
00146    33*          DL=SQRT(DX*DX+DY*DY)                                                000166
00147    34*          SA=DY/DL                                                            000176
00150    35*          CA=DX/DL                                                            000201
00151    36*          SL=DL-SF                                                            000204
00152    37*          IF(SL .GE. 0.0) GO TO 2                                             000206
00154    38*          FA=SL*SKA-SP                                                        000210
00155    39*          IF(-SL .LE. SD) GO TO 1                                             000214
00157    40*          FA =FA+(SL+SD)*(SKB-SKA)                                            000220
00160    41*        1 CONTINUE                                                            000230
00161    42*          IF(-FA .GE. SFL) GO TO 2                                            000230
00163    43*          FI(K,I,1)=-FA*CA                                                    000233
00164    44*          FI(K,I,2)=-FA*SA                                                    000242
00165    45*          FI(K,I,3)=-FI(K,I,1)*(YM-YP(K))+FI(K,I,2)*(XM-XP(K))                000247
00166    46*          FI(L,I,1)=-FI(K,I,1)                                                000263
00167    47*          FI(L,I,2)=-FI(K,I,2)                                                000271
00170    48*          FI(L,I,3)=-FI(L,I,1)*(YN-YP(L))+FI(L,I,2)*(XN-XP(L))                000275
00171    49*          RETURN                                                              000311
00172    50*        2 CONTINUE                                                            000315
00173    51*          FI(K,I,1)=0.0                                                       000315
00174    52*          FI(K,I,2)=0.0                                                       000321
00175    53*          FI(K,I,3)=0.0                                                       000324
00176    54*          FI(L,I,1)=0.0                                                       000325
00177    55*          FI(L,I,2)=0.0                                                       000334
00200    56*          FI(L,I,3)=0.0                                                       000335
00201    57*          RETURN                                                              000336
00202    58*          END                                                                 000407

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
#FOR,IS .TAPB
FOR 6E3B-06/02/77-14145:09 (.0)

   SUBROUTINE TAPB       ENTRY POINT 000740

   STORAGE USED: CODE(1) 000766; DATA(0) 000233; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003    ELEM    013117
    0004    STAT    000660
    0005    FORC    063140
    0006    CONN    002720
    0007    SAVE    030470

   EXTERNAL REFERENCES (BLOCK, NAME)

    0010    RKIN
    0011    STRS
    0012    SQRT
    0013    ASIN
    0014    NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000403 165G      0001    000505 206G      0004    000600 AA        0006    001760 AC        0004 R 000140 AP
    0004 R  000360 AV        0000 R  000055 AX        0000 R  000056 AY        0000 R  000123 BI        0000 R 000124 BJ
    0000 R  000057 BX        0000 R  000060 BY        0000 R  000126 CA        0000 R  000110 CI        0000 R 000111 CJ
    0000 R  000127 DAMP      0000 R  000102 DD        0000 R  000115 DMI       0000 R  000116 DMJ       0000 R 000103 DL
    0000 R  000100 DX        0000 R  000101 DY        0000 R  000120 EI        0000 R  000121 EJ        0000 R 000107 EP
    0005    062720 F         0000 R  000130 FD        0000 R  000132 FDA       0000 R  000131 FDY       0005   062500 FE
    0005 R  000000 FI        0000 R  000113 HHI       0000 R  000114 HHJ       0003 I  000551 IA        0006   000000 IC
    0003 I  000265 ID        0003    000000 IE        0000 I  000061 IJKLM     0000    000145 INJP$     0003 I 001605 IP
    0003    000001 IT        0000 I  000050 J         0000 I  000112 JI        0000 I  000117 JJ        0000 I 000122 JK
    0000 I  000051 K         0000 I  000052 L         0000 I  000053 M         0000 I  000054 N         0000 R 000105 PI
    0000 R  000106 PJ        0003 R  004731 PP        0000 R  000125 SA        0000 R  000024 SI        0000 R 000036 SJ
    0007 R  000000 SV        0000 R  000104 TA        0000 R  000133 VI        0000 R  000134 VJ        0004   000440 XA
    0006 R  000060 XC        0000 R  000076 XD        0000 R  000066 XDDM      0000 R  000074 XDDN      0000 R 000064 XDM
    0000 R  000072 XDN       0000 R  000062 XM        0000 R  000070 XN        0004 R  000000 XP        0004   000220 XV
    0004    000520 YA        0006 R  001020 YC        0000 R  000077 YD        0000 R  000067 YDDM      0000 R 000075 YDDN
    0000 R  000065 YDM       0000 R  000073 YDN       0000 R  000000 YI        0000 R  000012 YJ        0000 R 000063 YM
    0000 R  000071 YN        0004 R  000060 YP        0004    000300 YV

00101       1*          COMPILER(XM=1),(ADR=IND)                                                    000061
00103       2*          SUBROUTINE TAPB(I)                                                          000061
00105       3*          INCLUDE PARM                                                                000061
00110       4*          INCLUDE ELEM                                                                000061
00112       5*          INCLUDE STAT                                                                000061
00114       6*          INCLUDE FORC                                                                000061
00116       7*          INCLUDE CONN                                                                000061
```

