SSC-216

PROGRAM "TRANSHIP"
A COMPUTER PROGRAM FOR THE DESIGN OF THE MIDSHIP SECTION OF A TRANSVERSELY-FRAMED DRY CARGO SHIP

PART TWO

SHIP STRUCTURE COMMITTEE

1971

SHIP STRUCTURE COMMITTEE

AN INTERAGENCY ADVISORY
COMMITTEE DEDICATED TO IMPROVING
THE STRUCTURE OF SHIPS

MEMBER AGENCIES:

UNITED STATES COAST GUARD
NAVAL SHIP SYSTEMS COMMAND
MILITARY SEALIFT COMMAND
MARITIME ADMINISTRATION
AMERICAN BUREAU OF SHIPPING

ADDRESS CORRESPONDENCE TO:

SECRETARY
SHIP STRUCTURE COMMITTEE
U.S. COAST GUARD HEADQUARTERS
WASHINGTON, D.C. 20591

SR-175
1972

Dear Sir:

The Ship Structure Committee has undertaken a series of research projects to develop analytical methods and computer programs which will apply modern high-speed electronic computational techniques to ship hull structures.

This report contains the details of the computer program discussed in SSC-215, A Guide for the Synthesis of Ship Structure--Part One--The Mid-Ship Hold of a Transversely Framed Dry Cargo Ship.

Comments on this report would be welcomed.

Sincerely,

W. F. REA, III
Rear Admiral, U. S. Coast Guard
Chairman, Ship Structure Committee

SSC-216

Final Technical Report

on

Project SR-175, "Rational Ship Structural Design"

PROGRAM "TRANSHIP"

A COMPUTER PROGRAM FOR THE DESIGN OF THE MIDSHIP SECTION OF A TRANSVERSELY-FRAMED DRY CARGO SHIP

PART TWO

by

Manley St. Denis
University of Hawaii

under
Department of the Navy
Naval Ship Engineering Center
Contract No. N00024-68-C-5403

This document has been approved
for public release and sale;
its distribution is unlimited.

U.S. Coast Guard Headquarters
Washington, D.C.
1971

ABSTRACT

This report presents the computer program corresponding to the method of design expounded in the Ship Structure Committee Report SSC-215, *A Guide for the Synthesis of Ship Structures - Part One - The Midship Hold of a Transversely-Framed, Dry Cargo Ship*. The program consists in an executive routine, called *TRANSHIP*, and twenty seven subroutines.

INDEX

INDEX (Cont'd)

INDEX (Cont'd)

SHIP STRUCTURE COMMITTEE

The SHIP STRUCTURE COMMITTEE is constituted to prosecute a research program to improve the hull structures of ships by an extension of knowledge pertaining to design, materials and methods of fabrication.

RADM W. F. Rea, III, USCG, Chairman
Chief, Office of Merchant Marine Safety
U.S. Coast Guard Headquarters

Capt. J. E. Rasmussen, USN
Head, Ship Systems Engineering and Design Department
Naval Ship Engineering Center
Naval Ship Systems Command

Mr. K. Morland, Vice President
American Bureau of Shipping

Mr. E. S. Dillon
Chief
Office of Ship Construction
Maritime Administration

Capt. L. L. Jackson, USN
Maintenance and Repair Officer
Military Sealift Command

SHIP STRUCTURE SUBCOMMITTEE

The SHIP STRUCTURE SUBCOMMITTEE acts for the Ship Structure Committee on technical matters by providing technical coordination for the determination of goals and objectives of the program, and by evaluating and interpreting the results in terms of ship structural design, construction and operation.

NAVAL SHIP ENGINEERING CENTER

Mr. P. M. Palermo - Chairman
Mr. J. B. O'Brien - Contract Administrator
Mr. G. Sorkin - Member
Mr. H. S. Sayre - Alternate
Mr. I. Fioriti - Alternate

U.S. COAST GUARD

LCDR C. S. Loosmore, USCG - Secretary
CDR C. R. Thompson, USCG - Member
CDR J. W. Kime, USCG - Alternate
CDR J. L. Coburn, USCG - Alternate

MARITIME ADMINISTRATION

Mr. F. Dashnaw - Member
Mr. A. Maillar - Member
Mr. R. Falls - Alternate
Mr. R. F. Coombs - Alternate

MILITARY SEALIFT COMMAND

Mr. R. R. Askren - Member
LTJG E. T. Powers, USNR - Member

AMERICAN BUREAU OF SHIPPING

Mr. S. G. Stiansen - Member
Mr. F. J. Crum - Member

OFFICE OF NAVAL RESEARCH

Mr. J. M. Crowley - Member
Dr. W. G. Rauch - Alternate

NAVAL SHIP RESEARCH & DEVELOPMENT CENTER

Mr. A. B. Stavovy - Alternate

NATIONAL ACADEMY OF SCIENCES - Ship Research Committee

Mr. R. W. Rumke, Liaison
Prof. R. A. Yagle, Liaison

SOCIETY OF NAVAL ARCHITECTS & MARINE ENGINEERS

Mr. T. M. Buermann, Liaison

BRITISH NAVY STAFF

Dr. V. Flint, Laision
CDR P. H. H. Ablett, RCNC, Liaison

WELDING RESEARCH COUNCIL

Mr. K. H. Koopman, Liaison
Mr. C. Larson, Liaison

INTRODUCTION

This report presents the computer program corresponding to the method of design expounded in the Ship Structure Committee Report SSC-215, "A GUIDE FOR THE SYNTHESIS OF SHIP STRUCTURES - PART ONE - THE MIDSHIP HOLD OF A TRANSVERSELY FRAMED, DRY CARGO SHIP". The program consists in an executive routine, called TRANSHIP, and twenty seven subroutines arranged in alphabetical order. These have the following names:

1	ASPECT
2	BNMAT
3	COSTING
4	DOCUMENT
5	EV
6	FRAME
7	FSHAPE
8	GEOM
9	GRILLAGE
10	INPUT
11	INTERMED
12	LONGIT
13	LONGMAT
14	MATINV
15	NTPLATE
16	PLATING
17	REACT
18	RTPLSUB
19	SECTION
20	SHAPES
21	SOLVE
22	STEP
23	STRESS
24	T
25	THETA
26	TRANSV
27	XLOAD

For each routine, the following items are given:

a) Description

b) Notation

c) Listing

d) Flow Chart

Appreciation is expressed for the help received by Mrs. Linda Drake for documentation of the program.

DESCRIPTION OF INPUT

Input data are introduced as follows:

<u>1st Card</u>	Format (9 I 5)
IK = 1	Center Keel only
IK = 2	2 Side Keelsons plus Center Keel
IK = 3	4 Side Keelsons plus Center Keel
IK = 4	6 Side Keelsons plus Center Keel
IPILL = 0	No pillars
IPILL = 1	Pillars included
IHOG = 0	Sag case
IHOG = 1	Hog case
NI	Number of iterations
<u>2nd Card</u>	Format (2I2, 8F8, 2)
NDECKS	Number of decks (double bottom excluded)
KPANELS	Number of bays
TLENGTH	Ship length(ft)
BEAM	Ship beam (ft)
DRAFT	Ship draft (ft)
HMAIN	Ship depth (ft)
XLHOLD	Length of hold (ft)
XLHATCH	Length of hatchway (ft)
HATCH	Width of hatchway (ft)
HFLOOR	Height of floor (ft)
<u>3rd Card</u>	Format (7F 10.3)
TWEENH	'Tween deck height (ft)
<u>4th Card</u>	Format (7F 10.3)
DKLO	Uniformly distributed deck loading (lb in^{-2})
<u>5th Card</u>	Format (3F 15.2)
YIELD	Elastic limit of material (lb in^{-2})
E	Young's modulus of material (lb in^{-2})
PRM	Criterion of primary stress intensity (lb in^{-2})
<u>6th Card</u>	Format (7F 10.3)
HRATE	Labor rate (dollars hr^{-1})
DOLLPP	Plate cost (dollars lb^{-1})
WPRICE	Welding cost (dollars lb^{-1})
TWF	Time-weight factor (dollars lb^{-1})
SG	Material density (lb in^{-3})
<u>7th Card</u>	Format (F 10.2)
A	*Frame spacing (in)*

Sample input values used in the computer runs for the WOLVERINE STATE are as follows:

2	0	1	10						(1)
4	1	496.	71.5	30.	43.5	100.	33.	20.	5. (2)
9.62	9.62	9.63	9.63	0.					(3)
17.76	15.54	2.1	2.1	2.1	1.78				(4)
35000.	30000000.	19000.							(5)
4.	0.065	3.	0.02	0.283					(6)
30.									(7)

Program TRANSHIP

Tranship is the executive routine. It calls all the primary subroutines as needed and makes any computation necessary for transition between them. A general Flow diagram of all routines is given starting on page 17. A detailed flow chart of TRANSHIP begins on page 19. A listing of terms common to all the routines precedes the flow diagrams for TRANSHIP, while those specific to a given subroutine are included with the description of that subroutine. In all cases, the appropriate mathematical symbol from the basic report SSC-215 are given when applicable.

Notation
Mathematical Symbols to FORTRAN

Mathematical Symbol	Definition	FORTRAN Term
a	Spacing of frames and floors, or side of plate rectangles (input)	A
	Frame spacing	AAA
b	Distance between longitudinals	B
	Effective width of plating	EFFW
b_e	Effective breadth or width of plating	EFFB1, EFFW1, EFFBR
d	Height of floors (input)	HFLØØR
$f(A_i)$	Time - size factor	F4(H)
$f_1(h)$	Weight per inch of vee weld	F1(H)
$f_2(h)$	Weight per inch of continuous double fillet weld	F2(H)
$f_3(h)$	Weight per inch of intermittent double fillet weld	F3(H)
$f(w_i)$	Time - weight factor	TWr
h	Thickness of plating	THIKK, THIKK1
	Wave height	WAVEH
r	Hourly rate of labor (input)	HRATE
$A_{i,j}$	Grillage matrix	AX
A_{pe}	Effective area of plating	APE
B	Beam of ship (input)	BEAM
$B_p(x)$	Boundary matrix used in grillage calculation	B
C	Combined cost of plating, labor and welding per inch of ship length	CØSTS
C_{fe}	Total labor cost of erection and fabrication	XLABOR
C_m	Total material cost	PLCØST

Mathematical Symbol	Definition	FORTRAN Term
C_w	Total cost of welding	TWELD
D	Depth of ship (input)	HMAIN
	Flexural rigidity of plating	FLEXR
E	Young's modulus (input)	E
H	Draft of ship (input)	DRAFT
I	Second central moment of area	SMAF
I_{pe}	Second central moment of effective plating (about the faying surface)	XIPE
I_{pf}	Second central moment of plate-frame combination	XIPFX
I_t	Second central moment of transference	XIT
K	Stiffness factor	XJ
L	Ship length (input)	TLENGTH
N_s	Number of weld seams in inner bottom	NSEAM
S	Total weight of hull structure per inch	WBAYTOT
$\alpha_{i,j}$	Influence coefficients	ALFA
β	Aspect ratio parameter	BE, B2
γ	Specific weight of material (input)	SG
θ	Joint rotation	T
λ	Normalized effective breadth	E1, E2
π	Circle ratio	PIE
σ^*_1	Criterion for primary stress intensity (input)	PRM
σ_1	Primary stress intensity (design stress intensity)	PSL
σ_{2x}	Secondary stress intensity in longitudinal direction	SESL
σ_{2y}	Secondary stress intensity in transverse direction	SEST

Mathematical Symbol	Definition	FORTRAN Term
σ_{2xf}	Secondary stress intensity in the longitudinal direction in the flange	FSESL1
σ_{2yf}	Secondary stress intensity in the transverse direction in the flange	FSEST1
σ_{3x}	Tertiary stress intensity in longitudinal direction	TESL
σ_{3y}	Tertiary stress intensity in transverse direction	TEST
σ_{cr}	Critical buckling stress intensity	CRITB
σ'_{xb}, σ_{xb}	Maximum bending stress intensities in the longitudinal direction (at the middle of the sides) when the compressive stresses are of such magnitude as to cause buckling	SIXB
σ''_{xb}	Maximum bending stress intensities in the longitudinal direction when there is no compressive stress	SXBP
σ'_{yb}, σ_{yb}	As SIXB, but in transverse direction	SIYB
σ''_{yb}	Maximum bending stress intensities in the transverse direction when there is no compressive stress	SYBP
φ_x	Factor depending on aspect ratio in equation for SXBP	PHIL
φ_y	Factor depending on aspect ratio in equation for SYBP	PHIS
χ_{ps}	Distance of neutral axis of plate-frame combination from center of plate	CHI

Notation
FORTRAN Symbols to Mathematical

FORTRAN Term	Definition	Mathematical Symbol
A	Spacing of frames and floors, or side of plate rectangles (input)	a
AA	Storage location for A	
AAA	Frame spacing	a
AFAC	Scale factor = 1.0	
ALFA	Influence coefficients	$\alpha_{i,j}$
ALØAD	Loading (life and/or hydrostatic load)	
AREAT1	Cross-sectional area of transverse material	
AX	Grillage matrix	$A_{i,j}$
B	Distance between longitudinals	b
B	Boundary matrix used in grillage calculation	$B_p(x)$
BB	Storage location for B	
BEAM	Beam of ship (input)	B
CC	Coefficients of characteristic equation in descending order	
CN	Determinant of grillage matrix	
CØDE	Code = 1 fixity of plating Code = - 1 simple support	
DEFL	Deflection	
DKLØ	Uniformly distributed deck loading (input)	
DØLLPP	Dollar per pound of plating material (input)	
DRAFT	Draft of ship (input)	H
E	Young's modulus (input)	E
EFFB1	Effective breadth of plating	b_e
EFFW1	Effective width of plating	b_e

FORTRAN Term	Definition	Mathematical Symbol
FIXCØDE	Flag denoting ideal fixities of the inner-bottom structure (= - 1 fixed support; = 1 simple support)	
FSECLØ	Frame section modulus in longitudinal direction	
FSECMØD	Required section modulus	
FSECTR	Frame section modulus in transverse direction	
FSESL	Maximum allowable secondary stress intensity in longitudinal direction in flange	
FSESL1	Secondary stress intensity in the longitudinal direction in the flange	σ_{2xf}
FSEST	Maximum allowable secondary stress intensity in transverse direction in flange	
FSEST1	Secondary stress intensity in the transverse direction in the flange	σ_{2yf}
GP	Grillage pressure load	
GNØT	Number of oil tight double bottom longitudinals	
GNNT	Number of non-tight double bottom longitudinals	
HATCH	Width of hatchway (input)	
HF1(I)	Plate thickness if the plating is fixed along the edges of the plate rectangle (I = 1 indicates bottom shell, I = 2 indicates inner bottom)	
HFLØØR	Height of floors (input)	d
HMAIN	Depth of ship (input)	D
HNEUT	Height of neutral axis of the midship section	
HNTX	Height of neutral axis of beams	
HRATE	Hourly rate of labor (input)	r

FORTRAN Term	Definition
HS1(I)	Plate thickness if the plating is simply supported along the edges of the plate rectangle (I = 1 indicates bottom shell, I = 2 indicates inner bottom)
HWF(L)	Thickness of oil tight floors (L = 1) or oil tight double bottom longitudinals (L = 2), fixed along edges
HWFS(L)	Thickness of oil tight floors (L = 1) or oil tight double bottom longitudinals (L = 2), simply supported along one long edge fixed along the other
HWS(L)	Thickness of oil tight floors (L = 1) or oil tight double bottom longitudinals (L = 2), simply supported along edges
IBT	Limiting index on innerbottom
IHØG	Code = 0 Sag case Code = 1 Hog case
IK	Code = 1 Center keel only (input) Code = 2 2 Side keelsons plus center keel Code = 3 4 Side keelsons plus center keel Code = 4 6 Side keelsons plus center keel
ILG	Limiting index on side shells
IPILL	Code = 0 No pillars Code = 1 Pillars included
IPLATE	Number of plating items
IPLT	Code = 1 Bilge Code = 2 O. T. Floors Code = 3 N. T. Floors

FORTRAN Term	Definition	Mathemati Symbol
LLGX	Code = 1 denotes longitudinals Code = 0 denotes transverses	
LONGLAB	Label For Longitudinals 1 = Keel 2 = Side girders 3 = Fourth deck 4 = Third deck 5 = Second deck 6 = Main deck	
NBBG	Code = 1 corresponds to 1 double bottom longitudina 2 corresponds to 3 double bottom longitudina 3 corresponds to 5 double bottom longitudina 4 corresponds to 7 double bottom longitudina	
NDECKS	Number of decks (input)	
NG	Number of longitudinal girders	
NNT	Number of non-tight longitudinals	
NØC	Code = 1 Unrestrained frame deflections at girder intersections 2 Influence coefficients 3 Frame bending moment distribution	
NØT	Number of oil tight longitudinals	
NØTS	Code = 1 Bulkhead Bay = 2 Hatch Bay	
NSB	Number of unsupported frames in bulkhead bay	
NSH	Number of frames in hatch bay	
NSTT	Number of non-tight floors	
PIE	Circle ratio	π
PLATLAB	Label For Plate Elements 1 = Double bottom 2 = Inner bottom 3 = Fourth deck 4 = Third deck 5 = Second deck 6 = Main deck 7 = First side strake 8 = Second side strake	

FORTRAN Term	Definition	Mathematical Symbol
	9 = Third side strake 10 = Top side strake 11 = Bilge 12 = Oil-tight floor 13 = Non-tight floor 14 = Oil-tight longitudinal 15 = Non-tight longitudinal	
PPP	Concentrated loading	
PRHEAD	Pressure head	
PRM	Criterion for primary stress intensity (input)	σ_1^*
PSL	Primary stress intensity (design stress intensity)	σ_1
RABI	Bilge radius	
RL	Output argument from EV for adjoint of matrix	
RØØT	Array of real roots of characteristic equation	
SESL	Secondary stress intensity in longitudinal direction	σ_{2x}
SEST	Secondary stress intensity in transverse direction	σ_{2y}
SG	Specific weight of material (input)	γ
SHMAX	Critical buckling shear stress	
SMAF	Second central moment of area	I
SMAL1	Second central moment of area of longitudinal girders	
SMAT1	Second central moment of area of transverse framing	
SMØDL1	Section modulus to the plate of longitudinal girders	
SMØDL2	Section modulus to the flange of longitudinal girders	
SMØDT1	Section modulus to the plate of transverse framing	
SMØDT2	Section modulus to the flange of transverse framing	

FORTRAN Term	Definition	Mathematical Symbol
SS	Spacing of stiffeners	
SUMAREA	Total cross-sectional area of plating	
SUMAREAL	Total cross-sectional area of all webs of longitudinals	
SUMAREAP	Total cross-sectional area of plating of a ship section	
SUMML	Total first moment of area of longitudinals about neutral axis of ship section (neutral axis from previous iteration or assumed)	
SUMMØM	Total first central moment of area of ship section	
SUMMP	Total first moment of area of plating material about neutral axis of cross section	
SUMSMA	Total second central moment of area of ship section	
SUMSMA1	Correct second central moment of area of ship section	
SUMSMAL	Total second central moment of area of longitudinals of ship section	
SUMSMAP	Total second central moment of area of plating of ship section	
TESL	Tertiary stress intensity in longitudinal direction	σ_{3x}
TEST	Tertiary stress intensity in transverse direction	σ_{3y}
THIKK1	Thickness of plating	h
TLENGTH	Ship length (input)	L
TRANLAB	Label For Transverse Elements 1 = Oil-tight floor 2 = Non-tight floor 3 = Fourth deck beam 4 = Third deck beam 5 = Second deck beam 6 = Main deck beam 7 = Innerbottom to fourth deck frame 8 = Fourth deck to third deck frame 9 = Third deck to second deck frame 10 = Second deck to main deck frame	

FORTRAN Term	Definition	Mathematical Symbol
TWEENH	'Tween deck height (input)	
TWF	Time-weight factor (input)	$f(w_i)$
V	Array of shear coefficients	
VS	Array of slope coefficients	
VV	Array of moment coefficients	
VY	Array of deflection coefficients	
WAVEH	Wave height	h
WEBL1	Web heights of longitudinals	
WPRICE	Cost of weld material per pound (input)	
XFL	Length of stiffener (between joints)	
XII	Maximum field bending on longitudinal elements	
XKHØG	Hog coefficient in equation for wave bending moment	
XKSAG	Sag coefficient in equation for wave bending moment	
XLHATCH	Length of hatchway (input)	
XLHØLD	Length of hold between two bulkheads (input)	
XLL	Length of longitudinal girder	
XLPANEL	Length of bay	
XMANH	Number of man hours per square foot of the "equivalent surface" of the structure	
XMMT	Maximum field bending moment on transverse elements	
XNG	Number of double bottom longitudinals	
XNST	Number of non-tight floors or frames per bay	
YIELD	Yield stress intensity (input)	

FORTRAN Term	Definition
ZL	Vertical coordinate of centroid of longitudinals
ZP	Vertical coordinate of platings

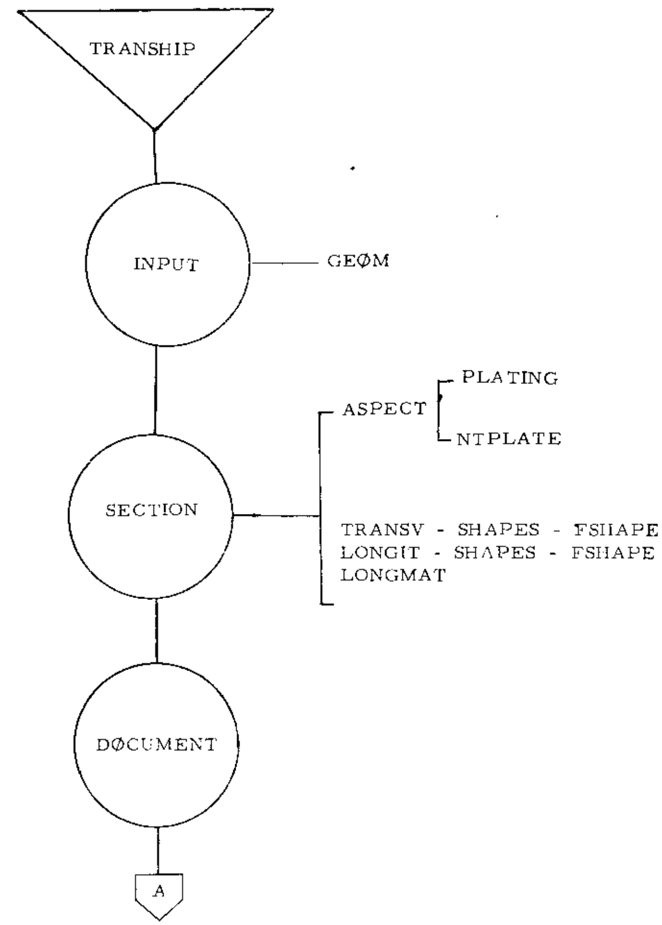
```
      PROGRAM TRANSHIP
      COMMON/TLEN/  TLENGTH    ,IHOG ,RABI,XKSAG,XKHOG
      DIMENSION HF1(2),HS1(2),HWF(2),HWS(2),CRF(2),CRS(2),HWFS(2),
     1CR1(2),H11(2),RC1(2),HWI1(2),H12(2),CR2(2),HWI2(2)
      COMMON/DK/ DKLO(10)
      COMMON/LABEL/ TRANLAB(10),PLATLAB(15),LONGLAB(6)
      COMMON/SH/   SHMAX
      COMMON/DE/ DEF(20)
      COMMON/COST/ HRATE,DOLLPP,WPRICE,TWF  ,SG
      COMMON/LO/FSECLO(12),FSESL(17),FSESL1(17),SMODL2(17,2,3),PSLL(17)
      COMMON/TR/ FSECTR(17),FSEST(17),FSEST1(17),SMODT2(17,2,3),SMODT3(
     1 17,2,5)
      COMMON /C/ TWELD
      COMMON/F/ HF1,HS1,HWF,HWS,HWFS,H12,HWI2,IFL,IIL
      COMMON/H/ XMMT(17)
      DIMENSION NSA(5),PPP(9),WZ(9),AXB(15,15)
      COMMON /E/ IBT,E
      COMMON /D/   VS(15),VY(25)
      DIMENSION   SEST1(17),SESL1(17)
      COMMON /B/ A,B,NDECKS,IPLATE,PSL,PRM,HMAIN,HNEUT,TESL,TEST,SESL,
     1     SEST,          KPANELS,BEAM,XNG,HATCH,THIKK,THIKK1,EFFBR,EFFB1,
     2     EFFW,EFFW1,HFLOOR,TWEENH,THIKX,EFBR,WEB,WEB1,XIPFX,CHI,
     3     XIS, AREAT1,SMAL1,SMODL1,WEBL1,SUMAREAP,SUMAREAL,SUMMP,SUMML,
     4     AREA,SUMAREA,SUMMOM,WPL,XPANEL,WBAYLP,XNST,NST,ITR,
     5     WTR,BAYTR,SUMWT,SUMWTR,LLGX,  YIELD,WBAYTR,XLABOR,PLCOST,
     6     COST,COSTMIN,GNOT,GNNT,KNEUT,SUMSMAP,SUMSMAL,SUMSMA,CODE,
     7WAVEH,PRHEAD,DRAFT,ALOAD,XLHOLD,XLPANEL,AAA,BBB,AA,BB,DELTESL,
     8     SUMSMA1,ZP,ZL,ILG,AREAL1,ASX,  SMAT1,SMODT1
      COMMON/G/ FIXCODE
      COMMON  ALFA(15,15),AX( 9,9 ),CC(10),CN,DEFL( 9),GP,H,JJ,K,NG,NS,
     1PD(15,40), ROOT(9),S(15),SS,V(15),VV(15),W(15),X,XID(15),XII(40),
     2XL,XLL,XML(15),XMD(15),XX(40),RR,SCAL,AFAC,NNN
      DIMENSION ZP(17),SESL(17),SEST(17),TESL(17),TEST(17),XNG(7),THIKK(
     117),       THIKK1(17,2,5),EFFBR(17),EFFW(17),EFFB1(17,2,5),EFFW1(17
     2,2,5),AREA(17,5),TWEENH(5),SMAT1(12,2,5),SMODT1(12,2,5),THIKX(12,2
     3),EFBR(12,2),AREAT1(12,2,5),WEB1(12,2,5),SMAL1(17,2,3),SMODL1(17,2
     4,3),WEBL1(7,2,3),AREAL1(7,2,3),SUMAREAP(5),SUMAREAL(5),SUMMP(5),SU
     5MML(5),SUMAREA(5),SUMMOM(5),WPL(5),XLPANEL(5),SUMWTR(5),WTR(12),ZL
     6(7),TIKK1(17,5),CODE(17),ALOAD(17),SUMSMAP(5),SUMSMAL(5),SUMSMA(5)
      COMMON/GR/ B1(9,9),B2(9,9),B3(9,9),XL1(9),XL2(9),Z(12),DL(17),
     1  ITEST,IPILL,NOOS,NSH,NSB,NSS,D,W1,HB,N,NOTS,NM,NBBG,NJ,XFL(15),
     2  SMAF(15)
      DATA(LONGLAB=48HKEEL      S GIRDER4TH DECK3RD DECK2ND DECKMAIN DK )
      DATA( TRANLAB=80HOT FLOORNT FLOOR4TH DECK3RD DECK2ND DECKMAIN DK I
     1,B,-4TH4TH-3RD 3RD-2ND 2ND-MAIN)
      DATA( PLATLAB=120HB, SHELLIN BUTT,4TH DECK3RD DECK2ND DECKMAIN DK
     1STRAKE 4STRAKE 3STRAKE 2S STRAKEB STRAKEOT FLOORNT FLOOROT LONG,NT
     2 LONG,)
 9991 FORMAT(*  TEST ZONE *, F10,1/)
      NOIT=0
      RABI=120,0
      READ 9218, IK, IPILL, IHDC, NI I
 9218 FORMAT(9I5  )
      CALL        INPUT(BENDING,NBBG, IK ,NJ,XFL,XII ,NSH,NSB,NG)
      NBM=NBBG
      IF(NBBG,EQ,1) NBM=2
```