```
      INCLUDE SAVE                                                          000061
      DIMENSION YI(10),YJ(10),SI(10),SJ(10)                                 000061
      J=ID(I)                                                               000061
      K=IA(I,1)                                                             000065
      L=IA(I,2)                                                             000067
      M=IP(I,1,1)                                                           000071
      N=IP(I,2,1)                                                           000073
      AX=0.0                                                                000075
      AY=0.0                                                                000076
      BX=XC(K,M)                                                            000077
      BY=YC(K,M)                                                            000105
      IJKLM = 4                                                             000111
      CALL RKIN(AX,AY,BX,BY,K,IJKLM,XM,YM,XDM,YDM,XDDM,YDDM)       K + M    000113
      BX=XC(L,N)                                                            000132
      BY=YC(L,N)                                                            000141
      CALL RKIN(AX,AY,BX,BY,L,IJKLM,XN,YN,XDN,YDN,XDDN,YDDN)       L + N    000145
      XD=XN-XM                                                              000163
      YD=YN-YM                                                              000166
      DX=XD-SV(1,I)                                                         000171
      DY=YD-SV(2,I)                                                         000173
      DD=2.*(SV(1,I)*DX+SV(2,I)*DY)+DX*DX+DY*DY                             000175
      DL=SQRT(SV(5,I)**2+DD)                                                000210
      TA=(SV(1,I)*DY-SV(2,I)*DX)/(SV(5,I)*DL)                               000220
      IF(ABS(TA).GT..01) TA=ASIN(TA)                                        000230
      PI=AP(K)-SV(3,I)-TA                                                   000240
      PJ=AP(L)-SV(4,I)-TA                                                   000246
      EP=DD/((SV(5,I)+DL)*SV(5,I))                                          000254
      CI=-2.*(2.*PI+PJ)/SV(5,I)                                             000262
      CJ=2.*(PI+2.*PJ)/SV(5,I)                                              000267
      JI=PP(9,J,22)+0.1                                                     000275
      HHI=PP(5,J,22)*0.5                                                    000307
      HHJ=PP(7,J,22)*0.5                                                    000312
      DHI=2.*HHI/(PP(9,J,22)-1.0)                                           000315
      DHJ=2.*HHJ/(PP(9,J,22)-1.0)                                           000322
      DO 1 JJ=1,JI                                                          000403
      YI(JJ)=HHI-FLOAT(JJ-1)*DHI                                            000407
      YJ(JJ)=HHJ-FLOAT(JJ-1)*DHJ                                            000414
      EI=EP-YI(JJ)*CI                                                       000420
      EJ=EP-YJ(JJ)*CJ                                                       000423
      JK=6*JJ                                                               000426
      CALL STRS(EI,PP(1,J,22),SV(JK,I),SI(JJ))                              000431
      JK=JK+3                                                               000450
      CALL STRS(EJ,PP(1,J,22),SV(JK,I),SJ(JJ))                              000453
    1 CONTINUE                                                              000475
      PI=0.0                                                                000475
      PJ=0.0                                                                000476
      BI=0.0                                                                000477
      BJ=0.0                                                                000500
      DO 2 JJ=2,JI                                                          000505
      PI=PI+SI(JJ-1)+SI(JJ)                                                 000505
      PJ=PJ+SJ(JJ-1)+SJ(JJ)                                                 000510
      BI=BI+SI(JJ-1)*(2.*YI(JJ-1)+YI(JJ))+SI(JJ)*(YI(JJ-1)+2.*YI(JJ))       000514
      BJ=BJ+SJ(JJ-1)*(2.*YJ(JJ-1)+YJ(JJ))+SJ(JJ)*(YJ(JJ-1)+2.*YJ(JJ))       000527
    2 CONTINUE                                                              000543
      SA = YD/DL                                                            000543
      CA = XD/DL                                                            000546
      DAMP = PP(10,J,22)                                                    000551
```

```
00221      65*            FD = DAMP*((XDN-XDM)*CA + (YDN-YDM)*SA)                                                    000553
00222      66*            FDY = PP(11,J,22)*((YDN-YDM)*CA-(XDN-XDM)*SA)                                              000560
00223      67*            FDA = PP(12,J,22)*(AV(L)-AV(K))                                                            000567
00224      68*            PJ=(PI*DHI*PP(6,J,22)+PJ*DHJ*PP(8,J,22))/4.0                                               000575
00225      69*            PJ = PJ + FD                                                                               000604
00226      70*            PI=-PJ                                                                                     000606
00227      71*            BI=BI*DHI*PP(6,J,22)/6.0                                                                   000607
00230      72*            BJ=-BJ*DHJ*PP(8,J,22)/6.0                                                                  000613
00231      73*            VI=(BI+BJ)/DL                                                                              000617
00232      74*            BI = BI-FDA                                                                                000622
00233      75*            BJ = BJ + FDA                                                                              000624
00234      76*            VI = VI - FDY                                                                              000626
00235      77*            VJ=-VI                                                                                     000630
00236      78*            FI(K,I,1)=PI*CA-VI*SA                                                                      000631
00237      79*            FI(K,I,2)=PI*SA+VI*CA                                                                      000636
00240      80*            FI(K,I,3)=BI-FI(K,I,1)*(YM-YP(K))+FI(K,I,2)*(XM-XP(K))                                     000645
00241      81*            FI(L,I,1)=PJ*CA-VJ*SA                                                                      000660
00242      82*            FI(L,I,2)=PJ*SA+VJ*CA                                                                      000666
00243      83*            FI(L,I,3)=BJ-FI(L,I,1)*(YN-YP(L))+FI(L,I,2)*(XN-XP(L))                                     000675
00244      84*            RETURN                                                                                     000710
00245      85*            END                                                                                        000765

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR*I8 *RKIN
FOR BE38=06/02/77-15:19:12 (*0)

   SUBROUTINE RKIN      ENTRY POINT 000632

   STORAGE USED: CODE(1) 0007118 DATA(0) 0000648 BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   STAT   000660

   EXTERNAL REFERENCES (BLOCK, NAME)

    0004   NERR2$
    0005   SIN
    0006   COS
    0007   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000101 1L        0001   000131 2L        0001   000165 3L        0001   000244 4L        0001   000316 5L
    0001   000406 6L        0001   000502 7L        0003 R 000600 AA        0000 R 000002 AC        0000 R 000005 ADA
    0000 R 000010 ADDA      0003 R 000140 AP        0003 R 000360 AV        0000 R 000011 C1        0000 R 000012 C2
    0000   000014 INJP$      0000 I 000013 N         0003 R 000440 XA        0000 R 000000 XC        0000 R 000003 XDA
    0000 R 000006 XDDA      0003 R 000000 XP        0003 R 000220 XV        0003 R 000520 YA        0000 R 000001 YC
    0000 R 000004 YDA       0000 R 000007 YDDA      0003 R 000060 YP        0003 R 000300 YV


00101      1*          COMPILER(XM=1),(ADR=IND)                                                    000032
00103      2*          SUBROUTINE RKIN(AX,AY,BX,BY,L,I,XR,YB,XDB,YDB,XDDB,YDDB)                    000032
00103      3*     C                                                                 00000100       000032
00103      4*     C    DISPLACEMENT - VELOCITY - ACCELERATION OF PT,B COMPUTED       00000200       000032
00103      5*     C    IN RELATION TO A FIXED REF, AXIES                             00000300       000032
00103      6*     C                                                                 00000400       000032
00103      7*     C                                                                 00000500       000032
00103      8*     C                                                                 00000600       000032
00103      9*     C                                                                 00000700       000032
00103     10*     C    (AX,AY) AND (BX,BY) ARE COORD, OF PT,A AND PT,B IN BODY AXIES SYST000008     000032
00103     11*     C                                                                 00000900       000032
00103     12*     C    (XA,YA) - COORD, OF PT,A IN FIXED REF, SYST,                  00001000       000032
00103     13*     C                                                                 00001100       000032
00103     14*     C    (AA) - ROTATION ANGLE OF BODY AXIES                           00001200       000032
00103     15*     C                                                                 00001300       000032
00103     16*     C    (XDA,YDA) - VEL, OF PT,A IN REF, CODRD,                       00001400       000032
00103     17*     C                                                                 00001500       000032
00103     18*     C    (ADA) - ANG, VEL, OF BODY AXIES                               00001600       000032
00103     19*     C                                                                 00001700       000032
00103     20*     C    (XDDA,YDDA) - ACC, OF  PT,A IN REF,CODRD,                     00001800       000032
00103     21*     C                                                                 00001900       000032
00103     22*     C    (ADDA) - ANG, ACC, OF BODY AXIES                              00002000       000032
00103     23*     C                                                                 00002100       000032
```

```
00103     24*     C     (I) - COMP. SELECTION INDEX                              00002200   000032
00103     25*     C                                                              00002300   000032
00105     26*           INCLUDE PARM                                                        000032
00110     27*           INCLUDE STAT,LIST                                                   000032
00110     27*     STAT  PROC                                                                000032
00111     27*           COMMON/STAT/XP(NUMM),YP(NUMM),AP(NUMM),                             000032
00111     27*          1            XV(NUMM),YV(NUMM),AV(NUMM),                             000032
00111     27*          2            XA(NUMM),YA(NUMM),AA(NUMM)                              000032
00111     27*      END                                                                      000032
00112     28*           XC = XP(L)                                                          000032
00113     29*           YC = YP(L)                                                          000034
00114     30*           AC = AP(L)                                                          000036
00115     31*           XDA = XV(L)                                                         000040
00116     32*           YDA = YV(L)                                                         000042
00117     33*           ADA = AV(L)                                                         000044
00120     34*           XDDA = XA(L)                                                        000046
00121     35*           YDDA = YA(L)                                                        000050
00122     36*           ADDA = AA(L)                                                        000052
00123     37*           C1=AX-BX                                                 00002600   000054
00124     38*           C2=AY-BY                                                 00002700   000057
00125     39*           N=I                                                      00002800   000062
00126     40*           GO TO (1,2,3,4,5,6,7),N                                  00002900   000064
00127     41*         1 CONTINUE                                                 00003000   000101
00127     42*     C                                                              00003100   000101
00127     43*     C     DISP. OF PT.B ----                                       00003200   000101
00127     44*     C                                                              00003300   000101
00130     45*           XB=XC+C2*SIN(AC)-C1*COS(AC)                                         000101
00131     46*           YB=YC-C1*SIN(AC)-C2*COS(AC)                                         000116
00132     47*           RETURN                                                   00003600   000125
00133     48*         2 CONTINUE                                                 00003700   000131
00133     49*     C                                                              00003800   000131
00133     50*     C     VEL. OF PT.B ----                                        00003900   000131
00133     51*     C                                                              00004000   000131
00134     52*           XDB=XDA+C2*COS(AC)*ADA+C1*SIN(AC)*ADA                               000131
00135     53*           YDB=YDA-C1*COS(AC)*ADA+C2*SIN(AC)*ADA                               000150
00136     54*           RETURN                                                   00004300   000161
00137     55*         3 CONTINUE                                                 00004400   000165
00137     56*     C                                                              00004500   000165
00137     57*     C     ACC. OF PT.B ----                                        00004600   000165
00137     58*     C                                                              00004700   000165
00140     59*           XDDB=XDDA+C2*COS(AC)*ADDA-C2*SIN(AC)*ADA**2                         000165
00140     60*          C+C1*SIN(AC)*ADDA+C1*COS(AC)*ADA**2                                  000165
00141     61*           YDDB=YDDA-C1*COS(AC)*ADDA+C1*SIN(AC)*ADA**2                         000223
00141     62*          C+C2*SIN(AC)*ADDA+C2*COS(AC)*ADA**2                                  000223
00142     63*           RETURN                                                   00005200   000240
00143     64*         4 CONTINUE                                                 00005300   000244
00143     65*     C                                                              00005400   000244
00143     66*     C     DISP. AND VEL. OF PT.B ----                              00005500   000244
00143     67*     C                                                              00005600   000244
00144     68*           XB=XC+C2*SIN(AC)-C1*COS(AC)                                         000244
00145     69*           YB=YC-C1*SIN(AC)-C2*COS(AC)                                         000263
00146     70*           XDB=XDA+C2*COS(AC)*ADA+C1*SIN(AC)*ADA                               000274
00147     71*           YDB=YDA-C1*COS(AC)*ADA+C2*SIN(AC)*ADA                               000303
00150     72*           RETURN                                                   00006100   000312
00151     73*         5 CONTINUE                                                 00006200   000316
00151     74*     C                                                              00006300   000316
00151     75*     C     DISP. AND ACC. OF PT.B ----                              00006400   000316
```