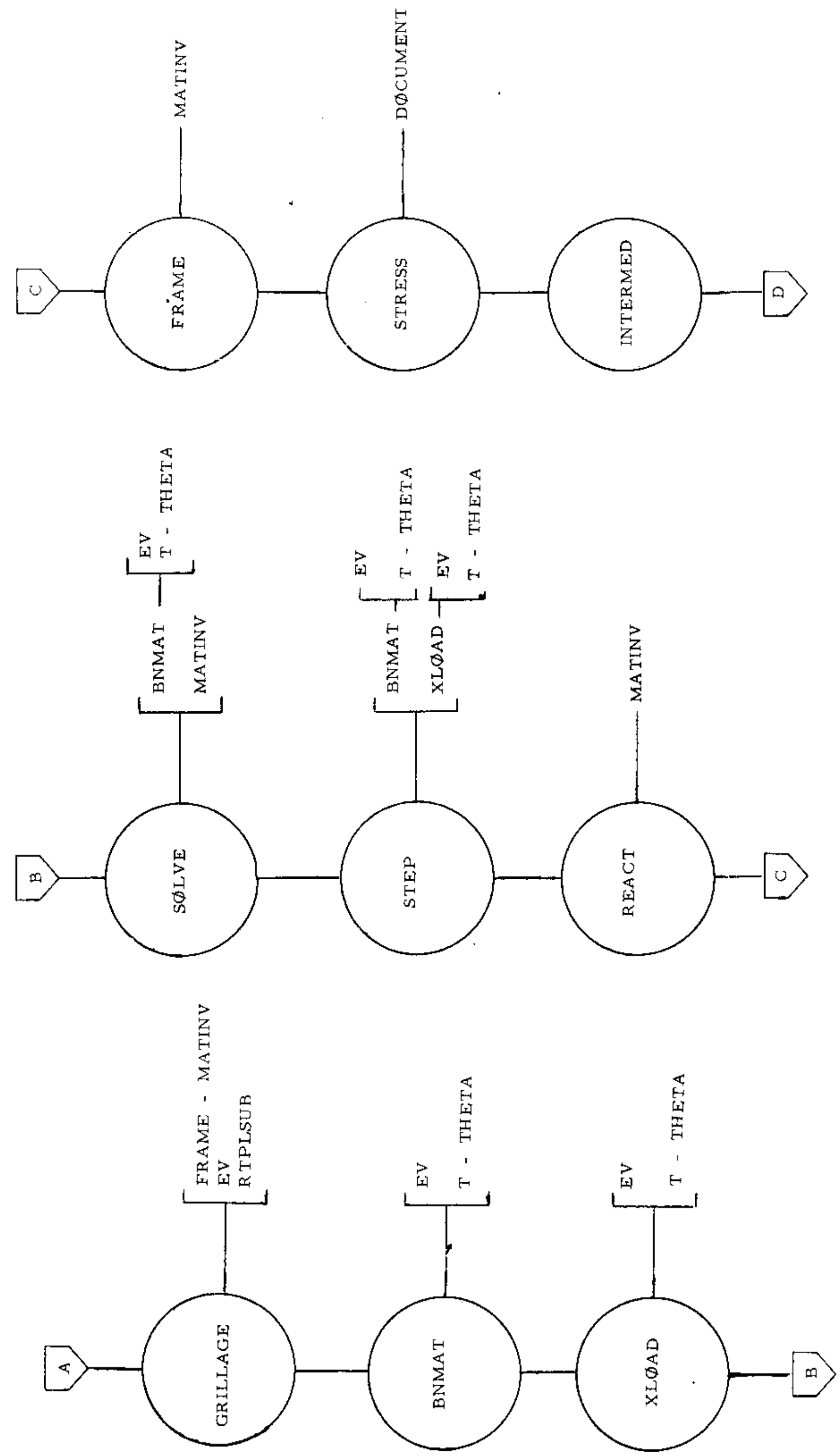
```
C     SECTION MODULUS CALCULATION
      PRMC=PRM
      PRM=19000,
      SMDNEC  = BENDING / PRM
      NOOS=0
        KOUNT=0
       FIXCODF=-1,0
       NX2=1
       CODE(7) = 1,0
  348  CONTINUE
       KNEUT = 0
  347  CALL  SECTION( KNEUT, SMDNEC,KOUNT,BENDING,NX2)
       CALL DOCUMENT
       PRINT 701
  701  FORMAT(1H0,* TRANS,         SMAT1     SMODT1     SMODT2       AREAT1     E
     1FFB1          ZP          ALOAD *,//)
       DO 702 I=1,10
  702  PRINT 705 ,TRANLAB(I),  SMAT1(I,1,1),SMODT1(I,1,1),SMODT2(I,1,1),
     1 AREAT1(I,1,1),EFFB1(I,1,1),ZP(I) ,ALOAD(I)
       PRINT  703
  703  FORMAT(1H0,* LONG,          SMAL1     SMODL1     SMODL2       AREAL1     E
     1FFW1          ZL *,//)
       DO 704  I=1,6
  704  PRINT 705 ,LONGLAB(I),SMAL1(I,1,1),SMODL1(I,1,1),SMODL2(I,1,1),
     1 AREAL1(I,1,1),EFFW1(I,1,1),ZL(I)
  705  FORMAT(*   *, A8,7F10,2)
       NX2=0
       PRM1=PRM
       IF(HNEUT,LE, 0,5 * HMAIN) GO TO 11
       PRM1=PRM * (HMAIN-HNEUT) /HNEUT
   11  DO 9547 I=1,10
       PSLL(I)=PRM1*(HNEUT-ZP(I))/(HMAIN-HNEUT)
       PSLL(I)=ABSF(PSLL(I))
       IF( ZP(I) ,LT, HNEUT ) PSLL(I)=PSLL(I)*XKHOG/XKSAG
 9547  CONTINUE
C
C     GRILLAGE JOINT CALCULATION
 3000  CONTINUE
  351   CONTINUE
        NSS=3
 9871  CONTINUE
       CALL GRILLAGE              (NDECKS, HATCH,PRHEAD,BEAM,HFLOOR,AAA,
     1 ALOAD,SMAT1,SMAL1  )
       DO 3050 LLL=1,NG
       IF(ROOT(LLL)) 2400,3050,3050
 2400  WRITE OUTPUT TAPE 6,9144,J,ROOT(J)
       GO TO  9999
 3050   CONTINUE
 3060  PRINT 9139, (ROOT(J),J=1,NG)
C
C     GRILLAGE MATRICES
C
       X = XLL
  800  CALL BNMAT(AX, 1, B1)
```

```
      CALL BNMAT(AX, 2, B2)
      CALL BNMAT(AX, 3, B3)
 1120 CALL XLOAD (AX, 2, XL2,XLL)
      CALL XLOAD (AX, 1, XL1,XLL)
      NSS=NSS-1
      IF(NSS.LT.1)  GO TO 3605
 3609 CONTINUE
 1410 CALL SOLVE (B1,B2,B2,B3, XL1, XL2, NODS,NSS,NBM ,IPILL)
       IF(NSS-1)3601,5950,3601
 5950 CONTINUE
      DO 1701 I=1,NG
      VV(I)=-VV(I)/E/XID(I)
       VS(I)=-VS(I)
      V (I)=V (I)/E/XID(I)
 1701 CONTINUE
      GO TO 1460
 3606 DO 3607 I=1,NG
      V (I)=-V(I)/E/XID(I)
      VS(I)=-VS(I)
      VV(I)= VV(I)/E/XID(I)
 3607 CONTINUE
 3601 IF(NSS.GT.0)  GO TO 9871
 1460 CONTINUE
C
C       GRILLAGE MEMBER CALCULATION
      NS3=4
      IF(NSS.EQ.0.AND.NG.GT.6)  NS3=2
      NS2=NS+2
      NSA(1)=1
      NSA(2)=2
      NSA(3)=NS2-2
      IF(NSS.EQ.0)  NSA(3)=NS2/2+1
      NSA(4)=NS2
      DO 3605 J=1,NS3
      KK=NSA(J)
      JJ=KK-1
      XJJ=FLOATF(JJ)*SS
      PRINT 76,XJJ
   76 FORMAT(1H1,*   STEP LOCATION  = *,F10.2  ,/)
      IF(NSS.EQ.0)  PRINT 78
   78 FORMAT(/,*  HATCH  BAY  *,/)
      IF(NSS.EQ.1)  PRINT 77
   77 FORMAT(/, *  BULKHEAD  BAY  *,/)
      CALL STEP(AX,KK,NS2,XJJ,NBM)
      IF(KK-NS2) 3604,3605,3605
 3604 CONTINUE
       DO 9027 I=1,NG
       DO 9027 K=1,NG
      AXB(I,K)=ALFA(I,K)
 9027 CONTINUE
       CALL REACT (NG,AXB,PPP,W,DEFL)
      SHMAX=0.
      DO 8319 I=1,NBBG
 8319 IF(SHMAX.LT. ABS(PPP(I)))  SHMAX=ABS(PPP(I))
      NOC =3
      N1=NM
      IF(NBHG.GT.1)  GO TO 9736
      PPP(6)=PPP(5)
      PPP(5)=PPP(4)
      PPP(4)=PPP(3)
      PPP(3)=PPP(2)
      PPP(2)=0.
      N1=NG+1
 9736 CALL FRAME(SMAF,NJ,XFL,DL,WI,D,Z,HB,PPP,Q,E,N1,NUTS,NOC,WZ,KK,NBM
 3605 CONTINUE
 3001 CONTINUE
      IF(NSS.LE.0)   GO TO 3615
      DO 8301 I=1,NG
      V (I)=V (I)*E*XID(I)
      VV(I)=VV(I)*E*XID(I)
 8301 CONTINUE
       GO TO 3601
 3615 CONTINUE
      XJA=SS*(NSH+1)
       IF(NSS.EQ.0.AND.NG.GT.6)  CALL STEP(AX,NS2,NS2,XJA,NBM)
C
C     STRESS CALCULATION
      CALL  STRESS(            N,NN)
      IF(NN) 8402,8402,323
 8402  IF(N) 8400,8400,323
 323  KKM = KKM + 1
       KKMM = 4
      IF(KKM - KKMM)324,324,8400
 324  PRINT 326, KKM
      SESL(2) = SESL(1)
      SEST(2) = SEST(1)
       NOIT=NOIT+1
      IF(NOIT.GE.NI)  GO TO 8400
      GO TO 347
 8400 CONTINUE
      PRINT 326, KKM
      K = 1
      L = 1
C      INTERMEDIATE FIXITY CALCULATION
      PRINT 2221 ,FIXCODE
 2221 FORMAT( * FIXCODE = *,F4.1/)
 2222 FORMAT( * *,A8,F4.1/)
      DO 9632 I=1,15
 9632 PRINT 2222, PLATLAB(I),CODE(I)
      IF(FIXCODE.EQ.1) GO TO 8501
      HF1(1) = THIKK1(1,1,K)
      HF1(2) = THIKK1(2,1,K)
      IFL = ITR + 2
      IIL = ITF + 4
      HWF(1) = THIKK1(IFL,1,K)
      HWF(2) = THIKK1(IIL,1,K).
      FIXCODE = 1.0
```

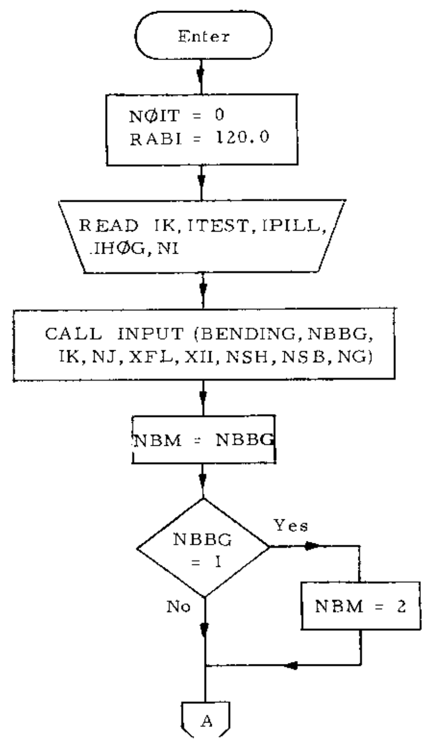
```
      DO 8502 I = 1, IPLATE
 8502 CODE(I) = -1,0
      GO TO 348
 8501 HS1(1) = THIKK1(1,1,K)
      HS1(2) = THIKK1(2,1,K)
      HWS(1) = THIKK1(IFL,1,K)
      HWS(2) = THIKK1(IIL,1,K)
      HWFS(L) = 0.5(HWF(L) + HWS(L))
      CALL INTERMED
 9999 CALL DOCUMENT
      CALL COSTING
 404  FORMAT (2F15,2)
 326  FORMAT( * KKM=*,I3/)
 9139 FORMAT (1H0,10X,13H EIGEN, EQU, // 5X, 7H REAL  9E13,5)
 9144 FORMAT (1H0,10X, 6H ROUT(I2,4H)=   E13,5)
      END
```

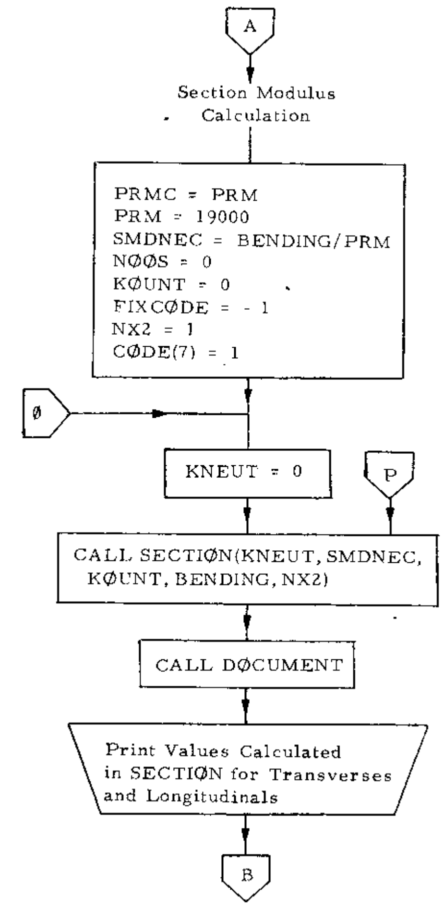
Macro Flow Diagram - No Loops Shown

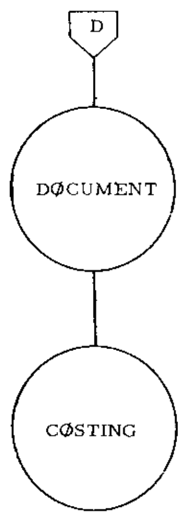


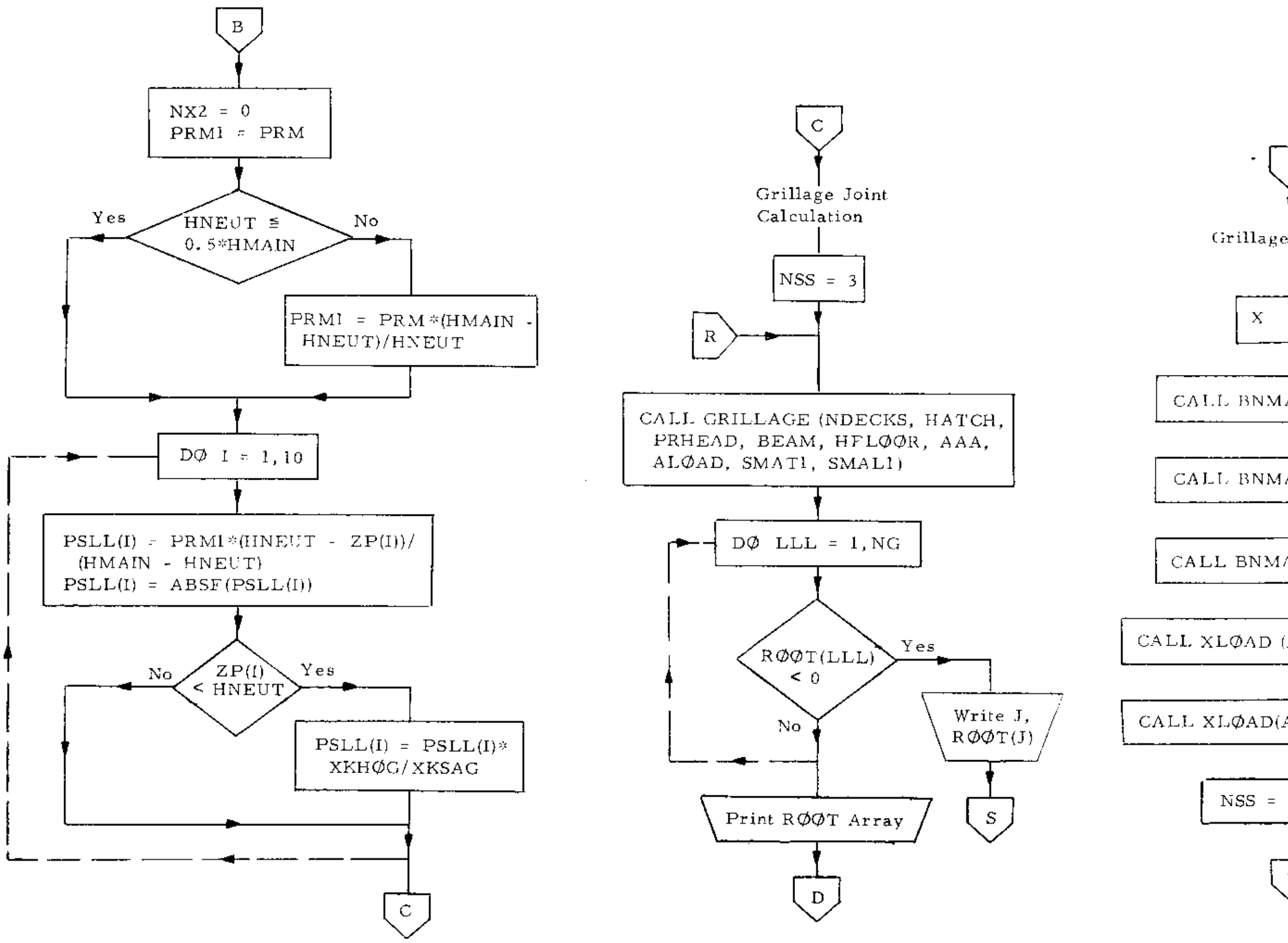

A
GRILLAGE
FRAME - MATINV
EV
R TPLSUB
BNMAT
EV
T - THETA
XLØAD
EV
T - THETA
B
SØLVE
BNMAT
MATINV
EV
T - THETA
STEP
BNMAT
XLØAD
EV
T - THETA
EV
T - THETA
REACT
MATINV
C
FRAME
MATINV
STRESS
DØCUMENT
INTERMED
D

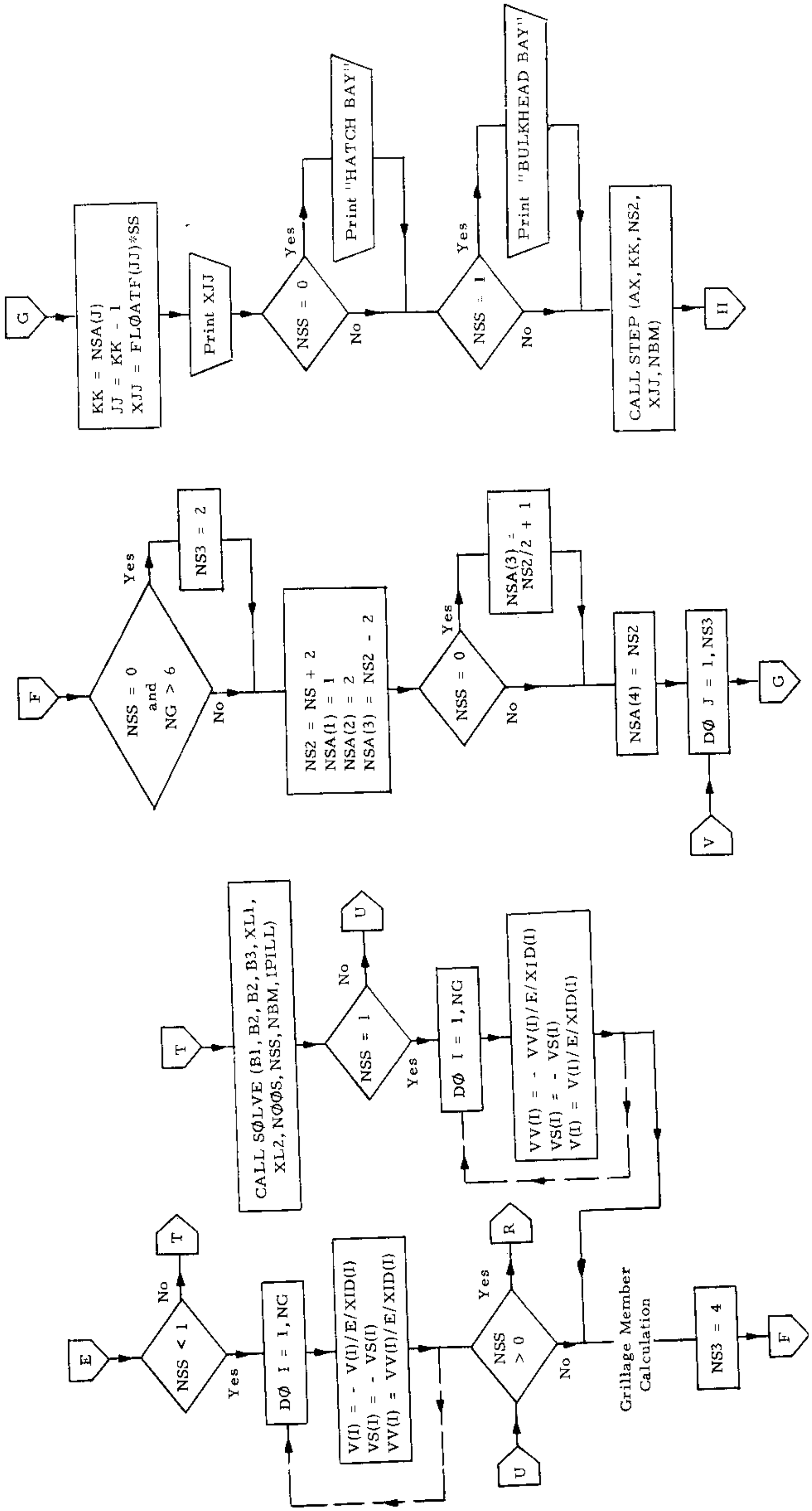
Program TRANSHIP (detailed flow chart)

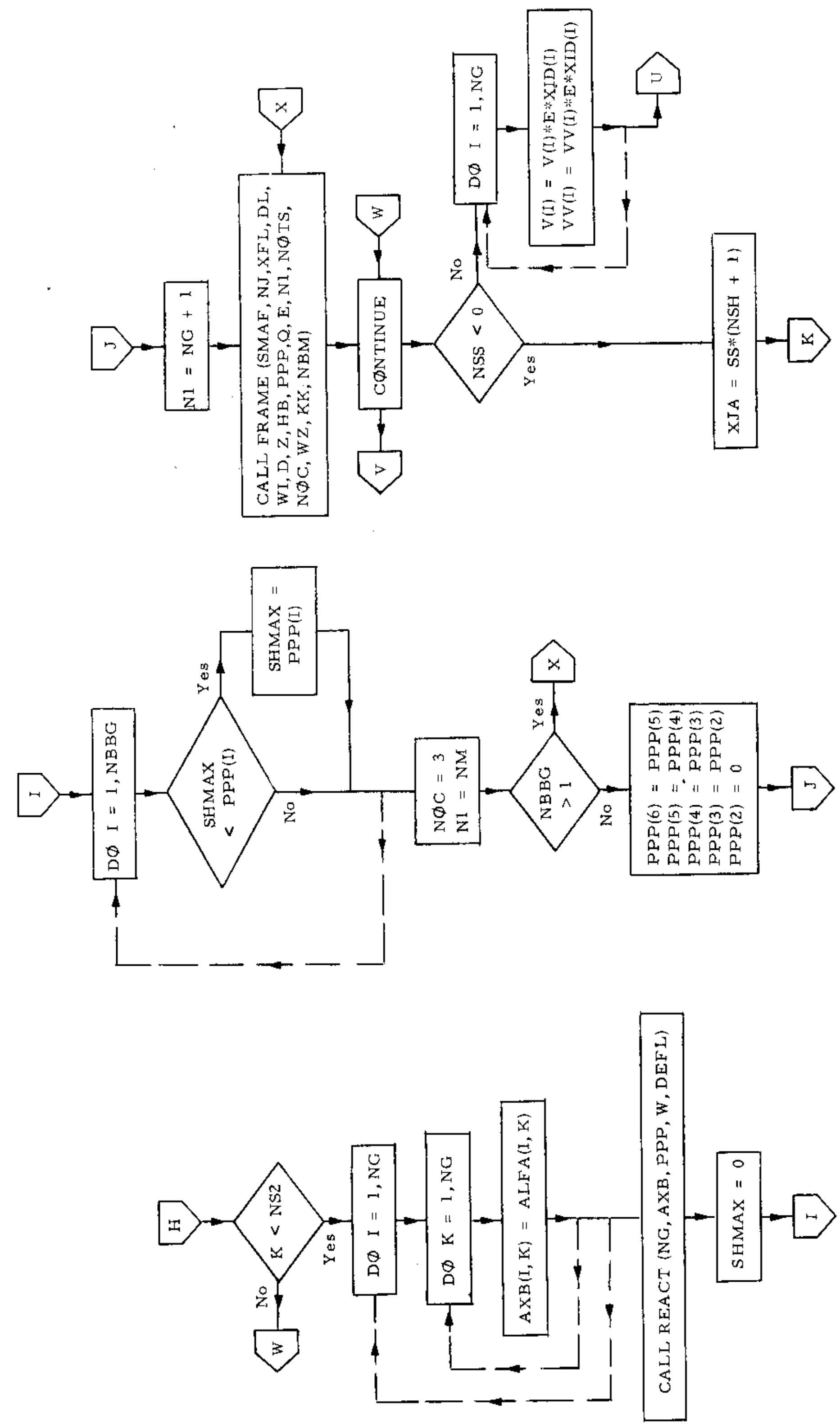


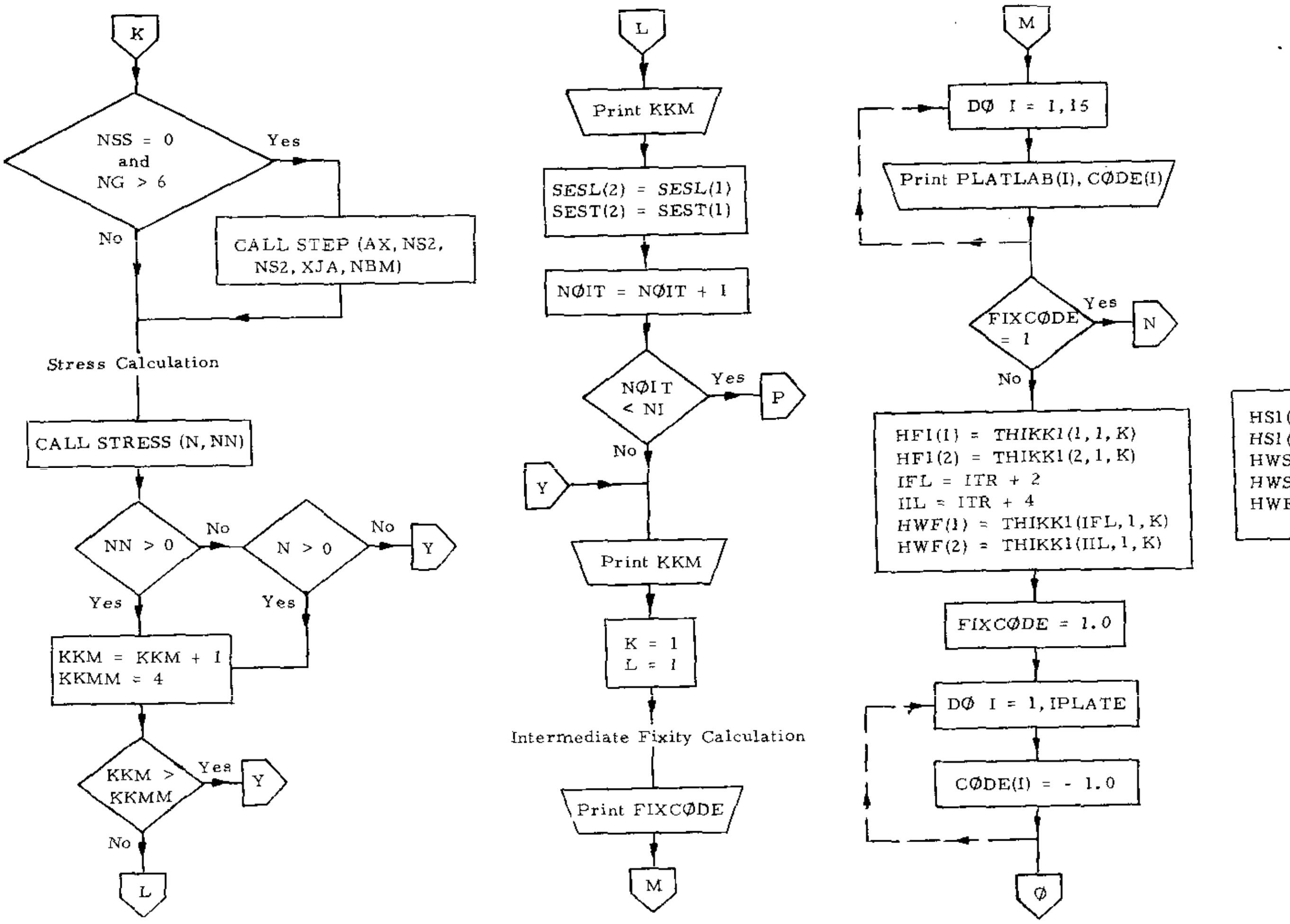





B
NX2 = 0
PRM1 = PRM
HNEUT ≦ 0.5*HMAIN
Yes
No
PRM1 = PRM*(HMAIN - HNEUT)/HNEUT
DØ I = 1,10
PSLL(I) = PRM1*(HNEUT - ZP(I))/(HMAIN - HNEUT)
PSLL(I) = ABSF(PSLL(I))
ZP(I) < HNEUT
No
Yes
PSLL(I) = PSLL(I)*XKHØG/XKSAG
C
C
Grillage Joint Calculation
NSS = 3
R
CALL GRILLAGE (NDECKS, HATCH, PRHEAD, BEAM, HFLØØR, AAA, ALØAD, SMAT1, SMAL1)
DØ LLL = 1,NG
RØØT(LLL) < 0
Yes
No
Write J, RØØT(J)
S
Print RØØT Array
D
Grillage
X
CALL BNM
CALL BNM
CALL BNM
CALL XLØAD (
CALL XLØAD(
NSS =


E
NSS < 1
No
T
Yes
DØ I = 1, NG
V(I) = - V(I)/E/XID(I)
VS(I) = - VS(I)
VV(I) = VV(I)/E/XID(I)
NSS > 0
Yes
R
No
U
Grillage Member Calculation
NS3 = 4
F
T
CALL SØLVE (B1, B2, B2, B3, XL1, XL2, NØØS, NSS, NBM, IPILL)
NSS = 1
No
U
Yes
DØ I = 1, NG
VV(I) = - VV(I)/E/XID(I)
VS(I) = - VS(I)
V(I) = V(I)/E/XID(I)
F
NSS = 0 and NG > 6
Yes
NS3 = 2
No
NS2 = NS + 2
NSA(1) = 1
NSA(2) = 2
NSA(3) = NS2 - 2
NSS = 0
Yes
NSA(3) = NS2/2 + 1
No
NSA(4) = NS2
DØ J = 1, NS3
G
V
G
KK = NSA(J)
JJ = KK - 1
XJJ = FLØATF(JJ)*SS
Print XJJ
NSS = 0
Yes
Print "HATCH BAY"
No
NSS = 1
Yes
Print "BULKHEAD BAY"
No
CALL STEP (AX, KK, NS2, XJJ, NBM)
H


H
K < NS2
No
Yes
W
DØ I = 1, NG
DØ K = 1, NG
AXB(I, K) = ALFA(I, K)
CALL REACT (NG, AXB, PPP, W, DEFL)
SHMAX = 0
I
I
DØ I = 1, NBBG
SHMAX < PPP(I)
Yes
No
SHMAX = PPP(I)
NØC = 3
N1 = NM
NBBG > 1
Yes
No
X
PPP(6) = PPP(5)
PPP(5) = PPP(4)
PPP(4) = PPP(3)
PPP(3) = PPP(2)
PPP(2) = 0
J
J
N1 = NG + 1
X
CALL FRAME (SMAF, NJ, XFL, DL, WI, D, Z, HB, PPP, Q, E, N1, NØTS, NØC, WZ, KK, NBM)
W
CONTINUE
V
NSS < 0
No
Yes
DØ I = 1, NG
V(I) = V(I)*E*XID(I)
VV(I) = VV(I)*E*XID(I)
U
XJA = SS*(NSH + 1)
K


K
NSS = 0 and NG > 6
Yes
No
CALL STEP (AX, NS2, NS2, XJA, NBM)
Stress Calculation
CALL STRESS (N, NN)
NN > 0
No
N > 0
No
Y
Yes
Yes
KKM = KKM + 1
KKMM = 4
KKM > KKMM
Yes
Y
No
L
L
Print KKM
SESL(2) = SESL(1)
SEST(2) = SEST(1)
NØIT = NØIT + 1
NØIT < NI
Yes
P
No
Y
Print KKM
K = 1
L = 1
Intermediate Fixity Calculation
Print FIXCØDE
M
M
DØ I = 1, 15
Print PLATLAB(I), CØDE(I)
FIXCØDE = 1
Yes
N
No
HF1(1) = THIKK1(1, 1, K)
HF1(2) = THIKK1(2, 1, K)
IFL = ITR + 2
IIL = ITR + 4
HWF(1) = THIKK1(IFL, 1, K)
HWF(2) = THIKK1(IIL, 1, K)
FIXCØDE = 1.0
DØ I = 1, IPLATE
CØDE(I) = - 1.0
Ø

1. Subroutine ASPECT

a) Abstract:

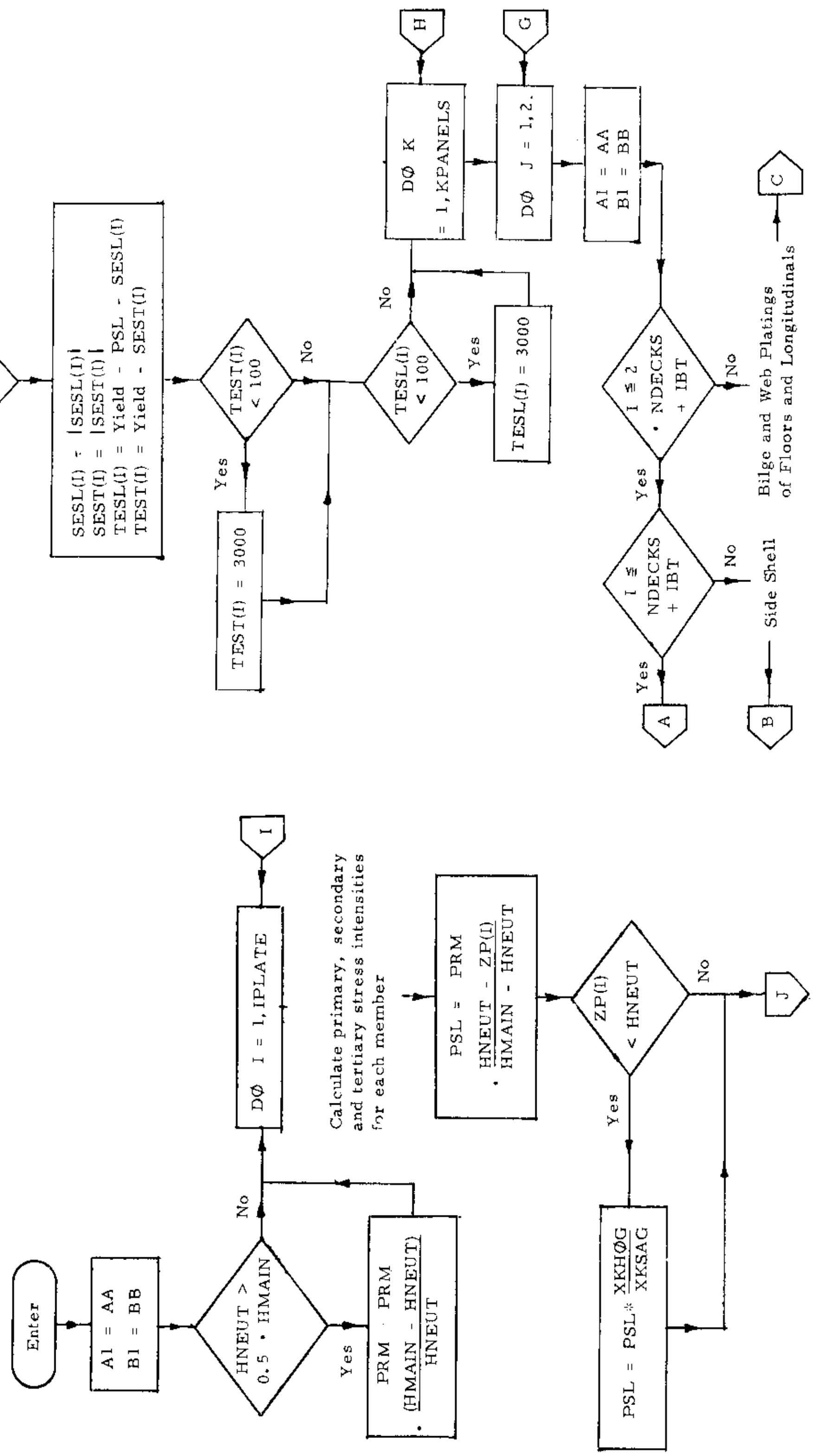
Subroutine ASPECT is called from subroutine SECTION. It calculates the aspect ratio of all plating and calls subroutine PLATING or NTPLATE as appropriate.

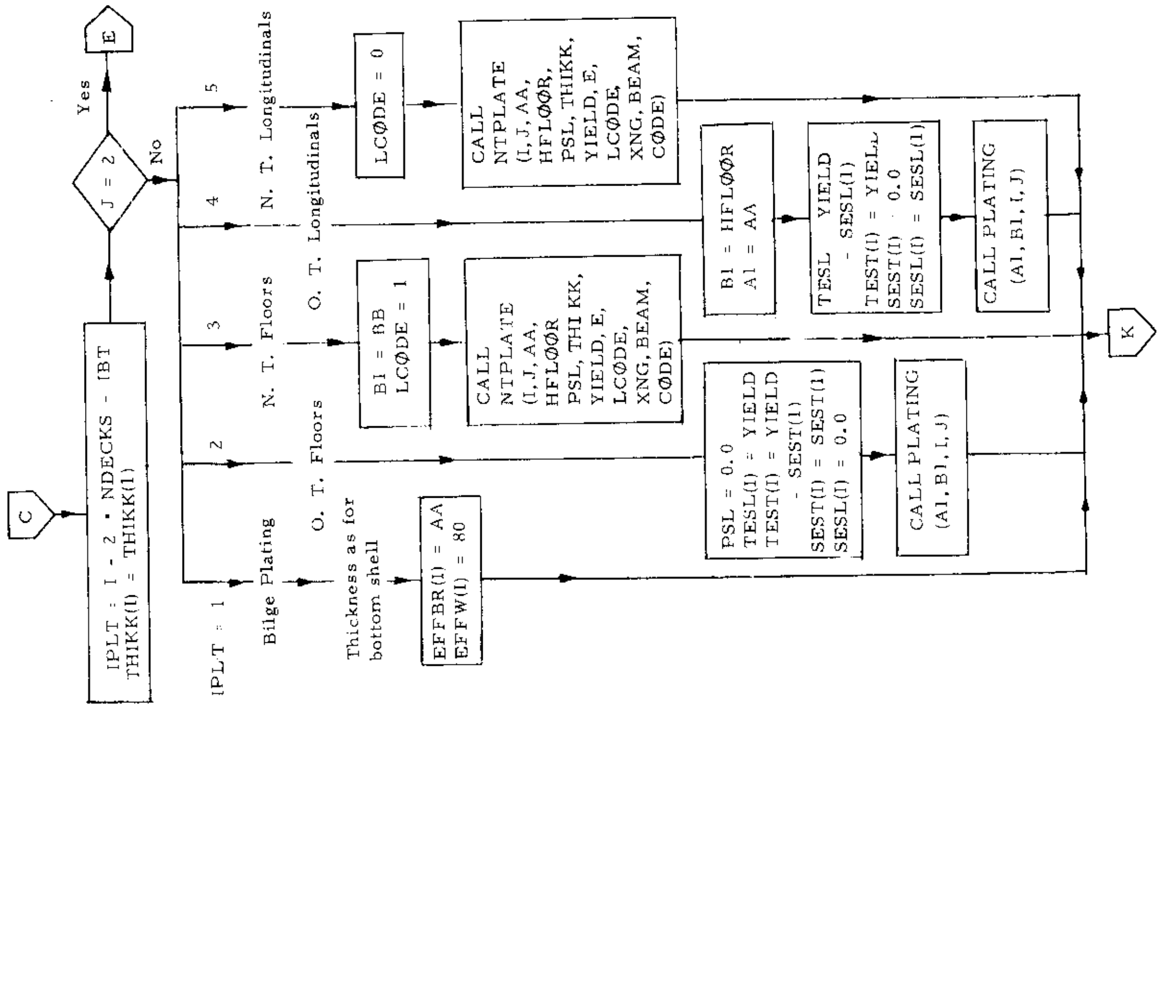
b) Terms specific to this subroutine:

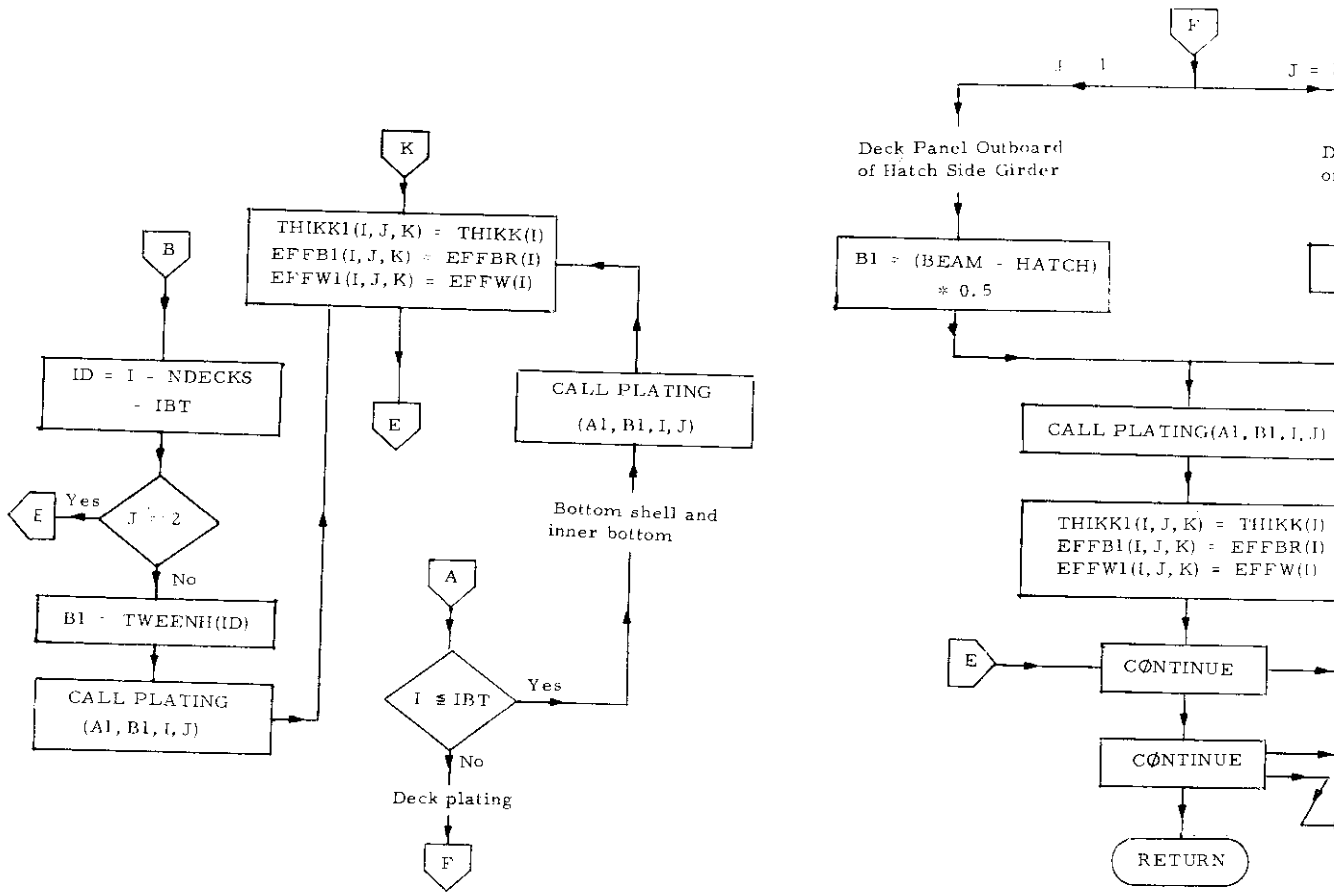
FORTRAN Term	Definition
A1	Side of plate rectangle used to determine aspect ratio
B1	Length of plate rectangle used to determine aspect ratio
LCØDE	Code = 1 denotes non-tight floors = 0 denotes non-tight longitudinals

```
      SUBROUTINE ASPECT
      COMMON/TLEN/  TLENGTH   ,IHOG ,RABI,XKSAG,XKHOG
      COMMON/DE/ DEF(20)
      COMMON /E/ IBT,E
      COMMON /B/ A,B,NDECKS,IPLATE,PSL,PRM,HMAIN,HNEUT,TESL,TEST,SESL,
     1    SEST,         KPANELS,BEAM,XNG,HATCH,THIKK,THIKK1,EFFBR,EFFB1,
     2    EFFW,EFFW1,HFLOOR,TWEENH,THIKX,EFBR,WEB,WEB1,XIPFX,CHT,
     3    XIS, AREAT1,SMAL1,SMODL1,WEBL1,SUMAREAP,SUMAREAL,SUMMP,SUMML,
     4    AREA,SUMAREA,SUMMOM,WPL,XPANEL,WBAYLP,XNST,NST,ITR,
     5    WTR,BAYTR,SUMWT,SUMWTR,LLGX,   YIELD,WBAYTR,XLABOR,PLCOST,
     6    COST,COSTMIN,GNOT,GNNT,KNEUT,SUMSMAP,SUMSMAL,SUMSMA,CODE,
     7WAVEH,PRHEAD,DRAFT,ALOAD,XLHOLD,XLPANEL,AAA,BBB,AA,BB,DELTESL,
     8     SUMSMA1,ZP,ZL,ILG,AREAL1,ASX,   SMAT1,SMODT1
      DIMENSION ZP(17),SESL(17),SEST(17),TESL(17),TEST(17),XNG(7),THIKK(
     117),      THIKK1(17,2,5),EFFBR(17),EFFW(17),EFFB1(17,2,5),EFFW1(17
     2,2,5),AREA(17,5),TWEENH(5),SMAT1(12,2,5),SMODT1(12,2,5),THIKX(12,2
     3),EFBR(12,2),AREAT1(12,2,5),WEB1(12,2,5),SMAL1(17,2,3),SMODL1(17,2
     4,3),WEBL1(7,2,3),AREAL1(7,2,3),SUMAREAP(5),SUMAREAL(5),SUMMP(5),SU
     5MML(5),SUMAREA(5),SUMMOM(5),WPL(5),XLPANEL(5),SUMWTR(5),WTR(12),ZL
     6(7),TIKK1(17,5),CODE(17),ALOAD(17),SUMSMAP(5),SUMSMAL(5),SUMSMA(5)
      B1=BB
      A1=AA
      IF(HNEUT.LE. 0.5 * HMAIN) GO TO 11
      PRM =PRM * (HMAIN-HNEUT) /HNEUT
   11 DO 111 I=1,IPLATE
      PSL =PRM *(HNEUT - ZP(I))/(HMAIN -HNEUT)
      PSL=ABSF(PSL)
      IF( ZP(I) .LT. HNEUT ) PSL=PSL*XKHOG/XKSAG
      SESL(I)=ABSF(SESL(I))
      SEST(I)=ABSF(SEST(I))
      TESL(I) = YIELD - PSL - SESL(I)
      TEST(I) = YIELD -SEST(I)
       IF(TEST(I).LT. 100.) TEST(I)= 3000.
       IF(TESL(I).LT. 100.) TESL(I)= 3000.
      DO 111  K = 1,KPANELS
      DO 9111 J=1,2
      A1=AA
      B1=BB
      IF(I-2*NDECKS-IBT)90,90,59
   90 IF(I-NDECKS-IBT)58,58,112
   58 IF(I-IBT)51,51,50
   51 CONTINUE
      GO TO 919
C     DECKS
   50 CONTINUE
      GO TO (55,56)J
   55 B1= (BEAM-HATCH)*0.5
      GO TO 10
   56 B1=HATCH
   10 CONTINUE
      GO TO 919
C     WEBS
C     SIDE SHELL
  112 ID=I-NDECKS-IBT
      IF(J.EQ.2)  GO TO 9111
      B1= TWEENH(ID)
      GO TO 919
   59 IPLT=I-2*NDECKS-IBT
      THIKK(I)=THIKK(1)
      IF(J.EQ.2)  GO TO 9111
      GO TO (67,68,69,75,118)IPLT
C     BILGE
   67 CONTINUE
C     CALL BILGE
      EFFBR(I) = AA
      EFFW(I) = 80.
      GO TO 119
C     O.T.FLOORS
   68 A1=HFLOOR
      B1=BB
       PSL = 0.0
      TESL(I) = YIELD
      TEST(I) = YIELD - SEST(1)
      SEST(I) = SEST (1)
      SESL(I) = 0.0
      GO TO 919
C     N.T.FLOORS
   69 B1=BB
C  NTFLOOR
      LCODE=1
      CALL  NTPLATE(I,J,AA,HFLOOR,PSL,THIKK,YIELD,E,LCODE,XNG,BEAM,CODE)
      GO TO 119
C     O.T.LONGITUDINALS
   75 B1=HFLOOR
      A1=AA
      TESL(I) = YIELD-SESL(1)
      TEST(I) = YIELD
      SEST(I) = 0.0
      SESL(I) = SESL(1)
      GO TO 919
C     N.T.LONGITUDINALS
C  NTLONG
  118 LCODE=0
      CALL  NTPLATE(I,J,AA,HFLOOR,PSL,THIKK,YIELD,E,LCODE,XNG,BEAM,CODE)
      GO TO 119
  919 CALL PLATING   (A1,B1,I,J)
  119 THIKK1(I,J,K) =THIKK(I)
      EFFB1(I,J,K) = EFFBR(I)
      EFFW1(I,J,K) = EFFW(I)
 9111 CONTINUE
  111 CONTINUE
      RETURN
      END
```

Subroutine ASPECT




C
IPLT : I - 2 · NDECKS - IBT
THIKK(I) = THIKK(I)
J = 2
Yes
E
No
IPLT = 1
2
3
4
5
Bilge Plating
O. T. Floors
N. T. Floors
O. T. Longitudinals
N. T. Longitudinals
Thickness as for bottom shell
EFFBR(I) = AA
EFFW(I) = 80
B1 = BB
LCØDE = 1
LCØDE = 0
CALL NTPLATE (I, J, AA, HFLØØR, PSL, THIKK, YIELD, E, LCØDE, XNG, BEAM, CØDE)
CALL NTPLATE (I, J, AA, HFLØØR, PSL, THIKK, YIELD, E, LCØDE, XNG, BEAM, CØDE)
PSL = 0.0
TESL(I) = YIELD
TEST(I) = YIELD - SEST(I)
SEST(I) = SEST(I)
SESL(I) = 0.0
CALL PLATING (A1, B1, I, J)
B1 = HFLØØR
A1 = AA
TESL = YIELD - SESL(I)
TEST(I) = YIELD
SEST(I) = 0.0
SESL(I) = SESL(I)
CALL PLATING (A1, B1, I, J)
K



B
ID = I - NDECKS - IBT
J ≥ 2
Yes
E
No
B1 = TWEENH(ID)
CALL PLATING (A1, B1, I, J)
K
THIKK1(I, J, K) = THIKK(I)
EFFB1(I, J, K) = EFFBR(I)
EFFW1(I, J, K) = EFFW(I)
E
CALL PLATING (A1, B1, I, J)
Bottom shell and inner bottom
A
I ≤ IBT
Yes
No
Deck plating
F
F
J = 1
J =
Deck Panel Outboard of Hatch Side Girder
B1 = (BEAM - HATCH) * 0.5
CALL PLATING(A1, B1, I, J)
THIKK1(I, J, K) = THIKK(I)
EFFB1(I, J, K) = EFFBR(I)
EFFW1(I, J, K) = EFFW(I)
E
CØNTINUE
CØNTINUE
RETURN

2. Subroutine BNMAT (A, B, M)

a) Abstract:

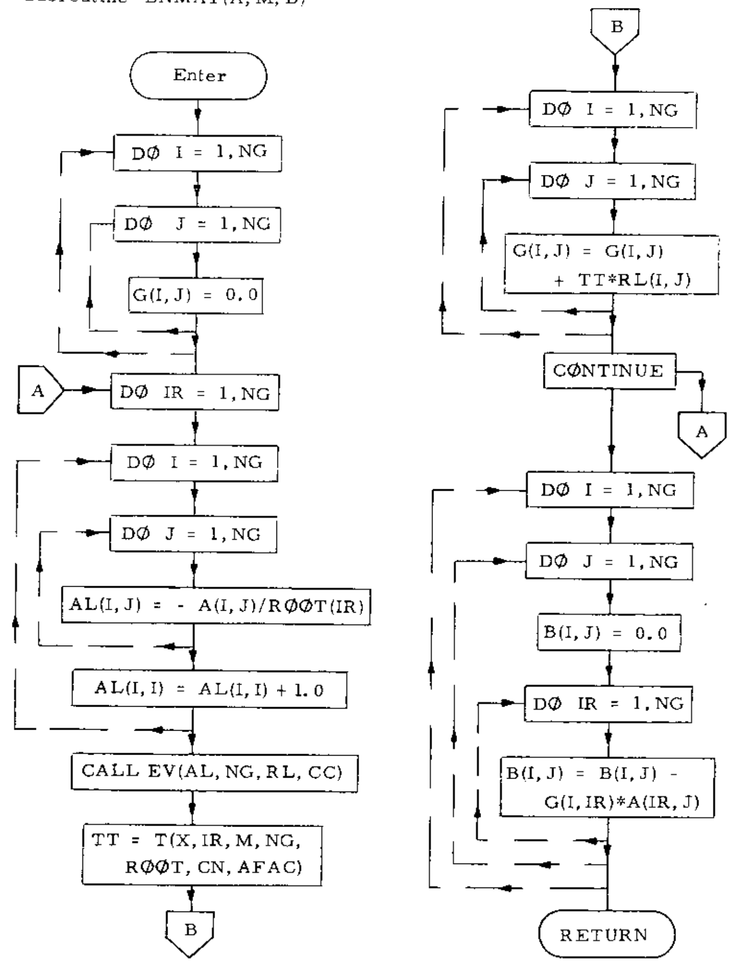
Subroutine BNMAT calculates the boundary matrices used in the grillage calculation called from subroutine GRILLAGE, SOLVE and STEP.

b) Terms specific to this subroutine:

FORTRAN Term	Definition
A	Grillage matrix
AL	Characteristic matrix
M	Index of Nielsen function

```
      SUBROUTINE BNMAT(A,M,B)
      COMMON /E/ IBT,E
      COMMON   ALFA, AX, CC, CN, DEFL,    GP, H, JJ, K, NG, NS, PD,
     1ROOT, S, SS, V, VV, W, X, XID, XII, XL, XLL, XML, XMO, XX,RR,SCAL
     2 ,AFAC,NNN
      DIMENSION  ALFA(15,15),AX( 9,9 ),CC(10),DEFL( 9),PD(15,40),
     1ROOT( 9),S(15),V(15),VV(15),W(15),XID(15),XML(15),XMO(15),XX(40),
     2XII(40),A(9,9),B(9,9),AL(9,9),RL(9,9),G(9,9)
   98 FORMAT (I3)
   91 FORMAT (F15,5)
      DO 1000 I=1,NG
      DO 1000 J=1,NG
      G(I,J)=0,0
 1000 CONTINUE
      DO 5000 IR=1,NG
      DO 4000 I=1,NG
      DO 3000 J=1,NG
      AL(I,J)=-A(I,J)/ROOT(IR)
 3000 CONTINUE
      AL(I,I)=AL(I,I)+1,0
 4000 CONTINUE
       CALL EV(AL,NG,RL,CC)
      TT=T(X,IR,M,NG,ROOT,CN,AFAC)
      DO 4600 I=1,NG
      DO 4500 J=1,NG
      G(I,J)=G(I,J)+TT*RL(I,J)
 4500 CONTINUE
 4600 CONTINUE
 5000 CONTINUE
      DO 6000 I=1,NG
      DO 6000 J=1,NG
      B(I,J)=0,0
      DO 6000 IR=1,NG
      B(I,J)=B(I,J)-G(I,IR)*A(IR,J)
 6000 CONTINUE
      RETURN
      END
```

Subroutine BNMAT(A, M, B)



3. Subroutine COSTING

a) Abstract:

Subroutine COSTING is called from program TRANSHIP. It calculates the weight and cost of all transverse and longitudinal members, as well as all welding and labor costs. The overall weight and cost of hull structure per inch of length are then determined.