```
00151     76*      C                                                                            00006500     000316
00152     77*              XB=XC+C2*SIN(AC)-C1*COS(AC)                                                       000316
00153     78*              YB=YC-C1*SIN(AC)-C2*COS(AC)                                                       000335
00154     79*              XDDB=XDDA+C2*COS(AC)*ADDA-C2*SIN(AC)*ADA**2                                       000346
00154     80*             C+C1*SIN(AC)*ADDA+C1*COS(AC)*ADA**2                                                000346
00155     81*              YDDB=YDDA-C1*COS(AC)*ADDA+C1*SIN(AC)*ADA**2                                       000365
00155     82*             C+C2*SIN(AC)*ADDA+C2*COS(AC)*ADA**2                                                000365
00156     83*              RETURN                                                           00007200     000402
00157     84*            6 CONTINUE                                                         00007300     000406
00157     85*      C                                                                        00007400     000406
00157     86*      C       VEL. AND ACC. OF PT.B ----                                       00007500     000406
00157     87*      C                                                                        00007600     000406
00160     88*              XDB=XDA+C2*COS(AC)*ADA+C1*SIN(AC)*ADA                                             000406
00161     89*              YDB=YDA-C1*COS(AC)*ADA+C2*SIN(AC)*ADA                                             000427
00162     90*              XDDB=XDDA+C2*COS(AC)*ADDA-C2*SIN(AC)*ADA**2                                       000442
00162     91*             C+C1*SIN(AC)*ADDA+C1*COS(AC)*ADA**2                                                000442
00163     92*              YDDB=YDDA-C1*COS(AC)*ADDA+C1*SIN(AC)*ADA**2                                       000461
00163     93*             C+C2*SIN(AC)*ADDA+C2*COS(AC)*ADA**2                                                000461
00164     94*              RETURN                                                           00008300     000476
00165     95*            7 CONTINUE                                                         00008400     000502
00165     96*      C                                                                        00008500     000502
00165     97*      C       DISP. AND VEL. AND ACC. OF PT.B ----                             00008600     000502
00165     98*      C                                                                        00008700     000502
00166     99*              XB=XC+C2*SIN(AC)-C1*COS(AC)                                                       000502
00167    100*              YB=YC-C1*SIN(AC)-C2*COS(AC)                                                       000521
00170    101*              XDB=XDA+C2*COS(AC)*ADA+C1*SIN(AC)*ADA                                             000532
00171    102*              YDB=YDA-C1*COS(AC)*ADA+C2*SIN(AC)*ADA                                             000541
00172    103*              XDDB=XDDA+C2*COS(AC)*ADDA-C2*SIN(AC)*ADA**2                                       000550
00172    104*             C+C1*SIN(AC)*ADDA+C1*COS(AC)*ADA**2                                                000550
00173    105*              YDDB=YDDA-C1*COS(AC)*ADDA+C1*SIN(AC)*ADA**2                                       000567
00173    106*             C+C2*SIN(AC)*ADDA+C2*COS(AC)*ADA**2                                                000567
00174    107*              RETURN                                                           00009600     000604
00175    108*              END                                                              00009700     000710

       END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS ,LEQS
FOR SE38-06/02/77-15:19:26 (,0)

   SUBROUTINE LEQS      ENTRY POINT 000605

   STORAGE USED: CODE(1) 000650; DATA(0) 000102; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000150 1L        0001   000316 10L       0001   000054 115G      0001   000104 124G      0001   000122 135G
    0001   000127 142G      0001   000166 160G      0001   000225 166G      0001   000515 17L       0001   000113 2L
    0001   000555 20L       0001   000267 203G      0001   000310 213G      0001   000377 222G      0001   000403 226G
    0001   000411 233G      0001   000452 244G      0001   000460 247G      0001   000473 260G      0001   000540 273G
    0001   000237 5L        0001   000440 50L       0000 R 000003 BIG       0000 I 000002 I         0000   000024 INJP$
    0000 I 000004 J         0000 I 000011 K         0000 I 000017 KK        0000 I 000013 L         0000 I 000007 NBGRW
    0000 I 000001 NBSIZ     0000 I 000014 NCOLB     0000 I 000006 NN        0000 I 000015 NROW      0000 I 000000 NSIZ
    0000 I 000005 NUMSYS    0000 I 000016 NXS       0000 R 000012 PMULT     0000 R 000010 TEMP


00101      1*          COMPILER(XM=1),(ADR=IND)                                          000023
00103      2*          SUBROUTINE LEQS(A,B,NEQS,NSOLNS,IA,IB,DET,ISCALE)                 000023
00105      3*             DIMENSION A(IA,IA),B(IA,IB)                                    000023
00106      4*          NSIZ = NEQS                                          TEST 340     000023
00107      5*          NBSIZ = NSOLNS                                       TEST 350     000025
00110      6*          NSIZ = NEQS                                          TEST7340     000027
00111      7*          NBSIZ = NSOLNS                                       TEST7350     000030
00112      8*          DET=1.0                                              TEST7360     000031
00113      9*           ISCALE=0                                                         000033
00114     10*          DO 1 I=1,NSIZ                                        TEST7370     000054
00117     11*          BIG=A(I,1)                                           TEST7380     000066
00120     12*          IF(NSIZ-1)50,50,51                                   TEST7390     000070
00123     13*       51 DO 2 J=2,NSIZ                                        TEST7400     000074
00126     14*          IF(ABS(BIG)-ABS(A(I,J))) 3,2,2                       TEST7410     000104
00131     15*        3 BIG=A(I,J)                                           TEST7420     000110
00132     16*        2 CONTINUE                                             TEST7430     000122
00134     17*          DO 4 J=1,NSIZ                                        TEST7440     000122
00137     18*        4 A(I,J)=A(I,J)/BIG                                    TEST7450     000122
00141     19*          DO 41 J=1,NBSIZ                                      TEST7460     000127
00144     20*       41 B(I,J)=B(I,J)/BIG                                    TEST7470     000127
00146     21*          DET=DET*BIG                                          TEST7480     000132
00147     22*           IF(ABS(DET)-1.E+20) 1,1,60                                       000135
00152     23*      60    DET=DET*1.E-10                                                  000141
00153     24*             ISCALE=ISCALE+1                                                000144
00154     25*        1 CONTINUE                                             TEST7490     000155
00156     26*          NUMSYS=NSIZ-1                                        TEST7500     000155
00157     27*          DO 14 I=1,NUMSYS                                     TEST7510     000160
00162     28*          NN=I+1                                               TEST7520     000205
```