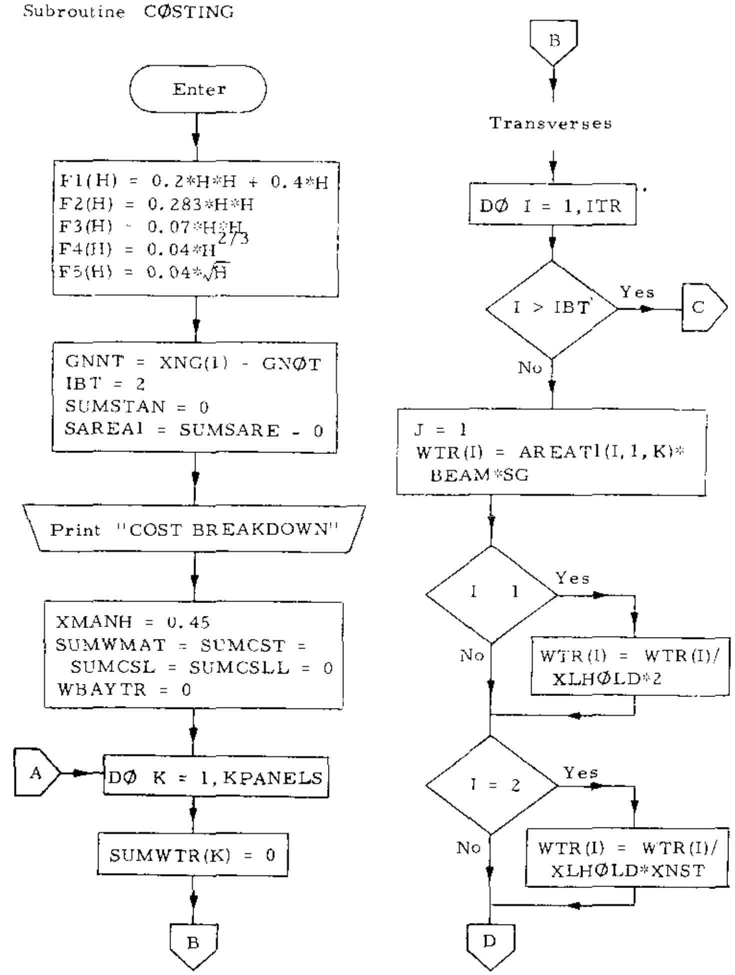
b) Terms specific to this subroutine:

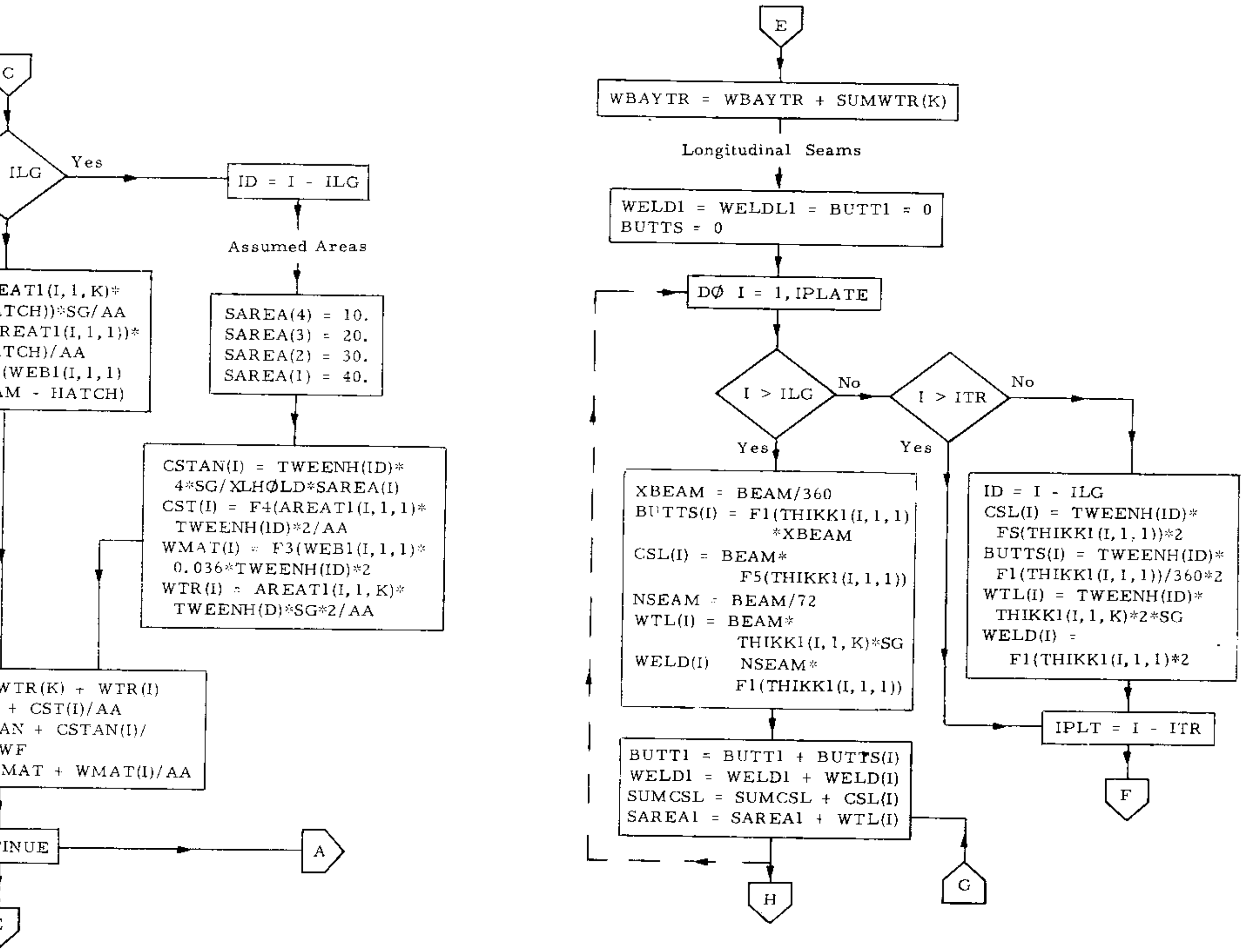
FORTRAN Term	Definition	Mathematical Symbol
COSTS	Combined cost of plating, labor and welding per inch of ship length	C
F1(H)	Weight per inch of vee weld	$f_1(h)$
F2(H)	Weight per inch of continuous double fillet weld	$f_2(h)$
F3(H)	Weight per inch of intermittent double fillet weld	$f_3(h)$
F4(H)	Time - size factor	$f(A_i)$
NSEAM	Number of weld seams in inner bottom	N_s
PLCOST	Total material cost	C_m
TWELD	Total cost of welding	C_w
WBAYTOT	Total weight of hull structure per inch	S
XLABOR	Total labor cost of erection and fabrication	C_{fe}

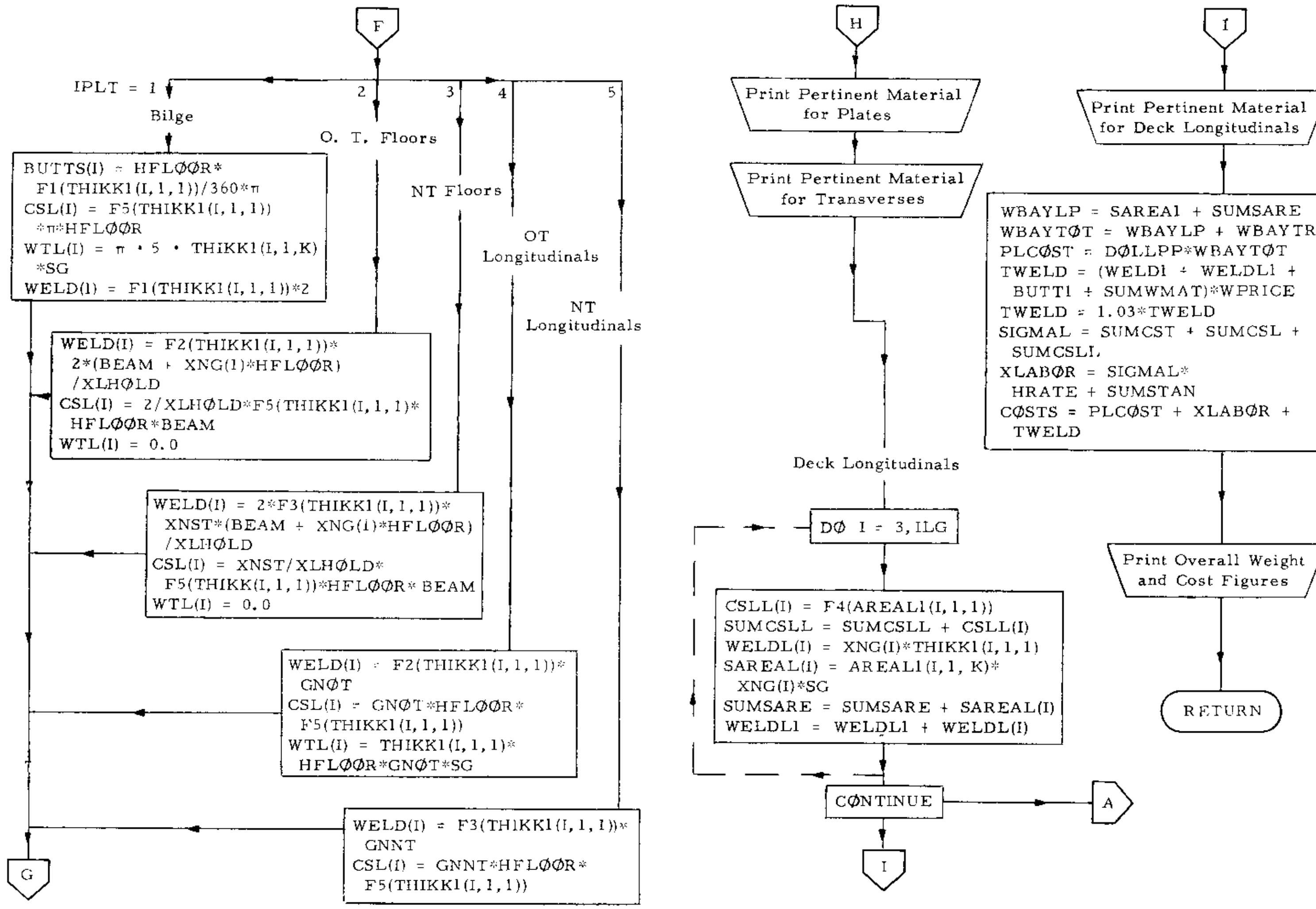
```
      SUBROUTINE COSTING
      COMMON/LABEL/ TRANLAB(10),PLATLAB(15),LONGLAB(6)
      DIMENSION CSTAN(10) ,SAREA(10) ,SAREAL(10) ,WTL(17)
      COMMON/COST/ HRATE,DOLLPP,WPRICE,TWF   ,SG
      COMMON /B/ A,B,NDECKS,IPLATE,PSL,PRM,HMAIN,HNEUT,TESL,TEST,SESL,
     1     SEST,         KPANELS,BEAM,XNG,HATCH,THIKK,THIKK1,EFFBR,EFFB1,
     2     EFFW,EFFW1,HFLOOR,TWEENH,THIKX,EFOR,WEB,WEB1,XIPFX,CHI,
     3     XIS, AREAT1,SMAL1,SMODL1,WEBL1,SUMAREAP,SUMAREAL,SUMMP,SUMML,
     4     AREA,SUMAREA,SUMMOM,WPL,XPANEL,WBAYLP,XNST,NST,ITR,
     5     WTR,BAYTR,SUMWT,SUMWTR,LLGX,   YIELD,WBAYTR,XLABOR,PLCOST,
     6     COST,COSTMIN,GNOT,GNNT,KNEUT,SUMSMAP,SUMSMAL,SUMSMA,CODE,
     7WAVEH,PRHEAD,DRAFT,ALOAD,XLHOLD,XLPANEL,AAA,BBB,AA,BB,DELTESL,
     8     SUMSMA1,ZP,ZL,ILG,AREAL1,ASX,   SMAT1,SMODT1
      COMMON /E/ IBT,E
      DIMENSION ZP(17),SESL(17),SEST(17),TESL(17),TEST(17),XNG(7),THIKK(
     117),        THIKK1(17,2,5),EFFBR(17),EFFW(17),EFFB1(17,2,5),EFFW1(17
     2,2,5),AREA(17,5),TWEENH(5),SMAT1(12,2,5),SMODT1(12,2,5),THIKX(12,2
     3),EFBR(12,2),AREAT1(12,2,5),WEB1(12,2,5),SMAL1(17,2,3),SMODL1(17,2
     4,3),WEBL1(7,2,3),AREAL1(7,2,3),SUMAREAP(5),SUMAREAL(5),SUMMP(5),SU
     5MML(5),SUMAREA(5),SUMMOM(5),WPL(5),XLPANEL(5),SUMWTR(5),WTR(12),ZL
     6(7),TIKK1(17,5),CODE(17),ALOAD(17),SUMSMAP(5),SUMSMAL(5),SUMSMA(5)
      DIMENSION WELD(17),BUTTS(17),WELDL(7)
      DIMENSION CSL(17),CST(17),WMAT(17),CSLL(17)
      F1(H)= (0.2*H*H+.04*H)
      F2(H)=  .283*H*H
        F3(H)= 0.07*H*H
      F4(H)=0.04*CUBERTF(H*H)
       F5(H)=0.04*SQRTF(H)
      GNNT=XNG(1)-GNOT
      IBT=2
      SUMSTAN=0.
      SAREA1=SUMSAREF  =0.
      PRINT 9985
 9985 FORMAT(1H1,*  COST BREAKDOWN *,/ ,*  PLATES *,/)
      XMANH = 0.45
      SUMWMAT=SUMCST=SUMCSL=SUMCSLL=0.
      WBAYTR =0.0
      DO 1  K =1,KPANELS
      SUMWTR(K)=0.0
C   TRANSVERSES
      DO 2  I = 1,ITR
      IF(I.GT.IBT) GO TO 5
    4 J=1
      WTR(I)  = AREAT1(I,1,K)*BEAM*SG
      IF(I.EQ.1)  WTR(I)=WTR(I)/XLHOLD*2
      IF(I.EQ.2)  WTR(I)=WTR(I)/XLHOLD*    XNST
      GO TO 6
    5 IF(I.GT.ILG) GO TO 12
   11 WTR(I)=(AREAT1(I,1,K)*(BEAM -HATCH))*SG/AA
      CST(I)=F4(AREAT1 (I,1,1))*(BEAM-HATCH) /AA
      WMAT(I)=F3(WEB1(I,1,1)*0.036)*(BEAM-HATCH)
      GO TO 6
   12 ID = I - ILG
C     ASSUMED AREAS
      SAREA(4)=10.
      SAREA(3)=20.
      SAREA(2)=30.
      SAREA(1)=40.
      CSTAN(I)=TWEENH (ID)*4.* SG /XLHOLD*SAREA(I)
      CST(I)=F4(AREAT1(I,1,1))*TWEENH(ID)*2. /AA
      WMAT(I)=F3(WEB1(I,1,1)*0.036)*TWEENH(ID)*2.
      WTR(I) = AREAT1(I,1,K) * TWEENH(ID) * SG    * 2.0    /AA
    6 SUMWTR(K) = SUMWTR(K) + WTR(I)
       SUMCST=SUMCST+CST(I)/AA
      SUMSTAN=SUMSTAN+CSTAN(I)/AA* TWF
        SUMWMAT=SUMWMAT+WMAT(I)/AA
    2 CONTINUE
      WBAYTR = WBAYTR + SUMWTR(K)
C     LONGITUDINAL SEAMS
      WELD1 = WELDL1 = BUTT1 = 0.0
      BUTTS=0.
      DO 21 I = 1,IPLATE
      IF(I.GT.ILG) GO TO 22
      XBEAM=BEAM/360.
      BUTTS(I)=F1(THIKK1(I,1,1)  )*XBEAM
       CSL(I)=BEAM*F5(THIKK1(I,1,1))
      NSEAM=BEAM/72.
        WTL(I)= BEAM*THIKK1(I,1,K)*SG
   23 WELD(I) =  NSEAM* F1(  THIKK1(I,1,1))
      GO TO 25
   22 IF(I.GT.ITR) GO TO 24
      ID = I - ILG
       CSL(I)=TWEENH(ID)*F5(THIKK1(I,1,1))    *2
      BUTTS(I)=TWEENH(ID)*F1(THIKK1(I,1,1))/360    *2.
         WTL(I)=TWEENH(ID)* THIKK1(I,1,K) *2.0  *SG
            WELD(I)=F1(THIKK1(I,1,1))           *2.
   24 IPLT = I-ITR
      GO TO (26,27,28,29,30) IPLT
C     BILGE
   26 BUTTS(I)=HFLOOR  *  F1(THIKK1(I,1,1))/360    *3.1416
       CSL(I)=            F5(THIKK1(I,1,1))      *3.1416*HFLOOR
         WTL(I) =3.1416* 5.0*THIKK1(I,1,K)  *SG
            WELD(I)=F1(THIKK1(I,1,1))    *2.
      GO TO 25
C     DT FLOORS
   27 WELD(I) =      F2(THIKK1(I,1,1) ) *2. *(BEAM+XNG(1)*HFLOOR)/XLHOLD
      CSL(I)=2.  /XLHOLD*F5(THIKK1(I,1,1))*HFLOOR*BEAM
        WTL(I)=0.0
      GO TO 25
C       NT FLOORS
   28 WELD(I)=2.0*  F3(THIKK1(I,1,1)  )*XNST*(BEAM+XNG(1)*HFLOOR)/XLHOLD
      CSL(I)=XNST/XLHOLD*F5(THIKK1(I,1,1))*HFLOOR*BEAM
        WTL(I)=0.0
      GO TO 25
C     DT LONG
   29 WELD(I) = F2(THIKK1(I,1,1) )    *GNOT
      CSL(I)=  GNOT*  HFLOOR*F5(THIKK1(I,1,1)    )
         WTL(I)  = THIKK1(I,1,1)* HFLOOR * GNOT*SG
        WTL(I)   = THIKK1(I,1,1)* HFLOOR * GNNT*SG
      GO TO 25
C     NT LONG
   30 WELD(I) =F3(THIKK1(I,1,1) )    *GNNT
```

```
      CSL(I)=  GNNT*  WELDOR*F5(THIKK1(I,1,1)     )
 25   BUTT1= BUTT1 + BUTTS(I)
      WELD1 = WELD1 + WELD(I)
      SUMCSL=SUMCSL+CSL(I)
      SAREA1=SAREA1+WTL(I)
 21   CONTINUE
      PRINT 999
 999 FORMAT(/, *                          SEAM WELD     BUTT WELD     MANHOURS      LB
     1/INCH *,/)
      DO 8799 I=1,15
 8799 PRINT 933, PLATLAB(I),WELD(I), BUTTS(I) ,CSL(I),WTL(I)
      PRINT 986       ,WELD1,BUTT1,SUMCSL ,SAREA1
      PRINT 93
 93   FORMAT(//,*  TRANSVERSES *,//,*                          LB/IN       WELD MATER
     1IAL  MANHOURS *,/)
      DO 5555 I=1,10
 5555 PRINT 933, TRANLAB(I), WTR(I),WMAT(I),CST(I)
      PRINT 986,SUMWTR(1),SUMWMAT,SUMCST
C     DECK LONGITUDINALS
      DO 31  I = 3, ILG
      CSLL(I)=F4(AREAL1(I,1,1))
      SUMCSLL=SUMCSLL+CSLL(I)
      WELDL(I) = XNG(I) * THIKK1(I,1,1)
      SAREAL(I)=     AREAL1(I,1,K)*XNG(I) *S,
      SUMSARE  =SUMSARE  +SAREAL(I)
       WELDL1 = WELDL1 + WELDL(I)
 31    CONTINUE
   1 CONTINUE
       PRINT 993
 993  FORMAT(/,* DECK LONGITUDINALS *,//,*                          WELD          MAN
     1HOURS         LB/IN*,/)
      DO 33,I=3,ILG
 33   PRINT 933, LONGLAB(I),WELDL(I),CSLL(I) ,SAREAL(I)
      PRINT 986 ,WELDL1,SUMCSLL    ,SUMSARE
      WBAYLP =      SAREA1+SUMSARE
      WBAYTOT = WBAYLP + WBAYTR
      PLCOST = DOLLPP * WBAYTOT
      TWELD = (WELD1+WELDL1 + BUTT1+SUMWMAT)*WPRICE
      TWELD=1,03*TWELD
      SIGMAL=SUMCST+SUMCSL+SUMCSLL
       XLABOR=SIGMAL*HRATE +SUMSTAN
      COSTS= PLCOST + XLABOR + TWELD
      PRINT 20,WBAYLP,WBAYTR,SIGMAL
      PRINT 9851 ,HRATE,DOLLPP,WPRICE,SG
 9851 FORMAT( * LABOR RATE =*,F10,3,* DOLLARS PER HOUR  */
     1          * PLATE COST =*,F10,3,* DOLLARS PER POUND */
     2          * WELD  COST =*,F10,3,* DOLLARS PER POUND */
     3          * DENSITY    =*,F10,3,* POUNDS PER CUBIC INCH*/)
 20   FORMAT(//,*    WT, LONG, =*,F10,2/,*    WT, TRANS =*,F10,2/,
     1*    SUM OF MH =*,F10,2//)
 933  FORMAT(*  *,A8 ,5E13,3 )
 986  FORMAT(*  TOTAL   *,5E13,3)
      PRINT 413, COSTS,PLCOST,XLABOR,TWELD
 413  FORMAT( 1H1,* TOTAL COST =*,F10,2/*COST OF PLATING =*,F10,2/
     1*COST OF LABOR =*,F10,2/*COST OF WELDING =*,F10,2//)
 343 RETURN
      END
```

Subroutine CØSTING




E
WBAYTR = WBAYTR + SUMWTR(K)
Longitudinal Seams
WELD1 = WELDL1 = BUTT1 = 0
BUTTS = 0
DØ I = 1, IPLATE
I > ILG
No
Yes
I > ITR
No
Yes
XBEAM = BEAM/360
BUTTS(I) = F1(THIKK1(I,1,1) *XBEAM
CSL(I) = BEAM* F5(THIKK1(I,1,1))
NSEAM = BEAM/72
WTL(I) = BEAM* THIKK1(I,1,K)*SG
WELD(I) NSEAM* F1(THIKK1(I,1,1))
ID = I - ILG
CSL(I) = TWEENH(ID)* FS(THIKK1(I,1,1))*2
BUTTS(I) = TWEENH(ID)* F1(THIKK1(I,1,1))/360*2
WTL(I) = TWEENH(ID)* THIKK1(I,1,K)*2*SG
WELD(I) = F1(THIKK1(I,1,1)*2
IPLT = I - ITR
F
BUTT1 = BUTT1 + BUTTS(I)
WELD1 = WELD1 + WELD(I)
SUMCSL = SUMCSL + CSL(I)
SAREA1 = SAREA1 + WTL(I)
H
G
C
ILG
Yes
ID = I - ILG
Assumed Areas
SAREA(4) = 10.
SAREA(3) = 20.
SAREA(2) = 30.
SAREA(1) = 40.
CSTAN(I) = TWEENH(ID)* 4*SG/XLHØLD*SAREA(I)
CST(I) = F4(AREAT1(I,1,1)* TWEENH(ID)*2/AA
WMAT(I) = F3(WEB1(I,1,1)* 0.036*TWEENH(ID)*2
WTR(I) = AREAT1(I,1,K)* TWEENH(D)*SG*2/AA
A


F
IPLT = 1
2
3
4
5
Bilge
O. T. Floors
NT Floors
OT Longitudinals
NT Longitudinals
BUTTS(I) = HFLØØR* F1(THIKK1(I, 1, 1))/360*π
CSL(I) = F5(THIKK1(I, 1, 1)) *π*HFLØØR
WTL(I) = π · 5 · THIKK1(I, 1, K) *SG
WELD(I) = F1(THIKK1(I, 1, 1))*2
WELD(I) = F2(THIKK1(I, 1, 1))* 2*(BEAM + XNG(I)*HFLØØR) /XLHØLD
CSL(I) = 2/XLHØLD*F5(THIKK1(I, 1, 1)* HFLØØR*BEAM
WTL(I) = 0.0
WELD(I) = 2*F3(THIKK1(I, 1, 1))* XNST*(BEAM + XNG(I)*HFLØØR) /XLHØLD
CSL(I) = XNST/XLHØLD* F5(THIKK(I, 1, 1))*HFLØØR* BEAM
WTL(I) = 0.0
WELD(I) = F2(THIKK1(I, 1, 1))* GNØT
CSL(I) = GNØT*HFLØØR* F5(THIKK1(I, 1, 1))
WTL(I) = THIKK1(I, 1, 1)* HFLØØR*GNØT*SG
WELD(I) = F3(THIKK1(I, 1, 1))* GNNT
CSL(I) = GNNT*HFLØØR* F5(THIKK1(I, 1, 1))
G
H
Print Pertinent Material for Plates
Print Pertinent Material for Transverses
Deck Longitudinals
DØ I = 3, ILG
CSLL(I) = F4(AREAL1(I, 1, 1))
SUMCSLL = SUMCSLL + CSLL(I)
WELDL(I) = XNG(I)*THIKK1(I, 1, 1)
SAREAL(I) = AREAL1(I, 1, K)* XNG(I)*SG
SUMSARE = SUMSARE + SAREAL(I)
WELDL1 = WELDL1 + WELDL(I)
CØNTINUE
A
I
I
Print Pertinent Material for Deck Longitudinals
WBAYLP = SAREA1 + SUMSARE
WBAYTØT = WBAYLP + WBAYTR
PLCØST = DØLLPP*WBAYTØT
TWELD = (WELD1 + WELDL1 + BUTT1 + SUMWMAT)*WPRICE
TWELD = 1.03*TWELD
SIGMAL = SUMCST + SUMCSL + SUMCSLL
XLABØR = SIGMAL* HRATE + SUMSTAN
CØSTS = PLCØST + XLABØR + TWELD
Print Overall Weight and Cost Figures
RETURN

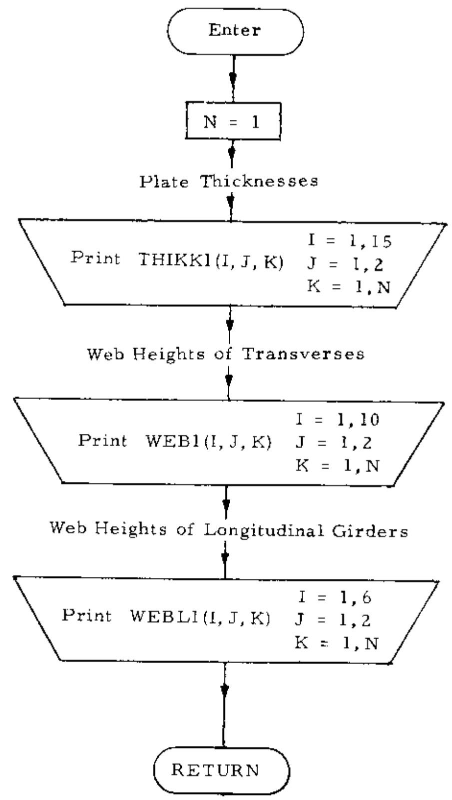
4. Subroutine DOCUMENT

a) Abstract:

Subroutine DOCUMENT is called from program TRANSHIP. It prints out tables of scantlings of all structural members, the values of which have been computed and stored in the memory of the computer.

```
      SUBROUTINE DOCUMENT
      COMMON /B/ A,B,NDECKS,IPLATE,PSL,PRM,HMAIN,WNLUT,TESL,TEST,SESL,
     1     SEST,       KPANELS,BEAM,XNG,HATCH,THIKK,THIK 1,EFFBR,EFFB1,
     2     EFFW,EFFW1,HFLOOR,TWEBNH,THIKX,EFUR,WEB,WEB1,XIP,X,CHI,
     3     XIS, AREAT1,SMAL1,SMODL1,WEBL1,SUMAREAP,SUMAREAL,SUMMP,SUMML,
     4     AREA,SUMAREA,SUMMOM,WPL,XPANEL,WBAYLP,XNST,IST,TTR,
     5     WTR,BAYTR,SUMWT,SUMWTR,LLGX,   YIELD,WBAYTR,XLABOR,PLCOST,
     6     COST,COSTMIN,GNOT,GVNT,KNOUT,SUMSHAP,SUMSMAL,SUMSMA,CODE,
     7WAVEH,PRHEAD,DRAFT,ALOAD,XLHOLD,XLPANEL,AAA,BBB,AA,BB,DELTESL,
     8      SUMSMA1,ZP,ZL,ILG,AREAL1,ASX,   SMAT1,SMODT1
      COMMON /E/ IBT,F
      DIMENSION ZP(17),SESL(17),SEST(17),TESL(17),TEST(17),XNG(7),THIKK(
     117),        THIKK1(17,2,5),EFFBR(17),EFFW(17),EFFB1(17,2,5),EFFW1(17
     2,2,5),AREA(17,5),TWEBNH(5),SMAT1(12,2,5),SMODT1(12,2,5),THIKX(12,2
     3),EFBR(12,2),AREAT1(12,2,5),WEB1(12,2,5),SMAL1(17,2,3),SMODL1(17,2
     4,3),WEBL1(7,2,3),AREAL1(7,2,3),SUMAREAP(5),SUMAREAL(5),SUMMP(5),SU
     5MML(5),SUMAREA(5),SUMMOM(5),WPL(5),XLPANEL(5),SUMWTR(5),WTR(12),ZL
     6(7),TIKK1(17,5),CODE(17),ALOAD(17),SUMSHAP(5),SUMSMAL(5),SUMSMA(5)
      PRINT 20
      N=1
      PRINT 200,(( THIKK1( 1,J,K),J=1,2),K=1,N)
      PRINT   1,(( THIKK1( 2,J,K),J=1,2),K=1,N)
      PRINT   2,(( THIKK1( 3,J,K),J=1,2),K=1,N)
      PRINT   3,(( THIKK1( 4,J,K),J=1,2),K=1,N)
      PRINT   4,(( THIKK1( 5,J,K),J=1,2),K=1,N)
      PRINT   5,(( THIKK1( 6,J,K),J=1,2),K=1,N)
      PRINT   7
      PRINT  13,(( THIKK1(11,J,K),J=1,2),K=1,N)
      PRINT   8,(( THIKK1( 7,J,K),J=1,2),K=1,N)
      PRINT   9,(( THIKK1( 8,J,K),J=1,2),K=1,N)
      PRINT  10,(( THIKK1( 9,J,K),J=1,2),K=1,N)
      PRINT  11,(( THIKK1(10,J,K),J=1,2),K=1,N)
      PRINT 77
      PRINT  14,(( THIKK1(12,J,K),J=1,2),K=1,N)
      PRINT  15,(( THIKK1(13,J,K),J=1,2),K=1,N)
      PRINT  16,(( THIKK1(14,J,K),J=1,2),K=1,N)
      PRINT  17,(( THIKK1(15,J,K),J=1,2),K=1,N)
      PRINT 21
      PRINT 14, (( WEB1 (1,J,K),J = 1,2), K = 1,N)
      PRINT 15, (( WEB1 (1,J,K),J = 1,2), K = 1,N)
      PRINT  2, ((  WEB1(3,J,K),J = 1,2) , K = 1,N)
      PRINT  3, ((  WEB1(4,J,K),J = 1,2) , K = 1,N)
      PRINT  4, ((  WEB1(5,J,K),J = 1,2) , K = 1,N)
      PRINT  5, ((  WEB1(6,J,K),J = 1,2) , K = 1,N)
      PRINT 7
      PRINT  8, ((  WEB1(7,J,K),J = 1,2) , K = 1,N)
      PRINT  9, ((  WEB1(8,J,K),J = 1,2) , K = 1,N)
      PRINT 10, ((  WEB1(9,J,K),J = 1,2) , K = 1,N)
      PRINT 11, ((  WEB1(10,J,K),J=1,2),K=1,N)
      PRINT 22
      PRINT 14, (( WEBL1( 1,J,K),J=1,2),K=1,N)
      PRINT 15, (( WEBL1( 1,J,K),J=1,2),K=1,N)
      PRINT  2, (( WEBL1( 3,J,K),J=1,2),K=1,N)
      PRINT  3, (( WEBL1( 4,J,K),J=1,2),K=1,N)
      PRINT  4, (( WEBL1( 5,J,K),J=1,2),K=1,N)
      PRINT  5, (( WEBL1( 6,J,K),J=1,2),K=1,N)
   20 FORMAT(1H1,*  CENTER HOLD *//*  PLATE THICKNESSES  *//,16X,
     1*   IN WAY OF HATCH                                    *//
     2* LOCATION          PANEL 1   PANEL 2  *//)
21    FORMAT(1H1,* CENTER HOLD *//* WEB HEIGHTS OF TRANSVERS FRAMES OR S
     1HAPES *///,15X,*  IN WAY OF HATCH
     2              *//* LOCATION          PANEL 1   PANEL 2  *//)
22    FORMAT(                         //* WEB HEIGHTS OF LONGITUDINAL GIRDERS
     1          *///,15X,*   IN WAY OF HATCH
     2   *//* LOCATION          PANEL 1   PANEL 2  *//)
  200 FORMAT(* BOTTOM SHELL       *,6(F6.2,4X)/ )
    1 FORMAT(* I.B. PLATING       *,6(F6.2,4X)/ )
    2 FORMAT(* DECK  4            *,6(F6.2,4X)/ )
    3 FORMAT(* DECK  3            *,6(F6.2,4X)/ )
    4 FORMAT(* DECK  2            *,6(F6.2,4X)/ )
    5 FORMAT(* DECK  1            *,6(F6.2,4X)/ )
    7 FORMAT(* SIDE SHELL*/ )
    8 FORMAT(*     I.B.TO 4       *,6(F6.2,4X)/ )
    9 FORMAT(*       4 TO 3       *,6(F6.2,4X)/ )
   10 FORMAT(*       3 TO 2       *,6(F6.2,4X)/ )
   11 FORMAT(*       2 TO 1       *,6(F6.2,4X)/ )
   13 FORMAT(* BILGE              *,6(F6.2,4X)/ )
   14 FORMAT(* O.T. FLOORS        *,6(F6.2,4X)/ )
   15 FORMAT(* N.T. FLOORS        *,6(F6.2,4X)/ )
   16 FORMAT(* O.T. GIRDERS       *,6(F6.2,4X)/ )
   17 FORMAT(* N.T. GIRDERS       *,6(F6.2,4X)/ )
      PRINT  18
18    FORMAT(1H1,*  SHAPE PROPERTIES *,//)
 77   FORMAT(* DOUBLE BOTTOM *//)
      RETURN
      END
```

Subroutine DØCUMENT



5. Subroutine EV

a) Abstract:

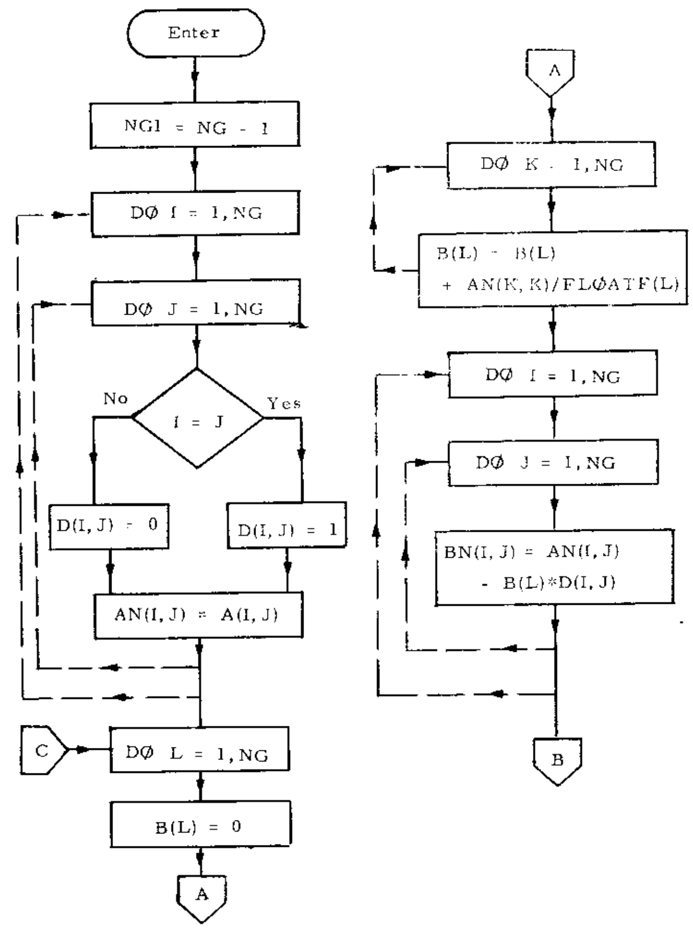
This subroutine is called from subroutine GRILLAGE, BNMAT, XLØAD. It calculates the characteristic equation of the A matrix. The Cayley-Hamilton theorem is used to reduce the characteristic equation of the A matrix.

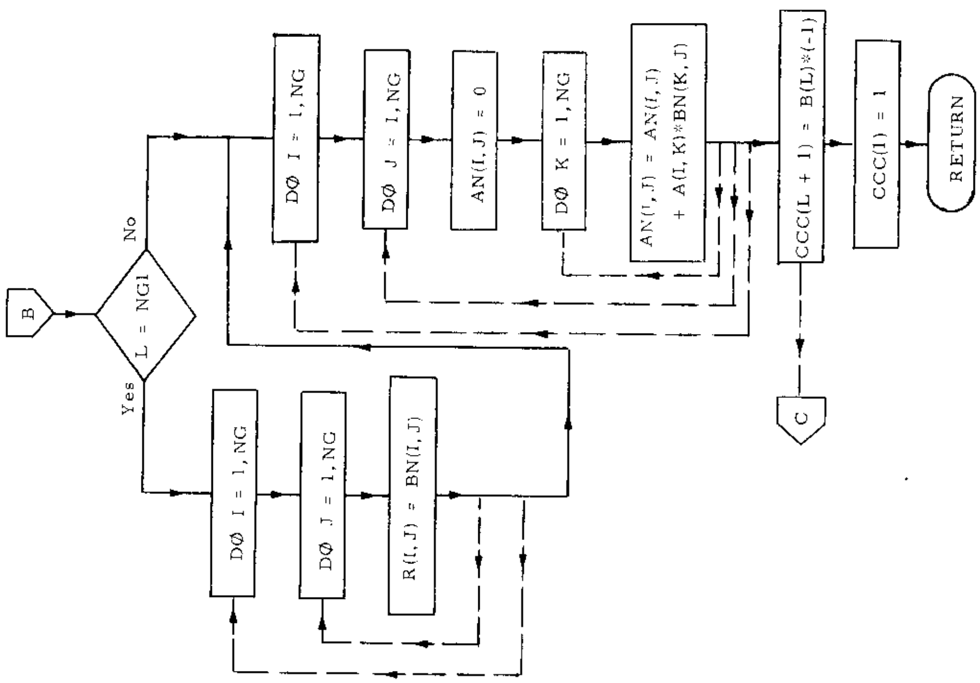
b) Terms specific to this subroutine:

FORTRAN Term	Definition
A	A matrix
CCC	Coefficients of characteristic equation in descending order
D	Identity matrix
NG	Order of matrix (Number of longitudinal girders)
R	Adjoint of A matrix

```
      SUBROUTINE EV(A,NG,R,CCC)
        DIMENSION A(9,9),B(9),D(9,9),AN(9,9),BN(9,9),CCC(10),R(9,9)
       NG1=NG-1
      DO 90 I=1,NG
      DO 90 J=1,NG
       IF(I-J) 1,2,1
1    D(I,J)=0,0
        GO TO 90
2     D(I,J)=1,0
90   AN(I,J)=A(I,J)
       DO 200 L=1,NG
          B(L)=0,0
         DO 110 K=1,NG
110  B(L)=B(L)+AN(K,K)/FLOATF(L)
            DO 120 I=1,NG
            DO 120 J=1,NG
120    BN(I,J)=AN(I,J)-B(L)*D(I,J)
     IF (L-NG1) 230,210,230
210      DO 220 I=1,NG
         DO 220 J=1,NG
220    R(I,J)=BN(I,J)
230      DO 130 I=1,NG
         DO 130 J=1,NG
       AN(I,J)=0,0
          DO 130 K=1,NG
130   AN(I,J)=AN(I,J)+A(I,K)*BN(K,J)
200   CCC(L+1)=B(L)*(-1,0)
        CCC(1)=1,0
         RETURN
          END
```

Subroutine EV(A, NG, R, CC)




B
L = NG1
No
Yes
DØ I = 1, NG
DØ J = 1, NG
AN(I, J) = 0
DØ K = 1, NG
AN(I, J) = AN(I, J)
+ A(I, K)*BN(K, J)
CCC(L + 1) = B(L)*(-1)
CCC(1) = 1
RETURN
C
DØ I = 1, NG
DØ J = 1, NG
R(I, J) = BN(I, J)

6. Subroutine FRAME

a) Abstract:

This subroutine is called from program GRILLAGE and TRANSHIP. It calculates the required frame moments and deflection, using the method of slope deflection. Three options of calculation are provided as follows:

NOC = 1, Unrestrained frame deflections at girder intersections
NOC = 2, Influence coefficients
NOC = 3, Frame bending moment distribution

b) Terms specific to this subroutine:

FORTRAN Term	Definition	Mathematical Symbol
A	Slope deflection matrix	
BB	Fixed bending moment	
D	External head at innerbottom	
DD	External head at the node	
DLL	Height of node above innerbottom	
PD	Deflection array	
T	Joint rotation	θ
TM	Transverse deck beam moment	
V	Loading	
WD	Uniformly distributed load on a given level	
WDL	External load at the node	
WDLX	Fixed end moment corresponding to external head of water	
WDLXT	Fixed end moment corresponding to external head of water	
WDX	Fixed end moment due to a distributed load	
XJ	Stiffness factor	K
XLF	Length of stiffener	
XXM	Array of joint moments	

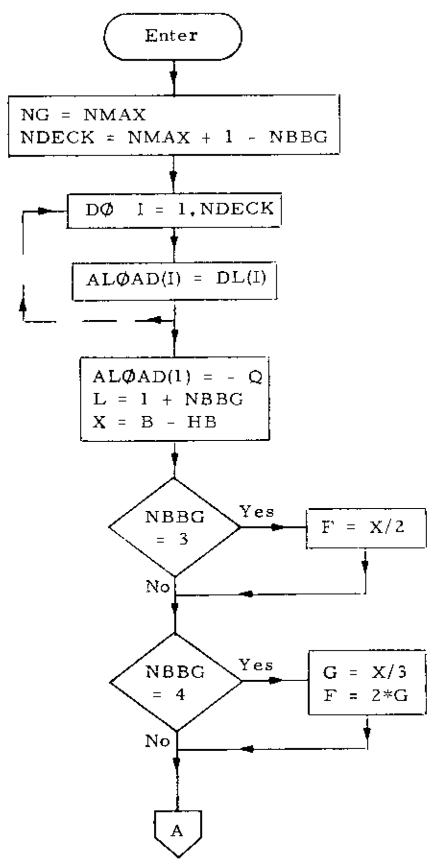
```
      SUBROUTINE FRAME(SMAF,XJ,XLF,DL    ,W,D,H,HB,P,Q,E,NMAX,NOTS,NOC,
     1 PD,EK,NBBG)
      DIMENSION WD(10),WDL(10),DD(10)
      DIMENSION DL(17)
      DIMENSION B(12) ,X(10)
      COMMON /H/ XMAT
      DIMENSION XMMT(17)
      DIMENSION XMT(12)
      DIMENSION XXM(5,6)
       DIMENSION WDLX(12),WDX(12),WDLXT(12),WW(7)
       DIMENSION TM(7)
      DIMENSION  SMAF(15),XLF(15),ALOAD(12),P(7),A(7,7),BB(6),T(7),
     1 XJ(15),IT1(7),IT2(7),IT3(7,2),V(7),XM(7),PD(7)
        NG=NMAX
      NDECK=NMAX+1-NBBG
       DO 3321 I=1,NDECK
 3321 ALOAD(I)=DL(I)
      ALOAD(1)=-Q
      L=1+NBBG
       X   =B-HB
      IF(NBBG.EQ.3)    F=X/2.
      IF(NBBG.EQ.4)    G=X/3.
      IF(NBBG.EQ.4)     F=2.*G
      QF=B-F
      QG=B-G
      DLL=0.
      WD(6)=W
      NTD=NJ+5
      DO 5001 I=7,NTD
      J=I-1
      DLL=DLL+XLF(I)
      IF(D.LT.DLL)  GO TO 4002
      WD(I)=W*(1.-DLL/D)
      WDL(I)=WD(J)-WD(I)
      DD(I)=XLF(I)
      GO TO 5001
 4002 WDL(I)=WD(J)
      WD(I)=0.
      DD(I)=D+XLF(I)-DLL
      IF(DD(I).LT.0.)                  DD(I)=0.
 5001 CONTINUE
 4899  FORMAT(10E10.2)
      DO 4888 I=7,10
       WDX(I)=WD(I)*XLF(I)**2/12.
       WDLX(I)=WDL(I)*DD(I)**2 /
     1 XLF(I)**2.0*(XLF(I)**2.0/6.0-XLF(I)*DD(I)/6.+DD(I)**2/20.)
 4888 WDLXT(I)=WDL (I)*DD(I)**3/XLF(I)**2.*(XLF(I)/12.-DD(I)/20.)
      WDX(11)=0.
      WDLXT(11)=0.
      IF(NOC.NE.2)  GO TO 302
      DO 204 KT=1,7
  204 ALOAD(KT)=0.0
C       BULKHEAD BAY
  302 IF(NOTS.EQ.2)  GO TO 6999
       XHB=0.
      DO 6000  I=1,10
      XJ(I)= SMAF(I)/XLF(I)
 6000 CONTINUE
C  FIXED BENDING MOMENT CALCULATION FOR BULKHEAD BAY
      BB(1)= P(1)*XLF(1)/4.0+P(2)*(B*B-HB*HB)/2.0/B-Q*B*B/3.
      BB(1)=BB(1)+WDX(7) +WDLX(7)
      IF(NBBG.EQ.3)    BB(1)=BB(1)+P(NBBG)*(B**2-QF**2)/2./B
      IF(NBBG.EQ.4)    BB(1)=BB(1)+P(3)*(B**2-QF**2)/2./ +P(4)*(B**2-
     1  QG**2)/2./B
      DO 10  N=2,NDECK
      L=NBBG+N-1
      M=N+6
      M1=M-1
       BB(N)=P(L)*(B(N)**2-HB**2)/2./B(N) +ALOAD(N)*B(N)**2/3.
     1 + WDX(M)-WDX(M1)+WDLX(M)-WDLXT(M1)
   10 CONTINUE
      GO TO 5999
 6999 CONTINUE
C       HATCH BAY
       XHB=HB
      NDECK1=NDECK+1
      DO 6749 I=3,NDECK1
      XLF(I)=B(I)-XHB
 6749 CONTINUE
      DO 6700  I=1,10
      XJ(I)= SMAF(I)/XLF(I)
 6700 CONTINUE
C  FIXED BENDING MOMENT CALCULATION FOR  HATCH BAY
      BB(1)= P(1)*XLF(1)/4.0+P(2)*(B*B-HB*HB)/2.0/B-Q*B*B/3.
      BB(1)=BB(1)+WDX(7) +WDLX(7)
      IF(NBBG.EQ.3)    BB(1)=BB(1)+P(NBBG)*(B**2-QF**2)/2./B
      IF(NBBG.EQ.4)    BB(1)=BB(1)+P(3)*(B**2-QF**2)/2./B+P(4)*(B**2-
     1  QG**2)/2./B
      DO 9 N=2,NDECK
      L=NBBG+N-1
      M=N+6
      M1=M-1
      NN=N+1
       BB(N)=P(L)*(B(N)-HB)+ALOAD(N)*(B(N)-HB)**2/2.
     1 + WDX(M)-WDX(M1)+WDLX(M)-WDLXT(M1)
      XLF(NN)=B(I)-XHB
    9 CONTINUE
 5999 CONTINUE
C       SLOPE DEFLECTION MATRIX
      A(1,1)= E*(XJ(1)+4*XJ(7))
      A(1,2)= 2.0*E*XJ(7)
      A(1,3)= 0.0
      A(1,4)=0.
      A(1,5)=0.
      A(2,1)=2.0*E*XJ(7)
      A(2,2)=E*4*(XJ(7)+XJ(8))
      IF(NOTS.EQ.1)     A(2,2)=A(2,2)+E*XJ(3)
      A(2,3)=2.0*E*XJ(8)
      A(2,4)=0.
       A(2,5)=0.
      A(3,1)=0.0
      A(3,2)=2.0*E*XJ(8)
```

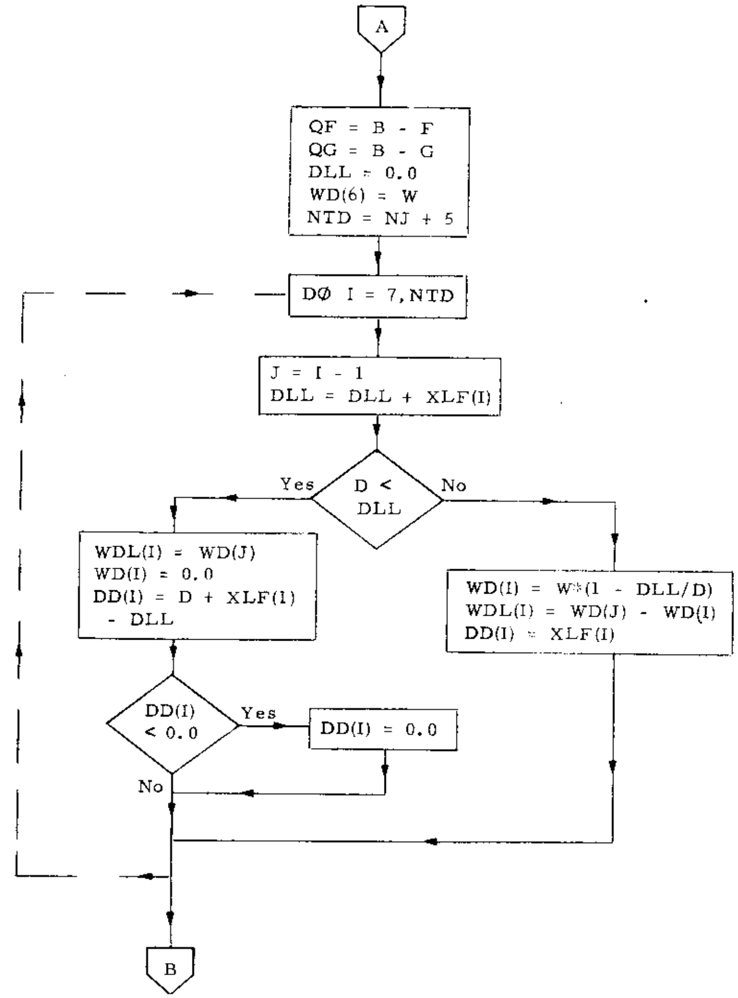
```
(8)  +E*4,*XJ(9)
  A(3,3)=A(3,3)+E*XJ(4)
(9)

(9)
J(9)+XJ(10))
  A(4,4)=A(4,4)+E*XJ(5)
(10)

10)
10)
  A(5,5)=A(5,5)+E*XJ(6)
ALCULATION
NJ,7  ,A,D,IT1, IT2, IT3, DETRM)

,J)*RB(J)
CULATION
*(2,*T(2)+T(1))+ WDX(7)+WDLXT(7)
)*(2,0*T(2)+T(3))-WDX(8)-WDLX(8)
)*(2,0*T(3)+T(2))+WDX(8)+WDLXT(8)
)*(2,0*T(3)+T(4)) +WDX(9)-WDLX(9)
)*(2,0*T(4)+T(3)) +WDX(9)+WDLXT(9)
))*(2,0*T(4)+T(5))-WDX(10)-WDLX(10)
))*(2,0*T(5)+T(4))+WDX(10)+WDLXT(10)
T(1)-P(1)*2/4,0-P(2)*(B*B-HB*HB)/2,0/B+Q*B*B/3,0
(1)=XM(1)-P(3)*(B**2-QF**2)/2,/B
(1)=XM(1)-P(3)*(B**2-QF**2)/2,/B-P(4)*(B**2-QG**2)

23

3

XM(1)=E*XJ(1)*T(1)+Q*B(1)**2/3,
9590,9575
S  ON DECK ELEMENTS

VERSE DECK BEAM   MOMENTS *,/)

CK

(I)*(H-XHR)
OAD(1)-P(1)/2, -P(2)
(1)=V(1)-P(3)
=V(I)-P(L)
NTS*,/,* LOCATION FROM *,/,*LOWER JOINT *,I3,/)
NTS*,/,* LOCATION FROM*,/,* NODE *,I3,/)
```

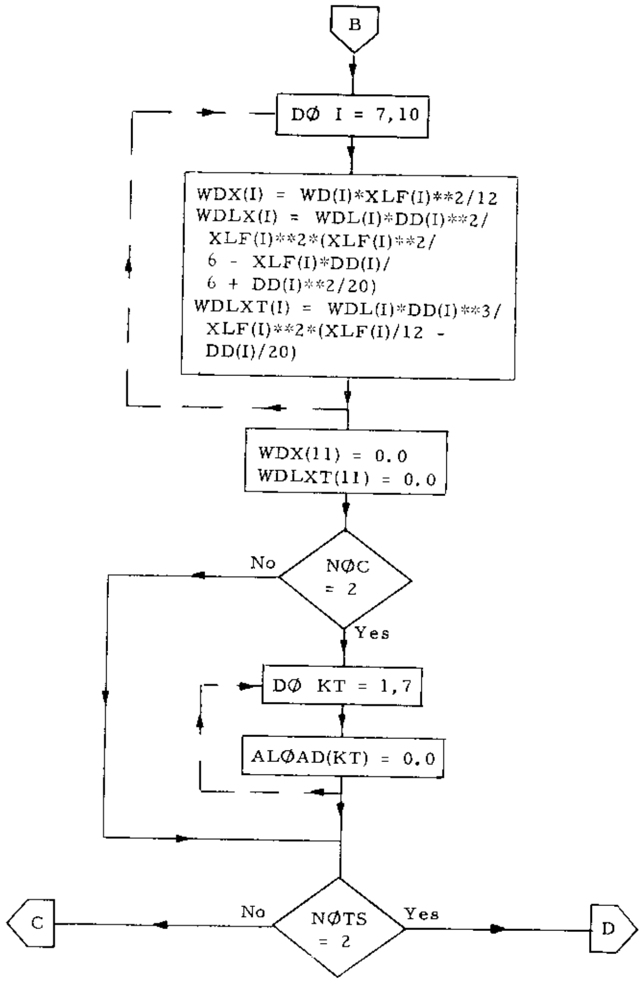
```
      NMP1=NMP+1
       DO 3661 J=1,NMP1
      KJ=J-1
      Z= B (I)-KJ/NMP
      IF(NDS.EQ.1.OR.I.EQ.1)  GO TO 3658
      IF(Z.LT.X)  GO TO 3658
2331  Z=B(I)-XMP
      X(I)=B(I)-HB
       J=NMP1
3658  TM(I)=-(-XM(I)+V(I)*Z)-ALOAD(I)*Z**2/2,
      IF(I.GT.1.OR.NBRG.LT.3)  GO TO 4211
      IF(NBRG.EQ.4 .AND.Z.GT.G)   TM(1)=TM(1)-P(4)*(Z-G)
      IF(Z.GT.F) TM(1)=TM(1)-P(3)*(Z-F)
4211  IF(Z.LT.X)  GO TO 3657
      TM(1)=TM(1)-P(2)*(Z-X)
C   FOR BULKHEAD BAY
      IF(I.GT.1)  TM(I)=TM(I)-P(L)*(Z-X)
3657  TMM=TM(I)
       PRINT 5050,TMM ,Z
      IF(KJ.EQ.0) GO TO 3661
      JH=I+1
      IF(ABSF(TMM)-XMT(JH))3661,3661,9931
9931  XMT(JH)=ABSF( TMM )
3661  CONTINUE
3660  CONTINUE
      XXM(1,2)=XM12
      XXM(2,1)=XM21
      XXM(2,3)=XM23
      XXM(3,2)=XM32
      XXM(3,4)=XM34
      XXM(4,3)=XM43
      XXM(4,5)=XM45
      XXM(5,4)=XM54
      ISS=NDECK+2
      ISST=ISS+NJ-2
      DO 3881 I=ISS,ISST
      PRINT 5059,I
      DO 3881  J=1,NMP1
      KJ=J-1
       Z=XLF(I)*KJ/NMP
      K6=I-6
      K5=I-5
      V(I)= . -((XXM(K5,K6)-XXM(K6,K5))/XLF(I)+(WDL(I)/3,
     1  +WD(I)/2,)*XLF(I)   )
      TM(I)=-(-XXM(K6,K5) +V(I)*Z)-(WD(I)+WDL(I))*Z**2/2,+WDL(I)*Z**3/6,
     1  /XLF(I)
      TMM=TM(I)
       PRINT 5050 ,TMM,Z
      IF(KJ.EQ.0.OR.KJ.EQ.NMP)  GO TO 3881
      IF(ABSF(TMM).LT.XMT(I))   GO TO 3881
      XMT(I)=ABSF(TMM)
3881  CONTINUE
8688  FORMAT(5E13.5)
      NJ2=NJ-1
      PRINT 4032
4032  FORMAT(/,*   UPWARD JOINT MOMENTS *,/)
```

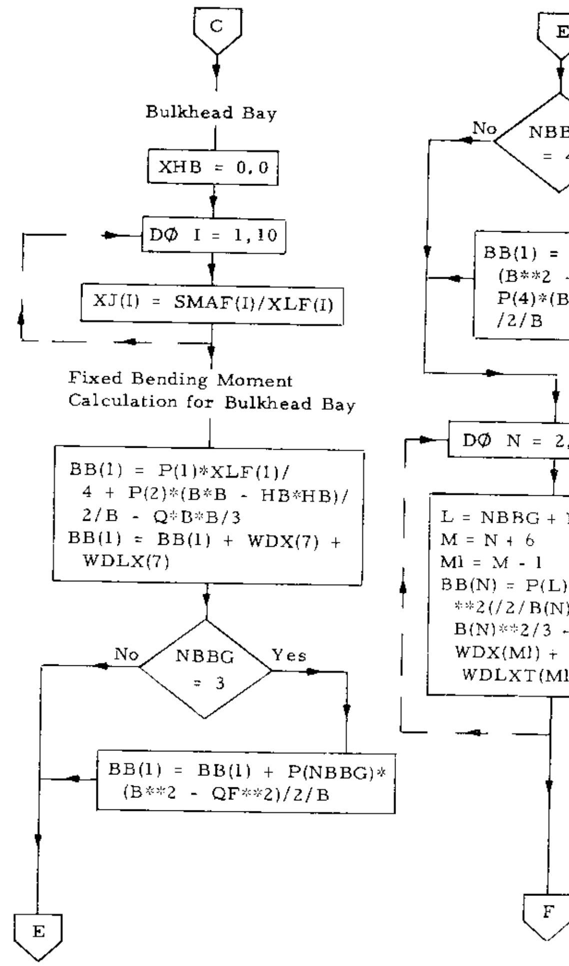
```
9521 PRINT 4030,(XXM(I,J),J=2,NJ)
     PRINT 4031
4031 FORMAT(/,* DOWNWARD JOINT MOMENTS *,/)
     DO 9522,I=2,NJ
9522 PRINT 4030,(XXM(I,J),J=1,NJ2)
4030 FORMAT(/,5E13.5,/)
C    SIDE SHELL
     DO 9630 I=1,ISST
     IF(XMMT(I)-XMT(I)) 9631,9631,9630
9631 XMMT(I)=XMT(I)
9630 CONTINUE
      XMMT(1)=XMMT(2)
     PRINT 5056
5056 FORMAT(/,* MAXIMUM MOMENTS ON TRANSVERSE ELEMENTS *,/)
     PRINT 5050,(XMMT(I),I=1,10)
C  DEFLECTION CALCULATION
9575 DO 6290 I=   2,NDECK
     J=NBBG-1+I
     L=I+1
     V(I)=-ALOAD(I)*(B(I)-XHB)
     PD(J)=(-(-XM(I)*X**2,0/2,0 + V(I)*X**3,0/6,0 )          -ALOAD(I)*
    1 X**4,0/24,0  )/E/SMAF(L)  -T(I)*X
6290 CONTINUE
      IF(NBBG.LT.3)  GO TO 9550
     PD(3)=(-(-XM(1)*F**2,0/2,0 + V(1)*F**3,0/6,0 ) -ALOAD(1)*
    1 F**4,0/24,0  )/E/SMAF(L)  -T(1)*F
     IF(NBBG,EQ,4)  PD(4)=
    1        (-(-XM(1)*G**2,0/2,0 + V(1)*G**3,0/6,0 ) -ALOAD(1)*
    2 G**4,0/24,0  )/E/SMAF(L)  -T(1)*G
9550 IF(NOC,NE,2)  GO TO 7570
C   INFLUENCE COEF,
9600 IF (KK-1) 9400,9500,9400
9500 V(1)=-0,5*P(1)
     PD(1)=(-(-XM(1)*B**2,0/2,0 + V(1)*B**3,0/6,0 ))/E/SMAF(1)-T(1)*B
     PD(2)=(-(-XM(1)*X**2,0/2,0 + V(1)*X**3,0/6,0 ))/E/SMAF(1)-T(1)*X
     IF(NBBG.LT.3)    GO TO 9551
     PD(3)=(-(-XM(1)*F**2,0/2,0 + V(1)*F**3,0/6,0 ))/E/SMAF(1)-T(1)*F
     IF(NBBG,EQ,4)  PD(4)=
    1        (-(-XM(1)*G**2,0/2,0 + V(1)*G**3,0/6,0 ))/E/SMAF(1)-T(1)*F
9551 CONTINUE
     V(1)=0,0
9400 IF(KK,EQ,1,) GO TO 9200
C   DEFLECTION UNDER LOAD
     V(KK)=-1,0* P(KK)
     I=KK-NBBG+1
     IF(KK,LE,2)  GO TO 979
     L=I+1
     PD(KK)=(-(-XM(I)*X**2,0/2,0 +V(KK)*X**3,0/6,0 ) )/E/SMAF(L)-T(I)*X
 979 CONTINUE
     PD(1)=(-(-XM(1)*B**2,0/2,0 + V(2)*B**3,0/6,0 )
    1            -P(2)*HB**3,/6,     )/E/SMAF(1)  -T(1)*B
     PD(2)=(-(-XM(1)*X**2,0/2,0 + V(2)*X**3,0/6,0 )
    1                               )/E/SMAF(1)  -T(1)*X
     IF(NBBG.LT.3, OR.KK,NE,3)  GO TO 9325
     PD(2)=PD(2)+(-P(3)*(X-F)**3/6,-V(3)*X**3/6,)/E /SMAF(1)
     PD(1)=PD(1)+(-P(3)*(B-F)**3/6,-V(3)*B**3/6,)/E /SMAF(1)
9325 IF(NBBG,GE,3,AND,KK,EQ,3)  PD(3)=
    1 (-(-XM(1)*F**2,0/2,0 + V(3)*F**3,0/6,0 ))/E/SMAF(1)-T(1)*F
     IF(NBBG,GE,4,AND,KK,EQ,3)  PD(4)=
    1 (-(-XM(1)*G**2,0/2,0 + V(3)*G**3,0/6,0 ))/E/SMAF(1)-T(1)*G
     IF(NBBG,GE,3,AND,KK,EQ,2)  PD(3)=
    1        (-(-XM(1)*F**2,0/2,0 + V(2)*F**3,0/6,0 ))/E/SMAF(1)-T(1)*B
     IF(NBBG,EQ,4,AND,KK,LT,NBBG)    PD(4)=
    1        (-(-XM(1)*G**2,0/2,0 + V(2)*G**3,0/6,0 ))/E/SMAF(1)-T(1)*G
     IF(NBBG,EQ,4,AND,KK,EQ,3)  PD(4)=PD(4)  -V(3)*G**3/6,/E/SMAF(1)
     IF(NBBG,LT,4,OR, KK ,NE,4)  GO TO 66
     PD(1)=PD(1)+(-P(4)*(B-G)**3/6,-V(4)*B**3/6,)/E/SMAF(1)
     PD(2)=PD(2)+(-P(4)*(X-G)**3/6,-V(4)*X**3/6,)/E /SMAF(1)
     PD(3)=  (XM(1)*F**2/2,) /E/SMAF(1)-T(1)*F
     PD(3)=PD(3)+(-P(4)*(F-G)**3/6,-V(4)*F**3/6,)/E /SMAF(1)
  66  CONTINUE
     IF(NBBG,GE,3,AND,KK,GT,NBBG)  PD(3)=
    1        (-(-XM(1)*F**2,0/2,0                        ))/E/SMAF(1)-T(1)*H
     IF(NBBG,EQ,4,AND,KK,GT,4)  PD(4)=
    1        (-(-XM(1)*G**2,0/2,0                        ))/E/SMAF(1)-T(1)*G
     V(KK)=0,0
7570 IF(NOC,NE,1)  GO TO 9200
C    LOAD DEFLECTIONS
      V(1)=-ALOAD(1)*B
     PD(1)=(-(-XM(1)*B**2,0/2,0 + V(1)*B**3,0/6,0 ) -ALOAD(1)*
    1 B**4,0/24,0                              )/E/SMAF(1)  -T(1)*B
     PD(2)=(-(-XM(1)*X**2,0/2,0 + V(1)*X**3,0/6,0 ) -ALOAD(1)*
    1 X**4,0/24,0                              )/E/SMAF(1)  -T(1)*X
     IF(NBBG,LT,3)  GO TO 7675
     PD(3)=(-(-XM(1)*F**2,0/2,0 + V(1)*F**3,0/6,0 ) -ALOAD(1)*
    1 F**4,0/24,0  )/E/SMAF(1)  -T(1)*F
     IF(NBBG,EQ,4)  PD(4)=
    1        (-(-XM(1)*G**2,0/2,0 + V(1)*G**3,0/6,0 ) -ALOAD(1)*
    2 G**4,0/24,0  )/E/SMAF(1)  -T(1)*G
7675 CONTINUE
     WRITE (3,4020) (XM(I), I=1,NJ)
     WRITE (3,4020) (T(I), I=1,NJ)
9200 CONTINUE
9999 RETURN
4000 FORMAT (1H0,    28H JOINT MOMENTS                     F12,2/)
4010 FORMAT (1H0,     13H FIXED MOMENT ,  12H               /9E13,5)
4020 FORMAT (1H0,    7H SLOPE ,11H             RAD / 9E13,5)
6200  FORMAT (1H0,12E10,3/12E10,3)
     END
```

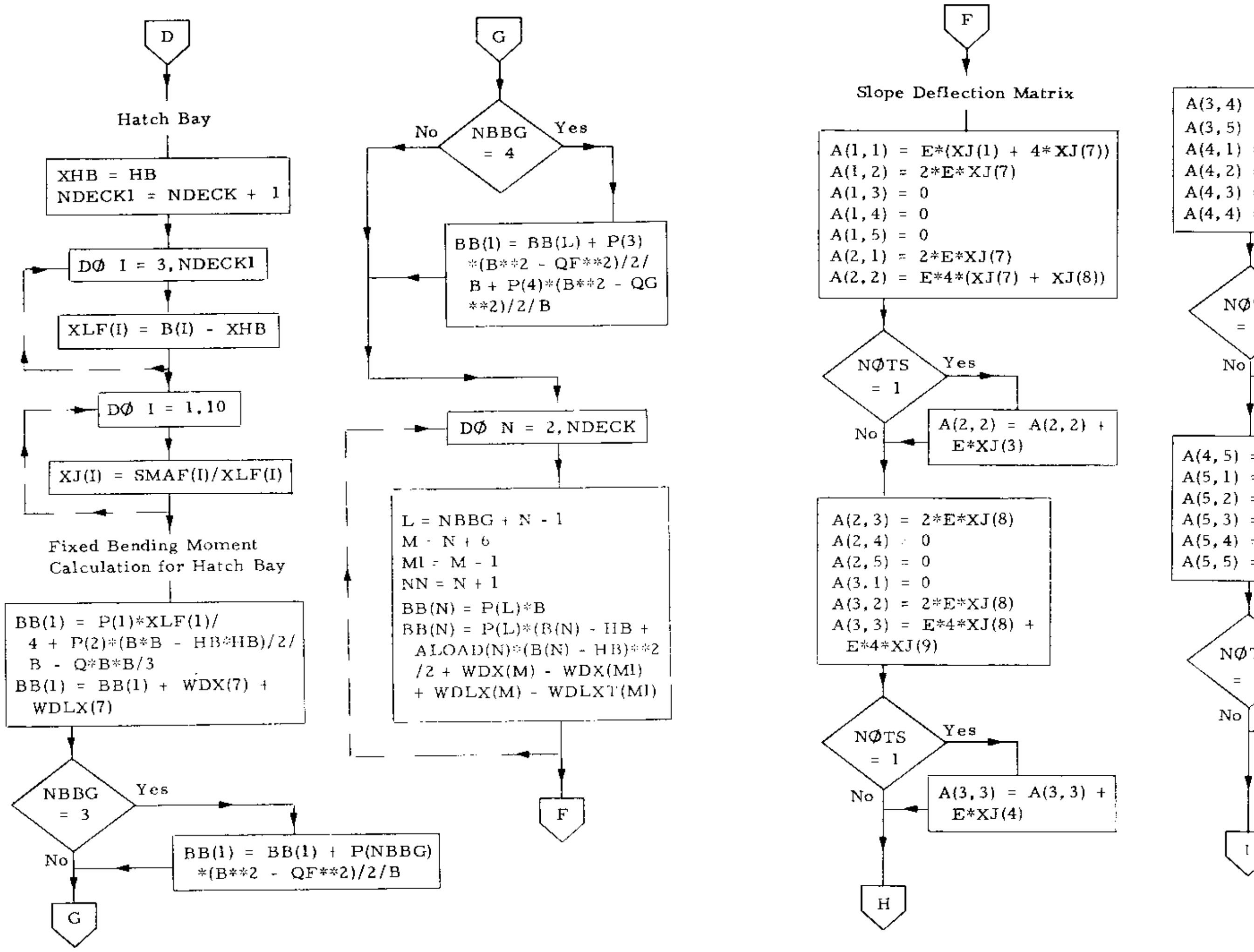
Subroutine FRAME(SMAF, NJ, XLF, DL, W, D, B, HB, P, Q, E, NMAX, NØTS, NØC, PD, KK, NBBG)

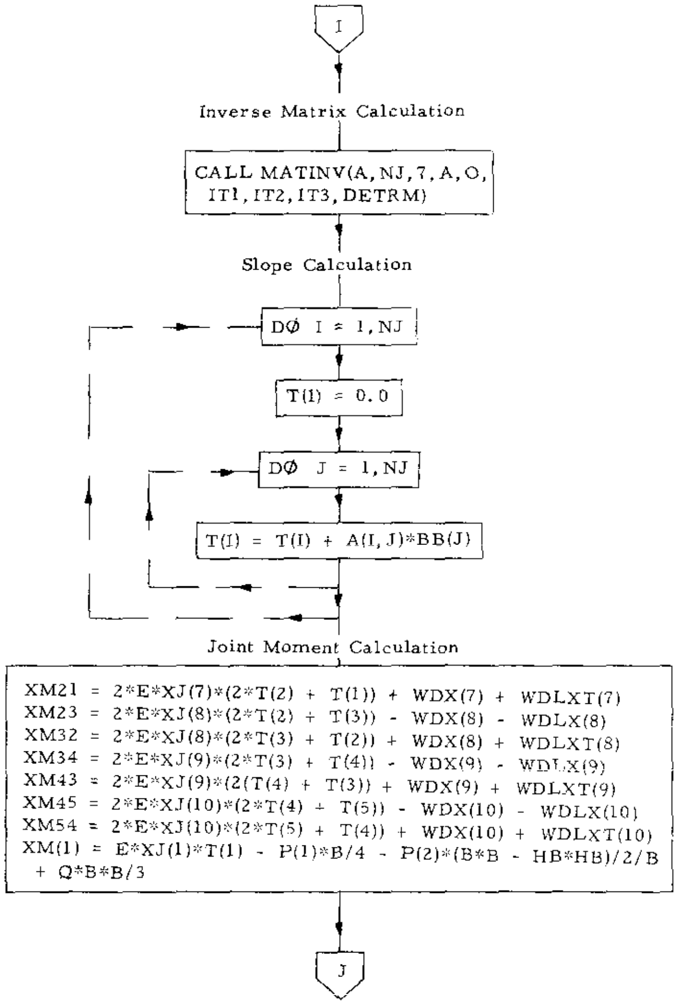


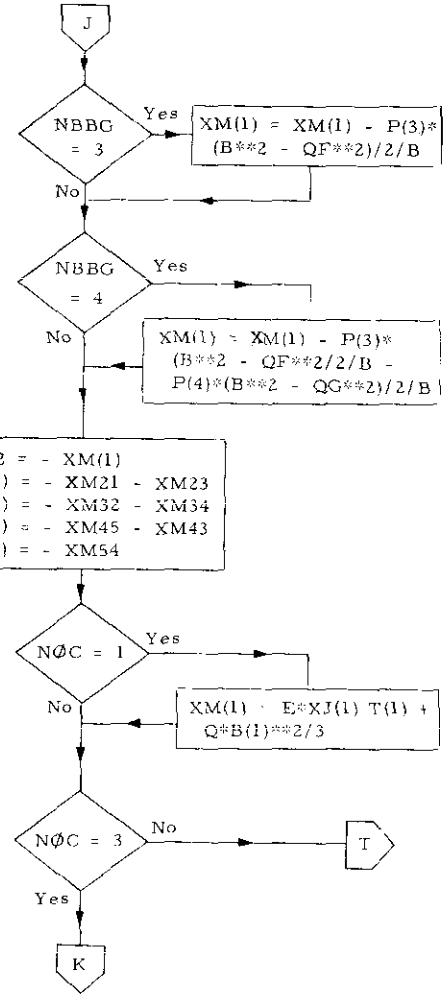


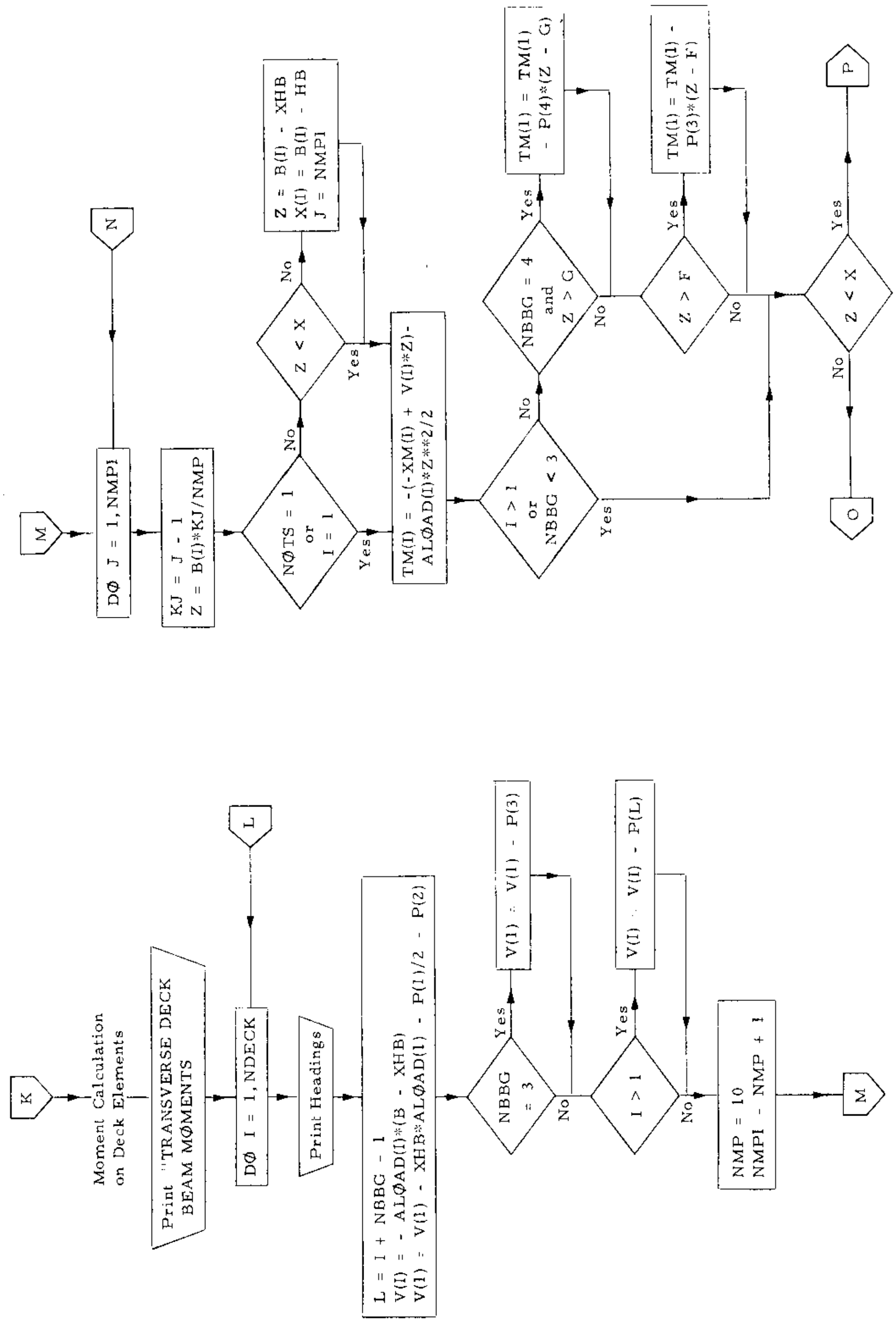

B
DØ I = 7,10
WDX(I) = WD(I)*XLF(I)**2/12
WDLX(I) = WDL(I)*DD(I)**2/
XLF(I)**2*(XLF(I)**2/
6 - XLF(I)*DD(I)/
6 + DD(I)**2/20)
WDLXT(I) = WDL(I)*DD(I)**3/
XLF(I)**2*(XLF(I)/12 -
DD(I)/20)
WDX(11) = 0.0
WDLXT(11) = 0.0
NØC = 2
No
Yes
DØ KT = 1,7
ALØAD(KT) = 0.0
NØTS = 2
No
Yes
C
D

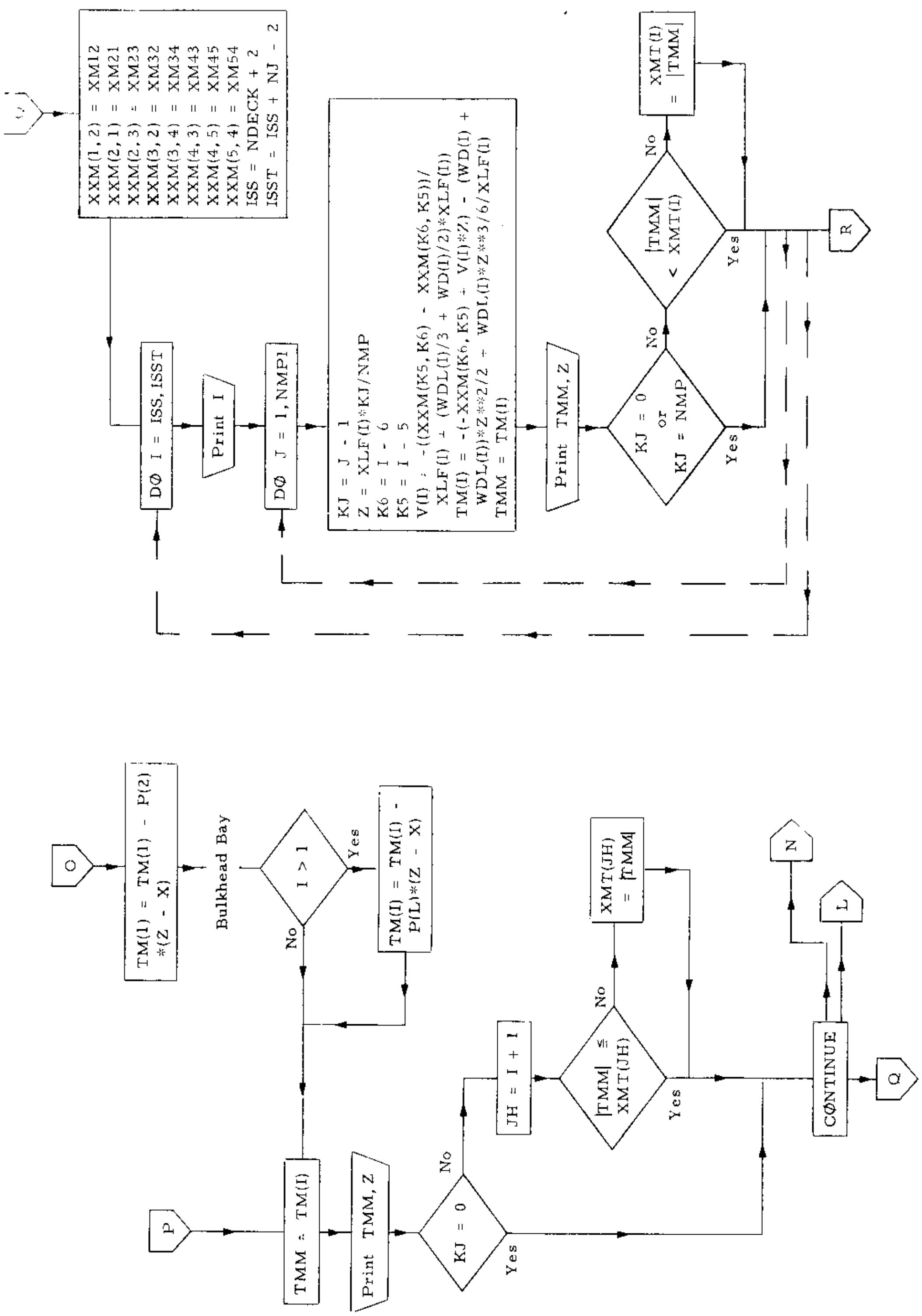

C
Bulkhead Bay
XHB = 0.0
DØ I = 1,10
XJ(I) = SMAF(I)/XLF(I)
Fixed Bending Moment
Calculation for Bulkhead Bay
BB(1) = P(1)*XLF(1)/
4 + P(2)*(B*B - HB*HB)/
2/B - Q*B*B/3
BB(1) = BB(1) + WDX(7) +
WDLX(7)
NBBG = 3
No
Yes
BB(1) = BB(1) + P(NBBG)*
(B**2 - QF**2)/2/B
E
No
BB(1) =
(B**2
P(4)*(B
/2/B
DØ N = 2,
L = NBBG +
M = N + 6
M1 = M - 1
BB(N) = P(L)
**2(/2/B(N)
B(N)**2/3
WDX(M1) +
WDLXT(M1
F

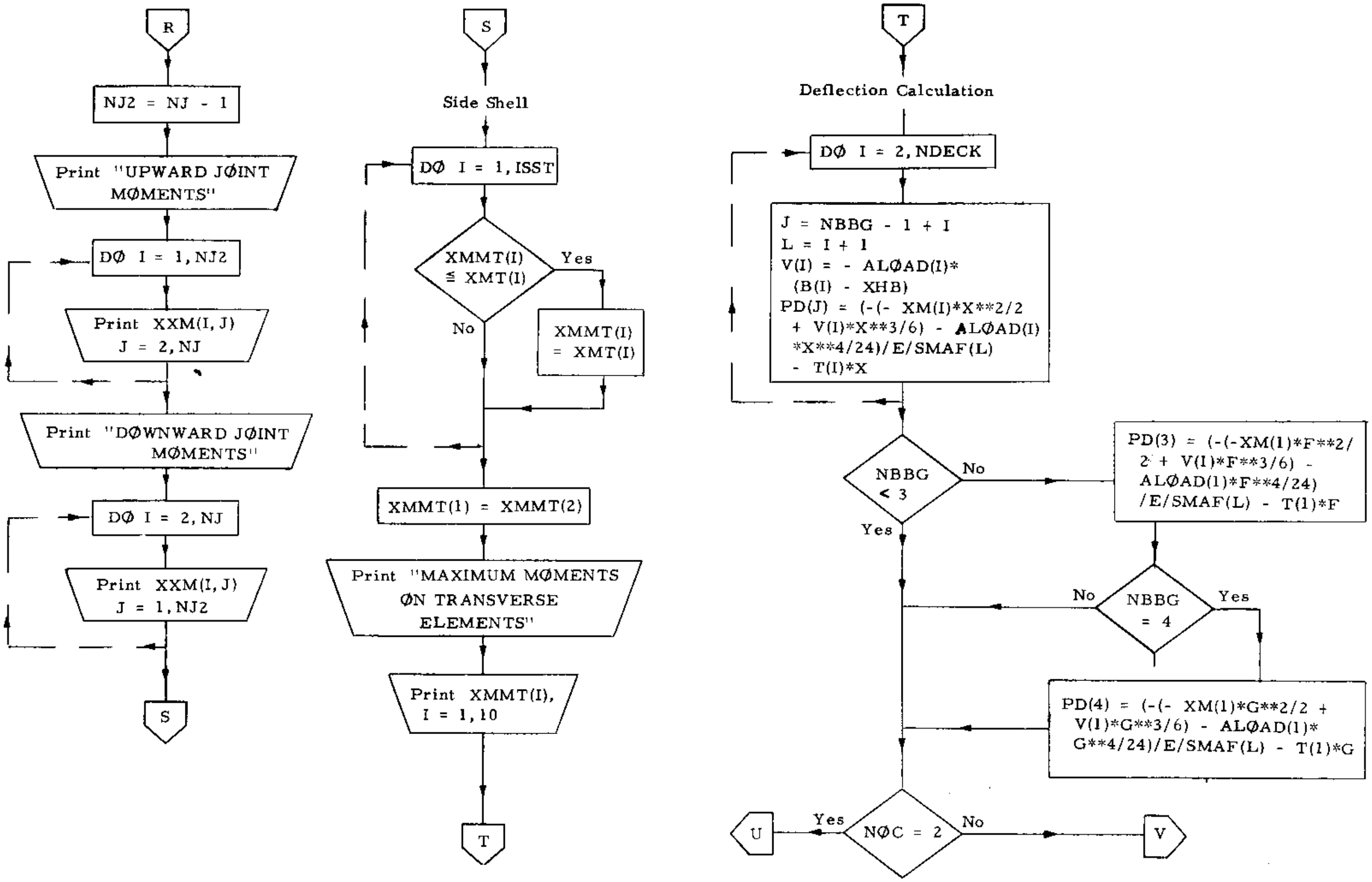

D
Hatch Bay
XHB = HB
NDECK1 = NDECK + 1
DØ I = 3, NDECK1
XLF(I) = B(I) - XHB
DØ I = 1, 10
XJ(I) = SMAF(I)/XLF(I)
Fixed Bending Moment
Calculation for Hatch Bay
BB(1) = P(1)*XLF(1)/4 + P(2)*(B*B - HB*HB)/2/B - Q*B*B/3
BB(1) = BB(1) + WDX(7) + WDLX(7)
NBBG = 3
Yes
No
BB(1) = BB(1) + P(NBBG)*(B**2 - QF**2)/2/B
G
G
NBBG = 4
No
Yes
BB(1) = BB(L) + P(3)*(B**2 - QF**2)/2/B + P(4)*(B**2 - QG**2)/2/B
DØ N = 2, NDECK
L = NBBG + N - 1
M = N + 6
M1 = M - 1
NN = N + 1
BB(N) = P(L)*B
BB(N) = P(L)*(B(N) - HB + ALOAD(N)*(B(N) - HB)**2/2 + WDX(M) - WDX(M1) + WDLX(M) - WDLXT(M1)
F
F
Slope Deflection Matrix
A(1, 1) = E*(XJ(1) + 4*XJ(7))
A(1, 2) = 2*E*XJ(7)
A(1, 3) = 0
A(1, 4) = 0
A(1, 5) = 0
A(2, 1) = 2*E*XJ(7)
A(2, 2) = E*4*(XJ(7) + XJ(8))
NØTS = 1
Yes
No
A(2, 2) = A(2, 2) + E*XJ(3)
A(2, 3) = 2*E*XJ(8)
A(2, 4) = 0
A(2, 5) = 0
A(3, 1) = 0
A(3, 2) = 2*E*XJ(8)
A(3, 3) = E*4*XJ(8) + E*4*XJ(9)
NØTS = 1
Yes
No
A(3, 3) = A(3, 3) + E*XJ(4)
H
A(3, 4)
A(3, 5)
A(4, 1)
A(4, 2)
A(4, 3)
A(4, 4)
No
A(4, 5)
A(5, 1)
A(5, 2)
A(5, 3)
A(5, 4)
A(5, 5)
No