```
00163            BIG=A(I,1)                                            TEST7530    000210
00164            NBGRW=I                                               TEST7540    000212
00165            DO 5 J=NN,NSIZ                                        TEST7550    000225
00170            IF(ABS(BIG)-ABS(A(J,I))) 6,5,5                        TEST7560    000225
00173          6 BIG=A(J,I)                                            TEST7570    000232
00174            NBGRW=J                                               TEST7580    000234
00175          5 CONTINUE                                              TEST7590    000241
00177            IF(NBGRW-I) 7,10,7                                    TEST7600    000241
00202          7 DO 8 J=I,NSIZ                                         TEST7610    000244
00205            TEMP=A(NBGRW,J)                                       TEST7620    000267
00206            A(NBGRW,J)=A(I,J)                                     TEST7630    000270
00207          8 A(I,J)=TEMP                                           TEST7640    000272
00211            DET=-DET                                              TEST7650    000274
00212            DO 9 J=1,NBSIZ                                        TEST7660    000310
00215            TEMP=B(NBGRW,J)                                       TEST7670    000310
00216            B(NBGRW,J)=B(I,J)                                     TEST7680    000311
00217          9 B(I,J)=TEMP                                           TEST7690    000313
00221         10 DO 13 K=NN,NSIZ                                       TEST7700    000316
00224            PMULT=-A(K,I)/A(I,I)                                  TEST7710    000377
00225            DO 11 J=NN,NSIZ                                       TEST7720    000403
00230         11 A(K,J)=PMULT*A(I,J)+A(K,J)                            TEST7730    000403
00232            DO 12 L=1,NBSIZ                                       TEST7740    000411
00235         12 B(K,L)=PMULT*B(I,L)+B(K,L)                            TEST7750    000411
00237         13 CONTINUE                                              TEST7760    000440
00241         14 CONTINUE                                              TEST7770    000440
00243         50 DO 15 NCOLB=1,NBSIZ                                   TEST7780    000440
00246            DO 19 I=1,NSIZ                                        TEST7790    000460
00251            NROW=NSIZ+1-I                                         TEST7800    000460
00252            TEMP=0.0                                              TEST7810    000463
00253            NXS=NSIZ-NROW                                         TEST7820    000464
00254            IF(NXS) 16,17,16                                      TEST7830    000466
00257         16 DO 18 K=1,NXS                                         TEST7840    000467
00262            KK=NSIZ+1-K                                           TEST7850    000473
00263         18 TEMP=TEMP+B(KK,NCOLB)*A(NROW,KK)                      TEST7860    000476
00265         17 B(NROW,NCOLB)=(B(NROW,NCOLB)-TEMP)/A(NROW,NROW)       TEST7870    000515
00266         19 CONTINUE                                              TEST7880    000540
00270         15 CONTINUE                                              TEST7890    000540
00272            DO 20 I=1,NSIZ                                        TEST7900    000540
00275            DET=DET*A(I,I)                                                    000540
00276               IF(ABS(DET)-1.E-10)61,61,20                                    000542
00301         61  DET=DET*1.E+10                                                   000546
00302               ISCALE=ISCALE-1                                                000551
00303         20  CONTINUE                                                         000556
00305            RETURN                                                TEST7920    000556
00306            END                                                   TEST7930    000647

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
#FOR,IS .MALG
FOR SE38-06/06/77-09:08:41 (.0)

   SUBROUTINE MALG       ENTRY POINT 000637

   STORAGE USED: CODE(1) 000730: DATA(0) 000105: BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NWDU$
    0004   NIO2$
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000   000005 1000F      0001   000477 101L      0001   000366 103L      0001   000253 105L      0001   000161 124G
    0001   000162 127G       0001   000215 136G      0001   000220 141G      0001   000222 145G      0001   000272 163G
    0001   000273 166G       0001   000326 175G      0001   000333 200G      0001   000615 200L      0001   000204 202L
    0001   000335 204G       0001   000246 204L      0001   000315 206L      0001   000361 208L      0001   000430 210L
    0001   000472 212L       0001   000537 214L      0001   000603 216L      0001   000405 222G      0001   000406 225G
    0001   000437 234G       0001   000442 237G      0001   000444 243G      0001   000514 261G      0001   000515 264G
    0001   000546 273G       0001   000553 276G      0001   000555 302G      0001   000610 999L      0000 I 000001 I
    0000   000014 INJP$      0000 I 000000 IS        0000 I 000002 J         0000 I 000004 K         0000 R 000003 I
```



```
00101      1*            COMPILER(XM=1),(ADR=IND)                                          000121
00103      2*            SUBROUTINE MALG(IOP,A,AM,IA,NRA,NCA,NRDA,NCDA,B,BM,IB,NRB,NCB,NR  000121
00103      3*           1OP,NCDB,C,NRC,NCC,NRDC,NCDC)                                      000121
00103      4*      C                                                                       000121
00103      5*      C     THIS ROUTINE PERFORMS MATRIX ALGEBRA OF THE FORM                  000121
00103      6*      C          AM*(A OR A') + BM*(B OR B') = (C)  FOR IOP = 0               000121
00103      7*      C          AM*(A OR A') + BM*(B OR B') = (C)  FOR IOP = 1               000121
00103      8*      C                                                                       000121
00103      9*      C     THE CODES ARE     AM = CONSTANT MULTIPLIER FOR A MATRIX         000121
00103     10*      C                       BM = CONSTANT MULTIPLIER FOR B MATRIX         000121
00103     11*      C                       IX = 0.  ORDINARY MATRIX                      000121
00103     12*      C                       IX = 1.  TRANSPOSED MATRIX                    000121
00103     13*      C                       NRX = NUMBER OF ROWS IN X                     000121
00103     14*      C                       NCX = NUMBER OF COLUMNS IN X                  000121
00103     15*      C                       NRDX = NUMBER OF ROWS DIMENSIONED FOR X       000121
00103     16*      C                       NCDX = NUMBER OF COLUMNS DIMENSIONED FOR X    000121
00103     17*      C                                                                       000121
00105     18*            DIMENSION A(NRDA,NCDA),B(NRDB,NCDB),C(NRDC,NCDC)                  000121
00106     19*            IS=14*IB                                                          000121
00107     20*            IF(IS) 999,100,101                                                000124
00112     21*       100  IF(IA) 999,102,103                                                000127
00115     22*       102  IF(IB) 999,104,105                                                000133
00115     23*      C                                                                       000133
00115     24*      C     PERFORM   AM*(A) IOP BM*(B)                                       000133
00115     25*      C                                                                       000133
00120     26*       104  IF(IOP) 999,201,202                                               000137
```

```
00123     27*        201 DO 203 I=1,NRA                                    000162
00126     28*            DO 203 J=1,NCB                                    000162
00131     29*        203 C(I,J)=AM*A(I,J)+BM*B(I,J)                        000162
00134     30*            GO TO 204                                         000202
00135     31*        202 DO 106 I=1,NRA                                    000220
00140     32*            DO 106 J=1,NCB                                    000220
00143     33*            T=0.                                              000220
00144     34*            DO 107 K=1,NCA                                    000222
00147     35*        107 T=T+AM*A(I,K)*BM*B(K,J)                           000222
00151     36*        106 C(I,J)=T                                          000230
00154     37*        204 NRC=NRA                                           000246
00155     38*            NCC=NCB                                           000247
00156     39*            GO TO 200                                         000251
00156     40*      C                                                       000251
00156     41*      C     PERFORM   AM*(A) IOP BM*(B')                      000251
00156     42*      C                                                       000251
00157     43*        105 IF(IOP) 999,205,206                               000253
00162     44*        205 DO 207 I=1,NRA                                    000273
00165     45*            DO 207 J=1,NRB                                    000273
00170     46*        207 C(I,J)=AM*A(I,J)+BM*B(J,I)                        000273
00173     47*            GO TO 208                                         000313
00174     48*        206 DO 108 I=1,NRA                                    000333
00177     49*            DO 108 J=1,NRB                                    000333
00202     50*            T=0.                                              000333
00203     51*            DO 109 K=1,NCA                                    000335
00206     52*        109 T=T+AM*A(I,K)*BM*B(J,K)                           000335
00210     53*        108 C(I,J)=T                                          000343
00213     54*        208 NRC=NRA                                           000361
00214     55*            NCC=NRB                                           000362
00215     56*            GO TO 200                                         000364
00215     57*      C                                                       000364
00215     58*      C     PERFORM   AM*(A') IOP BM*(B)                      000364
00215     59*      C                                                       000364
00216     60*        103 IF(IOP) 999,209,210                               000366
00221     61*        209 DO 211 I=1,NCA                                    000406
00224     62*            DO 211 J=1,NCB                                    000406
00227     63*        211 C(I,J)=AM*A(J,I)+BM*B(I,J)                        000406
00232     64*            GO TO 212                                         000426
00233     65*        210 DO 110 I=1,NCA                                    000442
00236     66*            DO 110 J=1,NCB                                    000442
00241     67*            T=0.                                              000442
00242     68*            DO 111 K=1,NRA                                    000444
00245     69*        111 T=T+AM*A(K,I)*BM*B(K,J)                           000444
00247     70*        110 C(I,J)=T                                          000452
00252     71*        212 NRC=NCA                                           000472
00253     72*            NCC=NCB                                           000473
00254     73*            GO TO 200                                         000475
00254     74*      C                                                       000475
00254     75*      C     PERFORM   AM*(A') IOP BM*(B')                     000475
00254     76*      C                                                       000475
00255     77*        101 IF(IOP) 999,213,214                               000477
00260     78*        213 DO 215 I=1,NCA                                    000515
00263     79*            DO 215 J=1,NRB                                    000515
00266     80*        215 C(I,J)=AM*A(J,I)+BM*B(J,I)                        000515
00271     81*            GO TO 216                                         000535
00272     82*        214 DO 112 I=1,NCA                                    000553
00275     83*            DO 112 J=1,NRB                                    000553
```

```
00300     84*            T=0.                                                  000553
00301     85*            DO 113 K=1,NRA                                        000555
00304     86*        113 T=T+AM*A(K,I)*BM*B(J,K)                               000555
00306     87*        112 C(I,J)=T                                              000563
00311     88*        216 NRC=NCA                                               000603
00312     89*            NCC=NRB                                               000604
00313     90*            GO TO 200                                             000606
00313     91*      C                                                           000606
00313     92*      C     ERROR OUTPUT                                          000606
00313     93*      C                                                           000606
00314     94*        999 WRITE(6,1000)                                         000610
00316     95*       1000 FORMAT(1H1,19H FAULTY INDEX INPUT)                    000615
00316     96*      C                                                           000615
00316     97*      C     RETURN TO CALLING PROGRAM                             000615
00316     98*      C                                                           000615
00317     99*        200 RETURN                                                000615
00320    100*            END                                                   000727

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
#FOR.IS ,DOTP
FOR SE3B=06/02/77-15:19:40 (.0)

   SUBROUTINE DOTP      ENTRY POINT 000135

   STORAGE USED: CODE(1) 000155: DATA(0) 000022: BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   SQRT
    0004   ACOS
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000   000003 INJP$


00101      1*            COMPILER(XM=1),(ADR=IND)                                    000001
00103      2*            SUBROUTINE DOTP(A,B,C,CAB,AB)                               000001
00103      3*      C                                                                 000001
00103      4*      C     DOT PRODUCT                                                 000001
00103      5*      C                                                                 000001
00103      6*      C     COMPUTES  C = A DOT B  AND  COS(AB)  WHERE                  000001
00103      7*      C         A = A(1)I + A(2)J + A(3)K                               000001
00103      8*      C         B = B(1)I + B(2)J + B(3)K                               000001
00103      9*      C         C = A(1)*B(1) + A(2)*B(2) +A(3)*B(3)                    000001
00103     10*      C         CAB = COS(AB) = COS(ANGLE BETWEEN A AND B)              000001
00103     11*      C         CAB = C/((MAGNITUDE OF A)*(MAGNITUDE OF B))             000001
00103     12*      C                                                                 000001
00103     13*      C     IF A OR B IS THE ZERO VECTOR, CAB IS SET TO 2               000001
00103     14*      C                                                                 000001
00105     15*            DIMENSION A(3),B(3)                                         000001
00106     16*            C=A(1)*B(1)+A(2)*B(2)+A(3)*B(3)                             000001
00107     17*            CAB=2.                                                      000012
00110     18*            AB=SQRT(A(1)*A(1)+A(2)*A(2)+A(3)*A(3))*SQRT(B(1)*B(1)+B(2)*B(2)+B(  000014
00110     19*           13)*B(3))                                                    000014
00111     20*            IF(AB.NE.0.) CAB=C/AB                                       000047
00113     21*            IF(CAB.GT.1..AND.CAB.LT.1.000001)CAB=1.                     000054
00115     22*            IF(CAB.LT.-1..AND.CAB.GT.-1.000001)CAB=-1.                  000074
00117     23*            IF(AB.NE.0.) AB=ACOS(CAB)                                   000114
00121     24*            RETURN                                                      000122
00122     25*            END                                                         000154