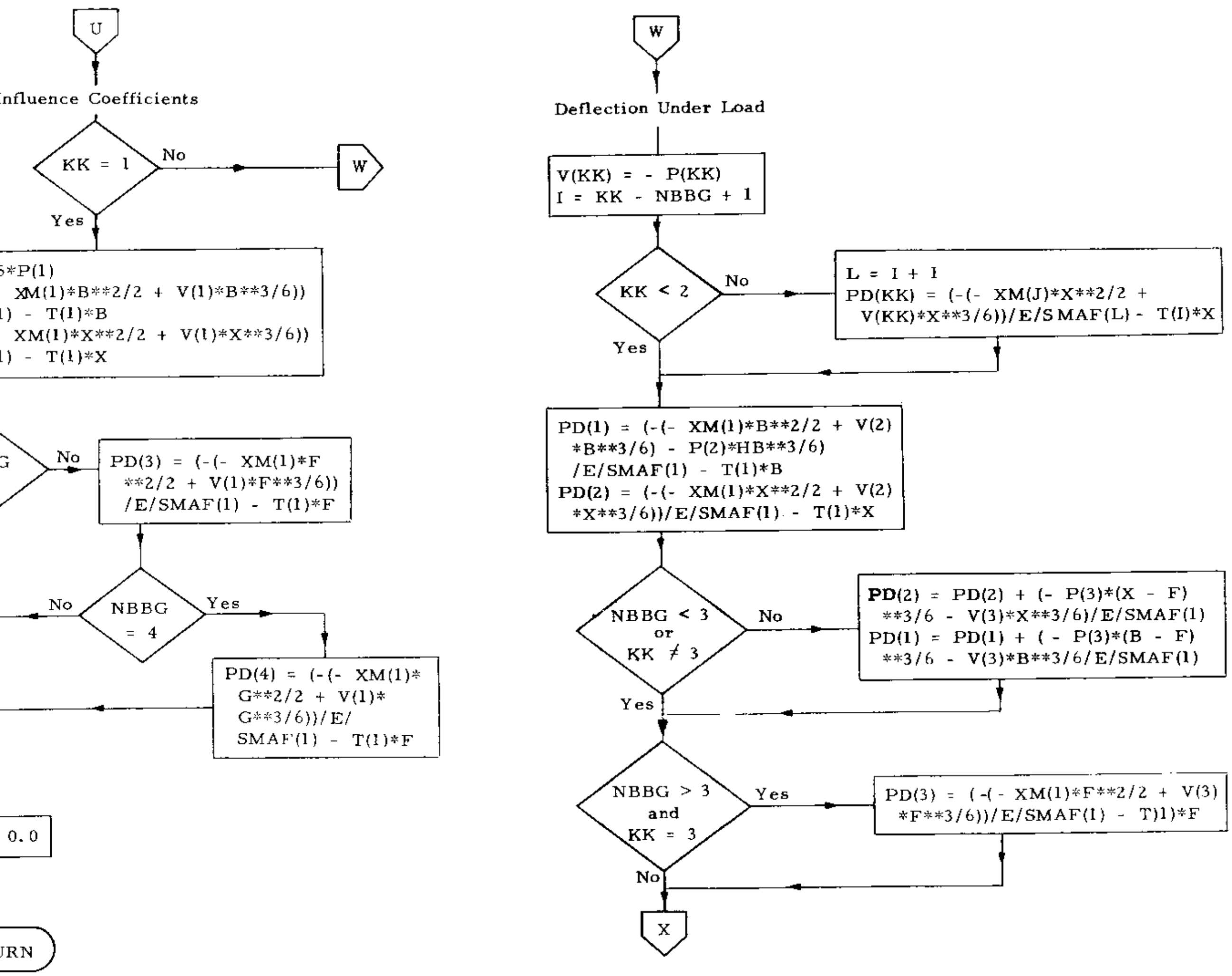

I
Inverse Matrix Calculation
CALL MATINV(A,NJ,7,A,O,
IT1,IT2,IT3,DETRM)
Slope Calculation
DØ I = 1,NJ
T(I) = 0.0
DØ J = 1,NJ
T(I) = T(I) + A(I,J)*BB(J)
Joint Moment Calculation
XM21 = 2*E*XJ(7)*(2*T(2) + T(1)) + WDX(7) + WDLXT(7)
XM23 = 2*E*XJ(8)*(2*T(2) + T(3)) - WDX(8) - WDLX(8)
XM32 = 2*E*XJ(8)*(2*T(3) + T(2)) + WDX(8) + WDLXT(8)
XM34 = 2*E*XJ(9)*(2*T(3) + T(4)) - WDX(9) - WDLX(9)
XM43 = 2*E*XJ(9)*(2(T(4) + T(3)) + WDX(9) + WDLXT(9)
XM45 = 2*E*XJ(10)*(2*T(4) + T(5)) - WDX(10) - WDLX(10)
XM54 = 2*E*XJ(10)*(2*T(5) + T(4)) + WDX(10) + WDLXT(10)
XM(1) = E*XJ(1)*T(1) - P(1)*B/4 - P(2)*(B*B - HB*HB)/2/B
+ Q*B*B/3
J

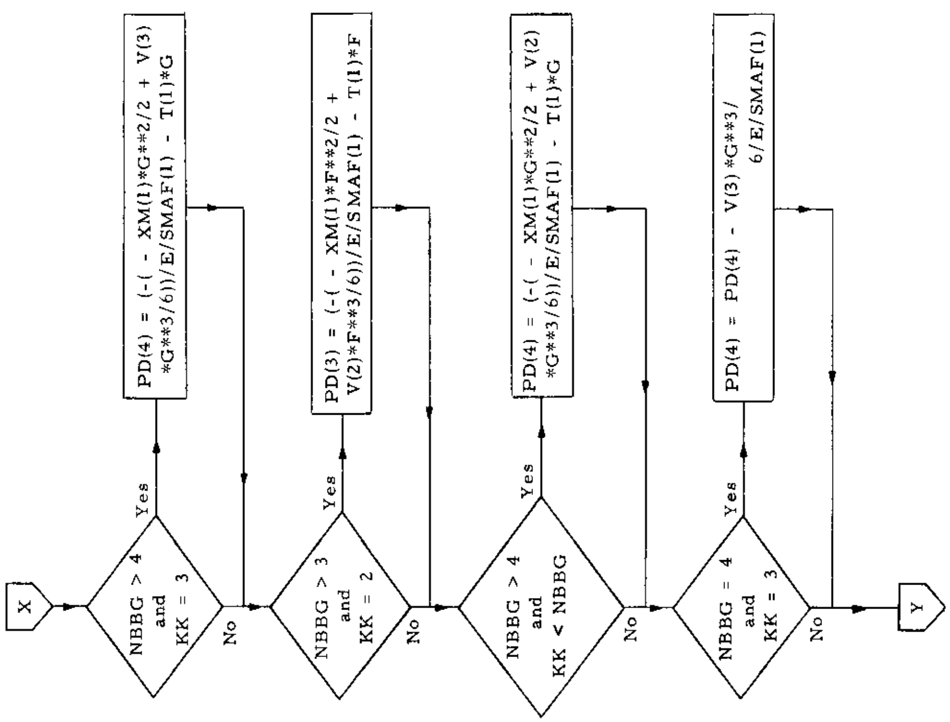

J
NBBG
= 3
Yes
XM(1) = XM(1) - P(3)*
(B**2 - QF**2)/2/B
No
NBBG
= 4
Yes
XM(1) = XM(1) - P(3)*
(B**2 - QF**2/2/B -
P(4)*(B**2 - QG**2)/2/B
No
XM12 = - XM(1)
XM(2) = - XM21 - XM23
XM(3) = - XM32 - XM34
XM(4) = - XM45 - XM43
XM(5) = - XM54
NØC = 1
Yes
XM(1) E*XJ(1) T(1) +
Q*B(1)**2/3
No
NØC = 3
No
T
Yes
K

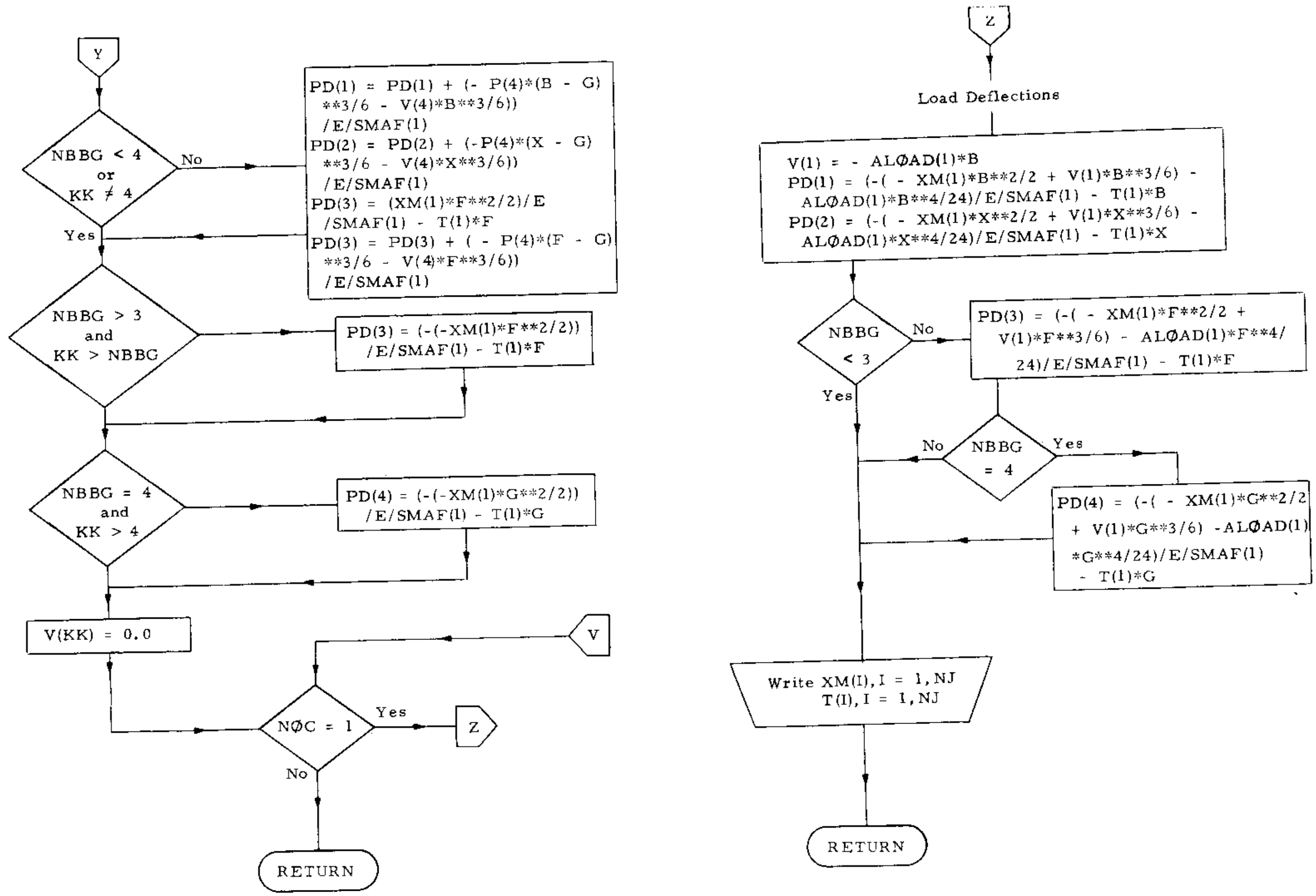

K
Moment Calculation on Deck Elements
Print "TRANSVERSE DECK BEAM MØMENTS
L
DØ I = 1, NDECK
Print Headings
L = I + NBBG - 1
V(I) = - ALØAD(I)*(B - XHB)
V(1) = V(1) - XHB*ALØAD(1) - P(1)/2 - P(2)
NBBG = 3
Yes
V(1) = V(1) - P(3)
No
I > 1
Yes
V(I) = V(I) - P(L)
No
NMP = 10
NMPI = NMP + 1
M
M
DØ J = 1, NMPI
N
KJ = J - 1
Z = B(I)*KJ/NMP
NØTS = 1 or I = 1
Yes
No
Z < X
No
Z = B(I) - XHB
X(I) = B(I) - HB
J = NMPI
Yes
TM(I) = -(-XM(I) + V(I)*Z)-
ALØAD(I)*Z**2/2
I > 1 or NBBG < 3
Yes
No
NBBG = 4 and Z > G
Yes
TM(1) = TM(1) - P(4)*(Z - G)
No
Z > F
Yes
TM(1) = TM(1) - P(3)*(Z - F)
No
Z < X
Yes
P
No
O


Q
XXM(1, 2) = XM12
XXM(2, 1) = XM21
XXM(2, 3) = XM23
XXM(3, 2) = XM32
XXM(3, 4) = XM34
XXM(4, 3) = XM43
XXM(4, 5) = XM45
XXM(5, 4) = XM54
ISS = NDECK + 2
ISST = ISS + NJ - 2
DØ I = ISS, ISST
Print I
DØ J = 1, NMP1
KJ = J - 1
Z = XLF(I)*KJ/NMP
K6 = I - 6
K5 = I - 5
V(I) = -((XXM(K5, K6) - XXM(K6, K5))/
XLF(I) + (WDL(I)/3 + WD(I)/2)*XLF(I))
TM(I) = -(-XXM(K6, K5) + V(I)*Z) - (WD(I) +
WDL(I))*Z**2/2 + WDL(I)*Z**3/6/XLF(I)
TMM = TM(I)
Print TMM, Z
KJ = 0
or
KJ = NMP
Yes
No
|TMM|
< XMT(I)
Yes
No
XMT(I)
= |TMM|
R
O
TM(1) = TM(1) - P(2)
*(Z - X)
Bulkhead Bay
I > 1
Yes
No
TM(I) = TM(I) -
P(L)*(Z - X)
P
TMM = TM(I)
Print TMM, Z
KJ = 0
Yes
No
JH = I + 1
|TMM| ≤
XMT(JH)
Yes
No
XMT(JH)
= |TMM|
CØNTINUE
N
L
Q


R
NJ2 = NJ - 1
Print "UPWARD JØINT MØMENTS"
DØ I = 1, NJ2
Print XXM(I, J) J = 2, NJ
Print "DØWNWARD JØINT MØMENTS"
DØ I = 2, NJ
Print XXM(I, J) J = 1, NJ2
S
S
Side Shell
DØ I = 1, ISST
XMMT(I) ≤ XMT(I)
Yes
No
XMMT(I) = XMT(I)
XMMT(1) = XMMT(2)
Print "MAXIMUM MØMENTS ØN TRANSVERSE ELEMENTS"
Print XMMT(I), I = 1, 10
T
T
Deflection Calculation
DØ I = 2, NDECK
J = NBBG - 1 + I
L = I + 1
V(I) = - ALØAD(I)*
(B(I) - XHB)
PD(J) = (-(- XM(I)*X**2/2
+ V(I)*X**3/6) - ALØAD(I)
*X**4/24)/E/SMAF(L)
- T(I)*X
NBBG < 3
No
Yes
PD(3) = (-(-XM(1)*F**2/
2 + V(1)*F**3/6) -
ALØAD(1)*F**4/24)
/E/SMAF(L) - T(1)*F
No
NBBG = 4
Yes
PD(4) = (-(- XM(1)*G**2/2 +
V(1)*G**3/6) - ALØAD(1)*
G**4/24)/E/SMAF(L) - T(1)*G
U
Yes
NØC = 2
No
V


U
Influence Coefficients
KK = 1
No
W
Yes
*P(1)
XM(1)*B**2/2 + V(1)*B**3/6))
1) - T(1)*B
XM(1)*X**2/2 + V(1)*X**3/6))
1) - T(1)*X
No
PD(3) = (-(- XM(1)*F **2/2 + V(1)*F**3/6)) /E/SMAF(1) - T(1)*F
No
NBBG = 4
Yes
PD(4) = (-(- XM(1)* G**2/2 + V(1)* G**3/6))/E/ SMAF(1) - T(1)*F
0.0
URN
W
Deflection Under Load
V(KK) = - P(KK)
I = KK - NBBG + 1
KK < 2
No
L = I + 1
PD(KK) = (-(- XM(J)*X**2/2 + V(KK)*X**3/6))/E/SMAF(L) - T(I)*X
Yes
PD(1) = (-(- XM(1)*B**2/2 + V(2) *B**3/6) - P(2)*HB**3/6) /E/SMAF(1) - T(1)*B
PD(2) = (-(- XM(1)*X**2/2 + V(2) *X**3/6))/E/SMAF(1) - T(1)*X
NBBG < 3 or KK ≠ 3
No
PD(2) = PD(2) + (- P(3)*(X - F) **3/6 - V(3)*X**3/6)/E/SMAF(1)
PD(1) = PD(1) + (- P(3)*(B - F) **3/6 - V(3)*B**3/6/E/SMAF(1)
Yes
NBBG > 3 and KK = 3
Yes
PD(3) = (-(- XM(1)*F**2/2 + V(3) *F**3/6))/E/SMAF(1) - T)1)*F
No
X


X
NBBG > 4
and
KK = 3
Yes
No
PD(4) = (-(- XM(1)*G**2/2 + V(3)
*G**3/6))/E/SMAF(1) - T(1)*G
NBBG > 3
and
KK = 2
Yes
No
PD(3) = (-(- XM(1)*F**2/2 +
V(2)*F**3/6))/E/SMAF(1) - T(1)*F
NBBG > 4
and
KK < NBBG
Yes
No
PD(4) = (-(- XM(1)*G**2/2 + V(2)
*G**3/6))/E/SMAF(1) - T(1)*G
NBBG = 4
and
KK = 3
Yes
No
PD(4) = PD(4) - V(3)*G**3/
6/E/SMAF(1)
Y


Y
NBBG < 4 or KK ≠ 4
No
Yes
PD(1) = PD(1) + (- P(4)*(B - G) **3/6 - V(4)*B**3/6)) /E/SMAF(1)
PD(2) = PD(2) + (-P(4)*(X - G) **3/6 - V(4)*X**3/6)) /E/SMAF(1)
PD(3) = (XM(1)*F**2/2)/E /SMAF(1) - T(1)*F
PD(3) = PD(3) + (- P(4)*(F - G) **3/6 - V(4)*F**3/6)) /E/SMAF(1)
NBBG > 3 and KK > NBBG
PD(3) = (-(-XM(1)*F**2/2)) /E/SMAF(1) - T(1)*F
NBBG = 4 and KK > 4
PD(4) = (-(-XM(1)*G**2/2)) /E/SMAF(1) - T(1)*G
V(KK) = 0.0
V
NØC = 1
Yes
Z
No
RETURN
Z
Load Deflections
V(1) = - ALØAD(1)*B
PD(1) = (-(- XM(1)*B**2/2 + V(1)*B**3/6) - ALØAD(1)*B**4/24)/E/SMAF(1) - T(1)*B
PD(2) = (-(- XM(1)*X**2/2 + V(1)*X**3/6) - ALØAD(1)*X**4/24)/E/SMAF(1) - T(1)*X
NBBG < 3
No
Yes
PD(3) = (-(- XM(1)*F**2/2 + V(1)*F**3/6) - ALØAD(1)*F**4/ 24)/E/SMAF(1) - T(1)*F
NBBG = 4
No
Yes
PD(4) = (-(- XM(1)*G**2/2 + V(1)*G**3/6) -ALØAD(1) *G**4/24)/E/SMAF(1) - T(1)*G
Write XM(I), I = 1, NJ
T(I), I = 1, NJ
RETURN

7. Subroutine FSHAPE

a) Abstract:

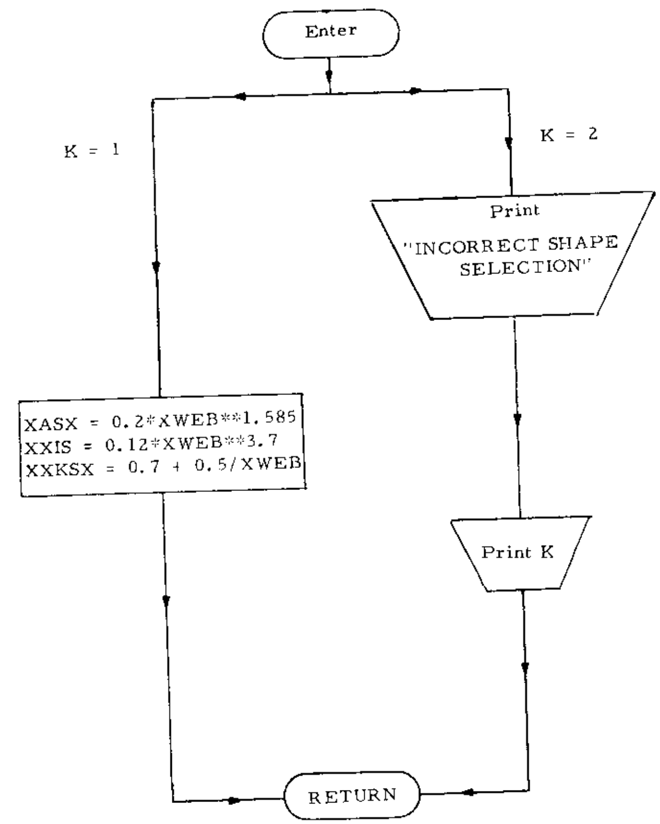
This subroutine is called from subroutine SHAPES and is used to calculate the scantlings of shapes by empirical formulae given in Appendix J.

b) Terms specific to this subroutine:

FORTRAN Term	Definition
K	Flag to select type of section (1 in this study)
XASX	Cross-sectional area of shape
XWEB	Web height of section
XXIS	Second central moment of area of shape
XXKSX	Toe - to - centroid distance

```
   SUBROUTINE  FSHAPE(XWEB,XXIS,XASX,XXKSX,K)
   GO TO (1,2)  K
 1 CONTINUE
   XASX= 0,2*XWEB**1,585
    XXIS=  ,12  *XWEB**3,7
   XXKSX=  0,7+ 0,5/XWEB
   GO TO 10
 2 CONTINUE
   PRINT 10,K
10 FORMAT(/,*  INCORRECT SHAPE SELECTION K= *, I2 ,/)
10 CONTINUE
   RETURN
   END
```

Subroutine FSHAPE (XWEB, XXIS, XASX, XXKSX, K)



8. Subroutine GEOM

a) Abstract:

This subroutine is called by program RATIONAL. It is used to calculate:

a) The vertical coordinate of the centroid of the cross sectional area of each item of plating.
b) The vertical coordinate of the centroid of the cross sectional area of each longitudinal stiffener (shape or longitudinal girder).
c) The external loading on the shell.
d) The internal loading on the decks and inner bottom.

b) Description

The following observations are pertinent:

• Each item of vertical plating is considered to extend from deck to deck, so that its centroid is located at midheight between decks. This artificiality is pro-tempore and can be later removed when the seams are located.

• The external hydrodynamic loading and the internal loads are treated as if statically applied.

• The pressure head on the side shell is calculated to correspond to the midheight between decks.

• The design head on the inner bottom is assumed to correspond to the second deck.

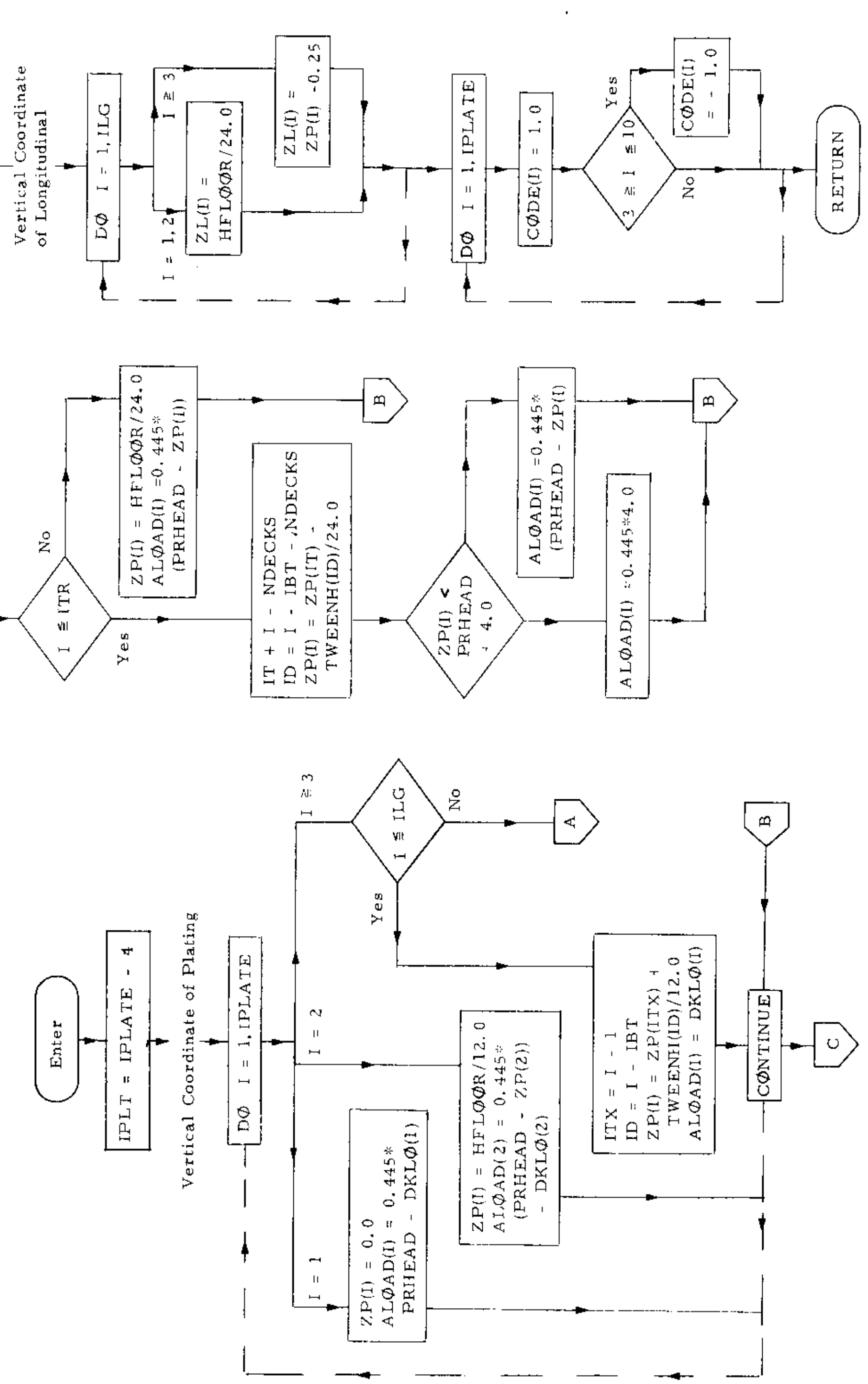
• For simplicity the centroid of a deck longitudinal is assumed to be 0.5 ft. below the deck plating.

• The initial value of the secondary stress intensity is introduced in this routine.