       END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
#FOR,I8 ,CR8P
FOR 8E38-06/02/77-15:19:45 (,0)

   SUBROUTINE CR8P       ENTRY POINT 000173

   STORAGE USED: CODE(1) 000222: DATA(0) 000027: BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003    SQRT
    0004    ASIN
    0005    NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000037 115G       0001    000151 137G       0000 R 000000 AMAG       0000 R 000001 BMAG       0000 I 000002 I
    0000    000006 INJP$
```

```
00101      1*            COMPILER(XM=1),(ADR=IND)                                  000002
00103      2*            SUBROUTINE CR8P(A,B,C,BETA,CMAG,I8)                       000002
00105      3*            DIMENSION A(3),B(3),C(3)                                  000002
00105      4*      C                                                               000002
00105      5*      C PROGRAM FORMS CROSS PRODUCT                                   000002
00105      6*      C AND ANGLE BETA.  C NORMED TO CMAG                             000002
00105      7*      C         C = A X B                                             000002
00105      8*      C                                                               000002
00106      9*            C(1)=A(2)*B(3)-A(3)*B(2)                                  000002
00107     10*            C(2)=A(3)*B(1)-A(1)*B(3)                                  000010
00110     11*            C(3)=A(1)*B(2)-A(2)*B(1)                                  000016
00111     12*            AMAG=0.0                                                  000024
00112     13*            BMAG=0.0                                                  000025
00113     14*            CMAG=0.0                                                  000026
00114     15*            DO 10 I=1,3                                               000037
00117     16*            AMAG=AMAG+A(I)*A(I)                                       000037
00120     17*            BMAG=BMAG+B(I)*B(I)                                       000042
00121     18*         10 CMAG=CMAG+C(I)*C(I)                                       000046
00123     19*            AMAG=SQRT(AMAG)                                           000053
00124     20*            BMAG=SQRT(BMAG)                                           000057
00125     21*            CMAG=SQRT(CMAG)                                           000063
00126     22*            BETA=CMAG/(AMAG*BMAG)                                     000067
00127     23*            IF(BETA.GT.1..AND.BETA.LT.1.0001)BETA=1.                  000073
00131     24*            IF(BETA.LT.-1..AND.BETA.GT.-1.0001)BETA=-1.               000113
00133     25*            BETA=ASIN(BETA)                                           000133
00134     26*            IF(I8.EQ.1)RETURN                                         000137
00136     27*            DO 20 I=1,3                                               000151
00141     28*         20 C(I)=C(I)/CMAG                                            000151
00143     29*            RETURN                                                    000154
00144     30*            END                                                       000221

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR,I8 ,INTP
FOR SE3B=06/02/77-15:28:22 (*0)

   SUBROUTINE INTP      ENTRY POINT 000472

   STORAGE USED: CODE(1) 000725: DATA(0) 000125: BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NWDU$
    0004   NIO2$
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000631 1L        0001   000040 102L      0001   000646 11L       0001   000652 111L      0001   000032 112G
    0001   000140 131G      0001   000166 140G      0001   000212 147G      0001   000506 15L       0001   000236 156G
    0001   000620 16L       0001   000120 202L      0001   000146 203L      0001   000174 204L      0001   000220 205L
    0001   000275 22L       0001   000330 33L       0001   000305 44L       0000   000063 444F      0001   000350 5L
    0001   000320 55L       0001   000340 59L       0001   000325 66L       0001   000345 69L       0001   000656 7L
    0001   000377 77L       0000   000062 777F      0001   000435 7AL       0001   000460 79L       0001   000405 80L
    0001   000547 88L       0001   000572 89L       0001   000517 90L       0001   000244 99L       0000 I 000053 I
    0000 I 000057 IA        0000   000100 INJP$     0000 I 000060 J         0000 I 000054 KA        0000 I 000061 KM
    0000 I 000050 LLA       0000 I 000051 MMA       0000 I 000052 NN        0000 I 000056 N1        0000 I 000055 N2
    0000 R 000000 X         0000 R 000024 Y

00101      1*            SUBROUTINE INTP (XA,YA,II,JJ,KK,LL,MM,XX,YY)                    000010
00103      2*            DIMENSION X(20),Y(20),YY(20),XX(20,20)                          000010
00104      3*            LLA=IABS(LL)                                                    000010
00105      4*            MMA=IABS(MM)                                                    000012
00106      5*            NN=MMA+1                                                        000014
00107      6*            IF(KK.LT.0) GO TO 102                                           000016
00111      7*            DO 91 I=1,KK                                                    000021
00114      8*            X(I)=XX(I,1)                                                    000032
00115      9*         91 Y(I)=YY(I)                                                      000033
00117     10*            GO TO 99                                                        000036
00120     11*        102 IF(LL.GT.0.AND.MM.GT.0) GO TO 202                               000040
00122     12*        103 IF(LL.GT.0.AND.MM.LT.0) GO TO 203                               000053
00124     13*        104 IF(LL.LT.0.AND.MM.GT.0) GO TO 204                               000067
00126     14*        105 IF(LL.LT.0.AND.MM.LT.0) GO TO 205                               000103
00130     15*        202 DO 212 I=1,LL                                                   000120
00133     16*            X(I)=XX(I,MM)                                                   000140
00134     17*        212 Y(I)=XX(I,NN)                                                   000141
00136     18*            GO TO 99                                                        000144
00137     19*        203 DO 213 I=1,LL                                                   000146
00142     20*            Y(I)=XX(I,MMA)                                                  000166
00143     21*        213 X(I)=XX(I,NN)                                                   000167
00145     22*            GO TO 99                                                        000172
00146     23*        204 DO 214 I=1,LLA                                                  000174
00151     24*            X(I)=XX(MM,I)                                                   000212
```

```
00152     25*       214 Y(I)=XX(NN,I)                                                    000215
00154     26*           GO TO 99                                                         000216
00155     27*       205 DO 215 I=1,LLA                                                   000220
00160     28*           Y(I)=XX(MMA,I)                                                   000236
00161     29*       215 X(I)=XX(NA,1)                                                    000237
00163     30*           GO TO 99                                                         000242
00164     31*        99 KA=IABS(KK)                                                      000244
00165     32*           IF(XA.GE.X(1).AND.XA.LE.X(KA)) GO TO 5                           000245
00167     33*           IF(XA.LT.X(1)) GO TO 22                                          000264
00171     34*           IF(XA.GT.X(KA)) GO TO 33                                         000270
00173     35*        22 IF(JJ.EQ.-1) GO TO 44                                            000275
00175     36*           IF(JJ.EQ.0) GO TO 55                                             000277
00177     37*           IF(JJ.EQ.1) GO TO 66                                             000301
00201     38*        44 WRITE(6,777)                                                     000305
00203     39*       777 FORMAT(1H1)                                                      000311
00204     40*           WRITE(6,444)                                                     000311
00206     41*       444 FORMAT(//10X,'THE GIVEN VALUE IS OUTSIDE THE DATA RANGE')        000316
00207     42*           GO TO 7                                                          000316
00210     43*        55 N2=2                                                             000320
00211     44*           N1=1                                                             000321
00212     45*           GO TO 1                                                          000323
00213     46*        66 N1=1                                                             000325
00214     47*           GO TO 11                                                         000326
00215     48*        33 IF(JJ.EQ.-1) GO TO 44                                            000330
00217     49*           IF(JJ.EQ.0) GO TO 59                                             000332
00221     50*           IF(JJ.EQ.1) GO TO 69                                             000334
00223     51*        59 N2=KA                                                            000340
00224     52*           N1=KA-1                                                          000341
00225     53*           GO TO 1                                                          000343
00226     54*        69 N2=KA                                                            000345
00227     55*           GO TO 111                                                        000346
00230     56*         5 IA=IABS(II)                                                     000350
00231     57*           IF(IA.EQ.1) IA=IA+1                                              000351
00233     58*           IF(IA.EQ.KA) IA=KA-1                                             000357
00235     59*           IF(II.GT.0) GO TO 77                                             000365
00237     60*           N2=IA                                                            000370
00240     61*           N1=N2-1                                                          000372
00241     62*           J=0                                                              000374
00242     63*           GO TO 89                                                         000375
00243     64*        77 N1=IA                                                            000377
00244     65*           N2=N1+1                                                          000400
00245     66*           J=0                                                              000402
00246     67*           GO TO 79                                                         000403
00247     68*        80 IF(MOD(J,2).EQ.0) GO TO 78                                       000405
00251     69*           N1=N1+J                                                          000411
00252     70*           N2=N1+1                                                          000414
00253     71*           IF(N2.GT.KA.OR.N1.LT.1) GO TO 15                                 000416
00255     72*           GO TO 79                                                         000433
00256     73*        78 N1=N1-J                                                          000435
00257     74*           N2=N1+1                                                          000437
00260     75*           IF(N2.GT.KA.OR.N1.LT.1) GO TO 15                                 000441
00262     76*           GO TO 79                                                         000456
00263     77*        79 IF(XA.GT.X(N1).AND.XA.LT.X(N2)) GO TO 1                          000460
00265     78*           IF(XA.EQ.X(N1)) GO TO 11                                         000477
00267     79*           IF(XA.EQ.X(N2)) GO TO 111                                        000502
00271     80*        15 J=J+1                                                            000506
00272     81*           KM=2*KA                                                          000510
```

```
00273     82*           IF(J.LE.KM) GO TO 80                                                    000513
00275     83*        90 IF(MOD(J+2).EQ.0) GO TO 88                                              000517
00277     84*           N1=N1-J                                                                 000523
00300     85*           N2=N1+1                                                                 000526
00301     86*           IF(N2.GT.KA.OR.N1.LT.1) GO TO 16                                        000530
00303     87*           GO TO 89                                                                000545
00304     88*        88 N1=N1+J                                                                 000547
00305     89*           N2=N1+1                                                                 000551
00306     90*           IF(N2.GT.KA.OR.N1.LT.1) GO TO 16                                        000553
00310     91*           GO TO 89                                                                000570
00311     92*        89 IF(XA.GT.X(N1).AND.XA.LT.X(N2)) GO TO 1                                 000572
00313     93*           IF(XA.EQ.X(N1)) GO TO 11                                                000611
00315     94*           IF(XA.EQ.X(N2)) GO TO 111                                               000614
00317     95*        16 J=J+1                                                                   000620
00320     96*           KM=2*KA                                                                 000622
00321     97*           IF(J.LE.KM) GO TO 90                                                    000625
00323     98*         1 YA=(Y(N2)-Y(N1))*(XA-X(N1))/(X(N2)-X(N1))+Y(N1)                         000631
00324     99*           GO TO 7                                                                 000644
00325    100*        11 YA=Y(N1)                                                                000646
00326    101*           GO TO 7                                                                 000650
00327    102*       111 YA=Y(N2)                                                                000652
00330    103*           GO TO 7                                                                 000654
00331    104*         7 RETURN                                                                  000656
00332    105*           END                                                                     000724

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

6. REFERENCES

1. Widmayer, E., Tanner, A. E., Klump, R., "Crashworthiness Analysis of the UMTA State-of-the-Art Cars," Boeing Vertol Company Final Report D174-10034-1 (DOT-TSC-791-3), June 1975

2. Cassidy, R. J., Romeo, D. J., "Assessment of Crashworthiness of Existing Urban Rail Vehicles," Calspan Corporation Final Report ZL-5368-K-1, June 1975

3. Raidt, J. B. (Principal Investigator), "A Preliminary Study of Vertical Motions During Impact," Pullman-Standard Research Project 38-1853, Final Report, August 1972

4. Hohenemser, K. H., Diboll, W. B., Yin, S. K., Szabo, B. A., "Computer Simulation of Tank Car Head Puncture Mechanisms Classification Yard Accidents," School of Engr and Appl Sci, Washington University Preliminary Report 64274 (FRA-ORD&D-75-23), February 1975

U.S. DEPARTMENT OF TRANSPORTATION
RESEARCH AND SPECIAL PROGRAMS ADMINISTRATION
TRANSPORTATION SYSTEMS CENTER
KENDALL SQUARE, CAMBRIDGE, MA. 02142

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE, $300

POSTAGE AND FEES PAID
U.S. DEPARTMENT OF TRANSPORTATION
513

U.S MAIL