• In the subroutine, each item of plating is flagged according to whether it is to be treated as having fixed or simply-supported boundaries.

```
      SUBROUTINE [illegible](IPLATE,ILG,ITR,NDECKS,PRHEAD,HFLOOR,ZP,ZL,CODE,
     1 ALOAD,TWEENH)
      COMMON/DK/ DKLO(10)
      COMMON /F/ IBT,F
      DIMENSION CODE(17),ALOAD(17),TWEENH(5) ,ZP(17),ZL(7)
C
      IPLT = IPLATE - 4
      DO 1 I = 1,IPLT
    1 CONTINUE
C
C     VERTICAL COORDINATES OF PLATING
C     LOADS
      DO  2  I =1,IPLATE
      GO TO (3,4,5)I
    3 ZP(I)=0.0
      ALOAD(1)= 0.445* (PRHEAD) -DKLO(1)
      GO TO 2
    4 ZP(I) = HFLOOR/ 12.0
      ALOAD(2)=0.445*(PRHEAD-ZP(2))     -DKLO(2)
      GO TO 2
  5   IF(I.LE.ILG) GO TO 6
      IF(I.LE.ITR)GO TO 7
      ZP(I) =  HFLOOR/24.0
      ALOAD(I) = 0.445 *(PRHEAD -ZP(I))
      GO TO 2
    7 IT = I - NDECKS
      ID = I - IBT-NDECKS
      ZP(I) = ZP(IT)-TWEENH(ID)/24.0
      IF(ZP(I)-PRHEAD+4.0)11,17,17
   11 ALOAD(I) = 0.445 *(PRHEAD -ZP(I))
      GO TO 2
   17 ALOAD(I)=0.445*4.0
      GO TO  2
    6 ITX = I-1
      ID = I - IBT
      ZP(I) = ZP(ITX) + TWEENH (ID)/12.0
      ALOAD(I) = DKLO(I)
    2 CONTINUE
C
C     VERTICAL COORDINATE OF LONGITUDINALS
      DO  8  I = 1,ILG
      GO TO (9,9,10)I
    9 ZL(I) = HFLOOR/24.0
      GO TO 8
   10 ZL(I) = ZP(I)-0.25
    8 CONTINUE
C FIXED=1.0   HINGED=-1.0
      DO  23  I =1,IPLATE
      CODE(I)= 1.0
   23 IF(I.GE.3.AND.I.LE.10)  CODE(I)=-1.0
      RETURN
      END
```

Subroutine GEØM (IPLATE, ILG, ITR, NDECKS, PRHEAD, HFLØØR, ZP, ZL, CØDE, ALØAD, TWEENH)



9. Subroutine GRILLAGE

a) Abstract:

This subroutine is called from program TRANSHIP and is the executive routine for the grillage calculation. It calls subroutine FRAME, EV, and RTPLSUB.

b) Terms specific to this subroutine:

FORTRAN Term	Definition
D	External head at innerbottom
HB	Half-breadth of hatchway
LE	Code = 1,2 indicates British units = 3,4 indicates Metric units
NS	Number of stiffeners
W	Deflection due to external load only

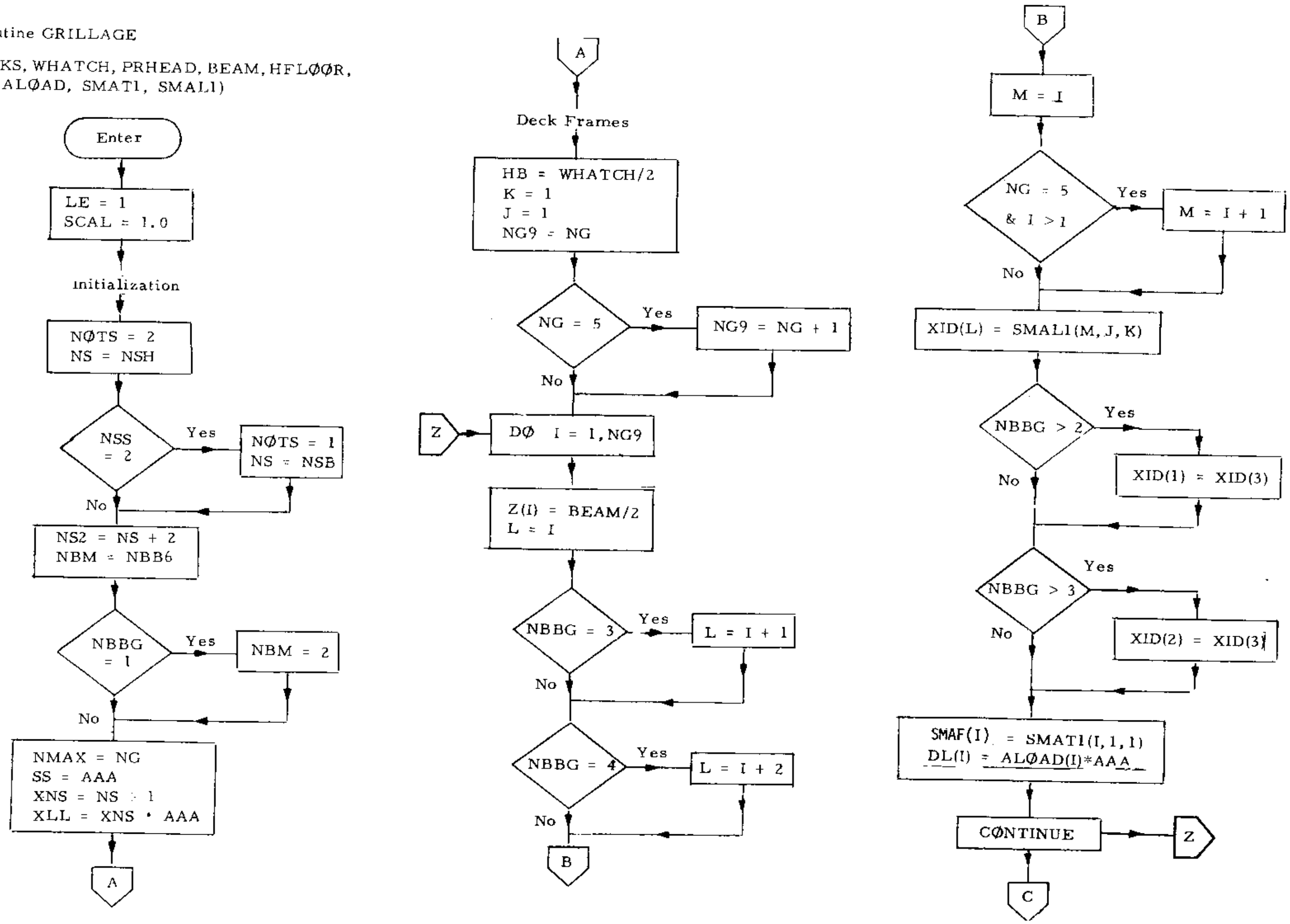
```
      SUBROUTINE GRILLAGE        (NDECKS,WHATCH,WTH  T,BEAM,HFLOOR,AAA,
     1 ALOAD,SMAT1,SMAL1   )
      DIMENSION PPP(9),ROOTI(9)
      COMMON  ALFA(15,15),AX( 9,9 ),CC(10),CN,DEFL( 9),GP,H,JJ,K,NG,NS,
     1PD(15,40), ROOT(9),S(15),SS,V(15),VV(15),W(15),X,XID(15),XII(40),
     2XL,XLL,XML(15),XMO(15),XX(40),RR,SCAL,AFAC,NNN
      COMMON/GR/ B1(9,9),B2(9,9),B3(9,9),XL1(9),XL2(9),Z(12),DL(17),
     1  ITEST,IPILL,NOOS,NSH,NSB,NSS,D,WI,HB,Q,NOTS,NM,NBBG,NJ,XFL(15),
     2  SMAF(15)
      COMMON/LABEL/ TRANLAB(10),PLATLAB(15),LONGLAB(6)
      COMMON/CCC/CONV(225)
      COMMON /E/ IBT,E
       COMMON/H/ XMMT(17)
      DIMENSION ALOAD(17),SMAL1(17,2,3),SMAT1(12,2,5)
        DIMENSION BN(9,9)
       LE = 1
       SCAL = 1,0
C
C     INITIALIZATION
C
      NOTS=2
      NS=NSH
      IF(NSS .EQ.2)  NOTS=1
      IF(NSS .EQ.2)  NS=NSB
       NS2=NS+2
      NBM=NBBG
      IF(NBBG.EQ.1) NBM=2
      NMAX=NG
      SS=AAA
       XNS = NS + 1
       XLL = XNS * AAA
      IF(ITEST.EQ.1.) GO TO  9933
C   DECK FRAMES
      HB=WHATCH/2.
      K=1
      J=1
       NG9=NG
      IF(NG,EQ,5) NG9=NG+1
      DO 4554 I=1,NG9
      Z(I)=BEAM/2.
      L=I
      IF(NBBG.EQ,3)  L=I+1
      IF(NBBG.EQ,4)  L=I+2
      M=I
      IF(NG,EQ.5,AND,I,GT,1)  M=I+1
      XID(L)=SMAL1(M,J,K)
      IF(NBBG.GT,2)  XID(1)=XID(3)
      IF(NBBG.GT,3)  XID(2)=XID(3)
       SMAF(I)=SMAT1(I,1,1)
      DL(I)=ALOAD(I)*AAA
 4554 CONTINUE
      Q=DL(1)
      DL(2)=DL(3)
```

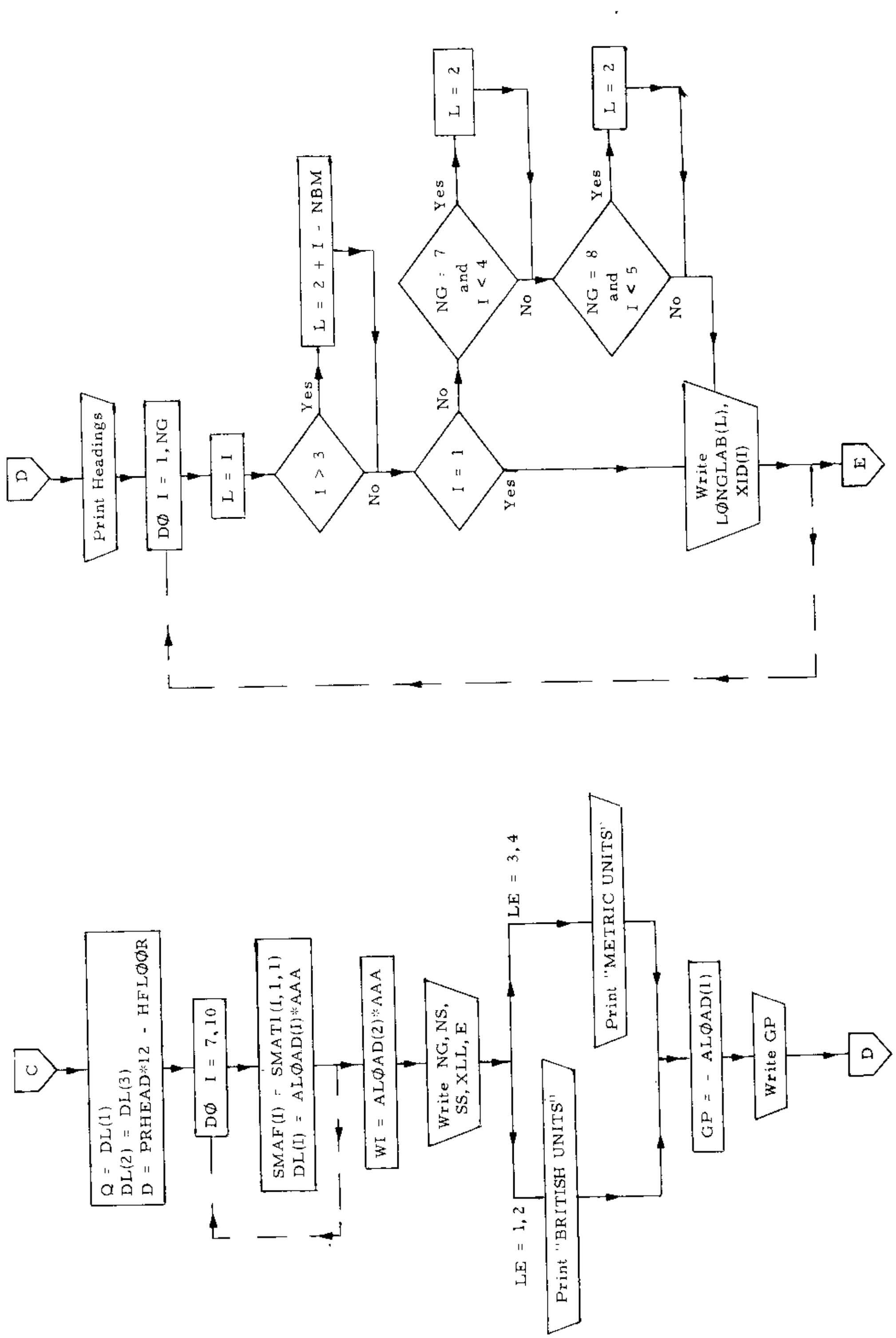
```
       D=PRHEAD*12,-HFLOOR
       DO 4555 I=7,10
       SMAF(I)=SMAT1(I,1,1)
 4555 DL(I)=ALOAD(I)*AAA
       WI=ALOAD(2)*AAA
 9933 CONTINUE
       WRITE OUTPUT TAPE 6,  143
       WRITE OUTPUT TAPE 6,  110,NG
       WRITE OUTPUT TAPE 6,  111,NS
       WRITE OUTPUT TAPE 6, 9112,SS
       WRITE OUTPUT TAPE 6,  11 ,XLL
       WRITE OUTPUT TAPE 6,  115,E
         GO TO (1,1,3,3),LE
    1  WRITE OUTPUT TAPE 6,  119
      GO TO 15
    3  WRITE OUTPUT TAPE 6,  121
   15 GP =-ALOAD(1)
       WRITE OUTPUT TAPE 6,  127,GP
      PRINT 1111
 1111 FORMAT( * LONGITUDINAL GIRDERS *,/,* SECOND MOMENT OF AREA *,/)
      DO 24 I=1,NG
      L=I
      IF(I.GT.3) L=2+I-NBM
      IF(I.EQ.1) GO TO 24
      IF(NG.EQ.7,AND,I,LT,4)   L=2
      IF(NG.EQ.8,AND,I. LT,5)  L=2
   24  WRITE OUTPUT TAPE 6, 9132,LONGLAB(L),         XID(I)
 9132 FORMAT(*  *,A8 ,4E13,3)
      PRINT 5467
 5467 FORMAT( /,* TRANSERVE MEMBERS *,/,* TRANS,   LOAD            LENGTH
     1          SECOND MOMENT*,/)
      DO 9108 I=1,10
 9108 PRINT 9132,TRANLAB(I),DL(I),XFL(I),SMAF(I)
      PRINT 103,WI
C
C     FRAME SOLUTION
C
      NOC=1
      NM=NMAX
      NM1X=NMAX
      IF(NBBG.EQ,1)  NM1X=NMAX+1
      N1=NM1X
      DO 421 N=1,NM1X
  421 PPP(N)=0,0
      CALL FRAME(SMAF,NJ,XFL,DL,WI,D,Z,HB,PPP,Q,E,N1,NOTS,NOC,W,XK,NBM)
      DO 9351 I=1,NM1X
      J=I
      IF(NBBG.EQ.1,AND,I,GT,1)  J=I+1
 9351 W(I)=W(J)
      WRITE(3,4040) ( W(I),I=1,NMAX)
 4040 FORMAT (1H0,     13H LOAD DEFLECT ,  12H              /9E13,5)
      NOC=2
```

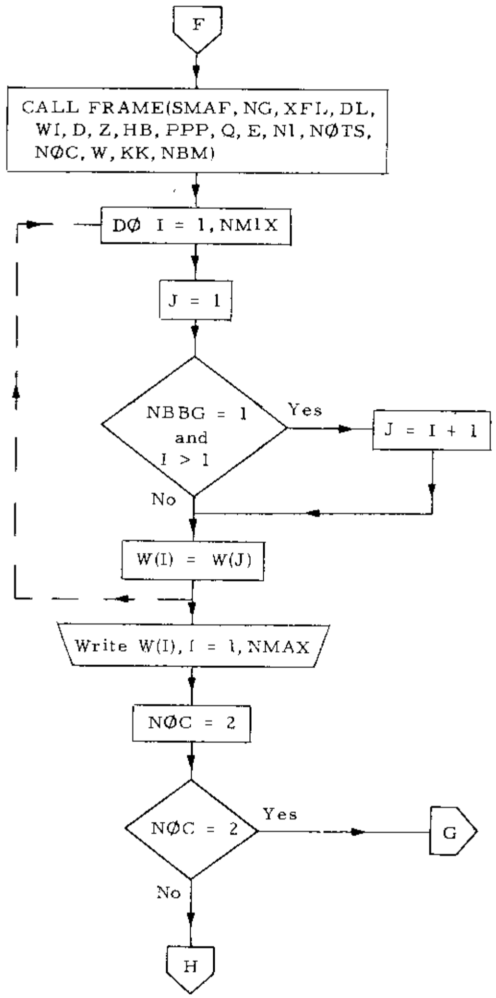
```
      IF(NOC-2) 302,301,302
  301 DO 211 MM=1,NM1X
  211 PPP(MM)=0.0
      DO 201 KK=1,NM1X
      K1=KK
      IF( KK.GT.1.AND.NBBG.EQ.1)  K1=KK+1
      PPP(K1)=1.0
      KJ=K1-1
      IF(KJ)302,302,202
  202 PPP(KJ)=0.0
  302 CONTINUE
      CALL FRAME(SHAF,NJ,XFL,DL,0,D,Z,HB,PPP,G,E,M1,NOTS,NOC,DEFL,KK,NBM
     1 )
      DO 3020 J1=1,NMAX
      JJ=J1
      IF( NBBG.EQ.1.AND.J1.GT.1) JJ=J1+1
      ALFA(J1,KK)=-DEFL(JJ)
 8899 FORMAT(F13.5)
 3020 CONTINUE
  201 CONTINUE
      IF(ITEST.EQ.0)  GO TO 8900
      IF(NOTS.EQ.2)  GO TO 8800
      ALFA(1,3)=-0.0003541
      ALFA(1,4)=0.0001556
      ALFA(2,3)=-0.0002929
       ALFA(2,4)= 0.0001287
      ALFA(3,1)=-.00017710
      ALFA(3,2)=-0.0002929
      ALFA(3,3)=0.03117
      ALFA(3,4)=-0.001877
      ALFA(4,1)=0.00007702
      ALFA(4,2)=0.0001287
      ALFA(4,3)=-0.001877
      ALFA(4,4)=0.02848
      GO TO 8900
 8800 ALFA(3,3)=0.056820
      ALFA(4,4)=0.05629
 8900 CONTINUE
 4100  WRITE OUTPUT TAPE 6,  123
      DO 26 IK=1,NG
   26 PRINT 107,(ALFA(IK,JZ),JZ=1,NG)
        NNN=0
  401 CONTINUE
          NNN=1
 4005 DO 14 II=1,NG
      DO 14 JJ=1,NG
   14 AX(II,JJ)=E*XID(JJ)*SS+ALFA(II,JJ)
      PRINT 124
      DO 27 L=1,NG
   27  WRITE OUTPUT TAPE 6,  107,(AX(L,N),N=1,NG)
C
C     ROOT SOLUTION
C
        CALL EV(AX,NG,BN,CC)
            CN=CC(NG+1)
        NGO=NG+1
       CALL RTPLSUR(NG,CC,  ROOT,ROOTI,CONV,IER)
        PRINT 9139,(CC(NGC),NGC=1,NGO)
 4907 CONTINUE
  939 FORMAT(F10.4)
  103 FORMAT(E16.8)
  107 FORMAT(9E12.4)
  110 FORMAT(18H NUMBER OF GIRDERS11X,I3)
  111 FORMAT(21H NUMBER OF STIFFENERS8X,I3)
 9112 FORMAT(22H SPACING OF STIFFENERS4X,F15.3)
  114 FORMAT(18H LENGTH OF GIRDERS8X,F15.3)
  118 FORMAT(22H MODULUS OF ELASTICITY1X,F18.3)
  119 FORMAT (/34H BRITISH UNITS(POUNDS AND INCHES)  )
  120 FORMAT(40H METRIC UNITS(KILOGRAMS AND CENTIMETERS))
  121 FORMAT(35H METRIC UNITS(TONS AND CENTIMETRES))
  122 FORMAT(/31H BRITISH UNITS(TONS AND INCHES))
  123 FORMAT(//15H VEDELER FIXITY)
  124 FORMAT(//9H A MATRIX)
  127 FORMAT(23H GRILLAGE PRESSURE LOAD3X,F15.3//)
  143 FORMAT (1H1,10X,9H RESTORE )
 9139 FORMAT (1H0,10X,13H EIGEN. EQU. //  5X, 7H REAL  9E13.5)
  318 CONTINUE
 9999 CONTINUE
      RETURN
      END
```

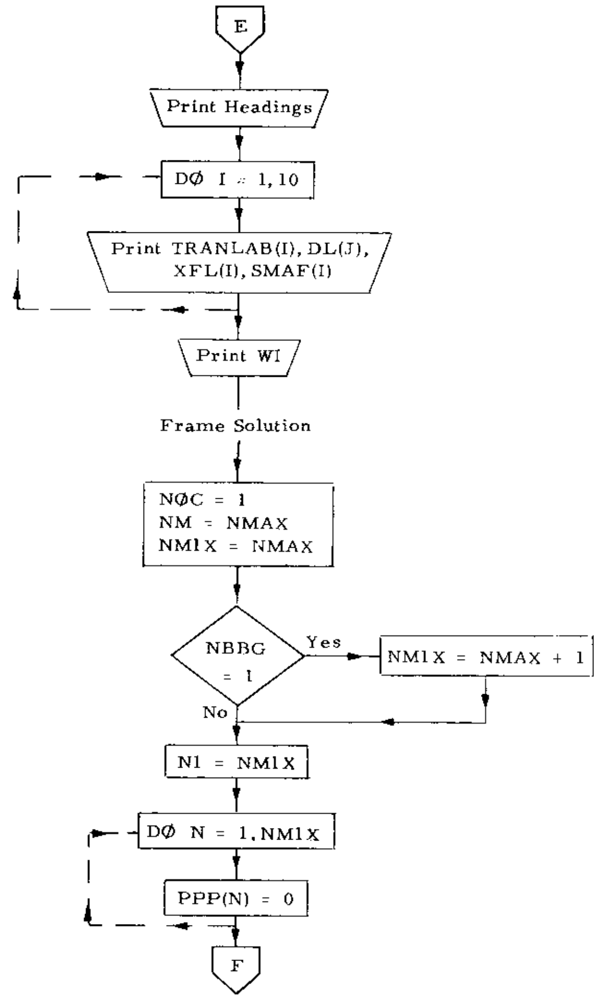
Subroutine GRILLAGE

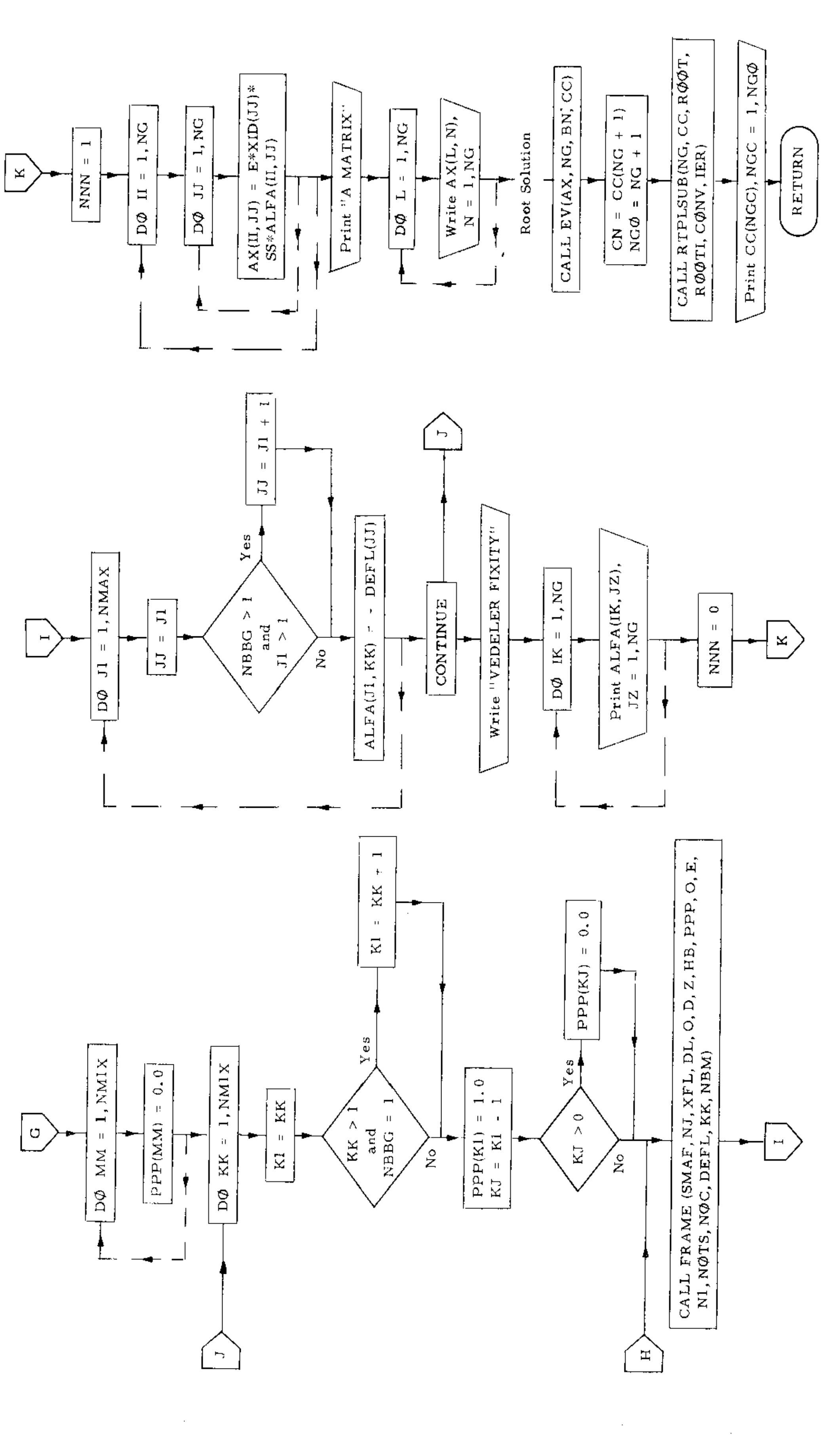
(NDECKS, WHATCH, PRHEAD, BEAM, HFLØØR, AAA, ALØAD, SMAT1, SMAL1)




C
Q = DL(1)
DL(2) = DL(3)
D = PRHEAD*12 - HFLØØR
DØ I = 7,10
SMAF(I) = SMAT1(I,1,1)
DL(I) = ALØAD(I)*AAA
WI = ALØAD(2)*AAA
Write NG,NS, SS,XLL,E
LE = 1,2
Print "BRITISH UNITS"
LE = 3,4
Print "METRIC UNITS"
GP = - ALØAD(1)
Write GP
D
D
Print Headings
DØ I = 1,NG
L = I
I > 3
Yes
No
L = 2 + I - NBM
I = 1
Yes
No
NG = 7 and I < 4
Yes
No
L = 2
NG = 8 and I < 5
Yes
No
L = 2
Write LØNGLAB(L), XID(I)
E


F
CALL FRAME(SMAF, NG, XFL, DL, WI, D, Z, HB, PPP, Q, E, N1, NØTS, NØC, W, KK, NBM)
DØ I = 1, NM1X
J = 1
NBBG = 1 and I > 1
Yes
J = I + 1
No
W(I) = W(J)
Write W(I), I = 1, NMAX
NØC = 2
NØC = 2
Yes
G
No
H


E
Print Headings
DØ I = 1, 10
Print TRANLAB(I), DL(J), XFL(I), SMAF(I)
Print WI
Frame Solution
NØC = 1
NM = NMAX
NM1X = NMAX
NBBG = 1
Yes
NM1X = NMAX + 1
No
N1 = NM1X
DØ N = 1, NM1X
PPP(N) = 0
F


G
DØ MM = 1, NMIX
PPP(MM) = 0.0
DØ KK = 1, NMIX
K1 = KK
KK > 1 and NBBG = 1
Yes
No
K1 = KK + 1
PPP(K1) = 1.0
KJ = K1 - 1
KJ > 0
Yes
No
PPP(KJ) = 0.0
CALL FRAME (SMAF, NJ, XFL, DL, O, D, Z, HB, PPP, O, E, N1, NØTS, NØC, DEFL, KK, NBM)
I
J
H
I
DØ J1 = 1, NMAX
JJ = J1
NBBG > 1 and J1 > 1
Yes
No
JJ = J1 + 1
ALFA(J1, KK) = - DEFL(JJ)
CONTINUE
J
Write "VEDELER FIXITY"
DØ IK = 1, NG
Print ALFA(IK, JZ), JZ = 1, NG
NNN = 0
K
K
NNN = 1
DØ II = 1, NG
DØ JJ = 1, NG
AX(II, JJ) = E*XID(JJ)* SS*ALFA(II, JJ)
Print "A MATRIX"
DØ L = 1, NG
Write AX(L, N), N = 1, NG
Root Solution
CALL EV(AX, NG, BN, CC)
CN = CC(NG + 1)
NGØ = NG + 1
CALL RTPLSUB(NG, CC, RØØT, RØØTI, CØNV, IER)
Print CC(NGC), NGC = 1, NGØ
RETURN

10. Subroutine INPUT

a) Abstract:

Subroutine INPUT is called from TRANSHIP. It reads in the input data and calculates initial stress conditions. Subroutine GEOM is called.

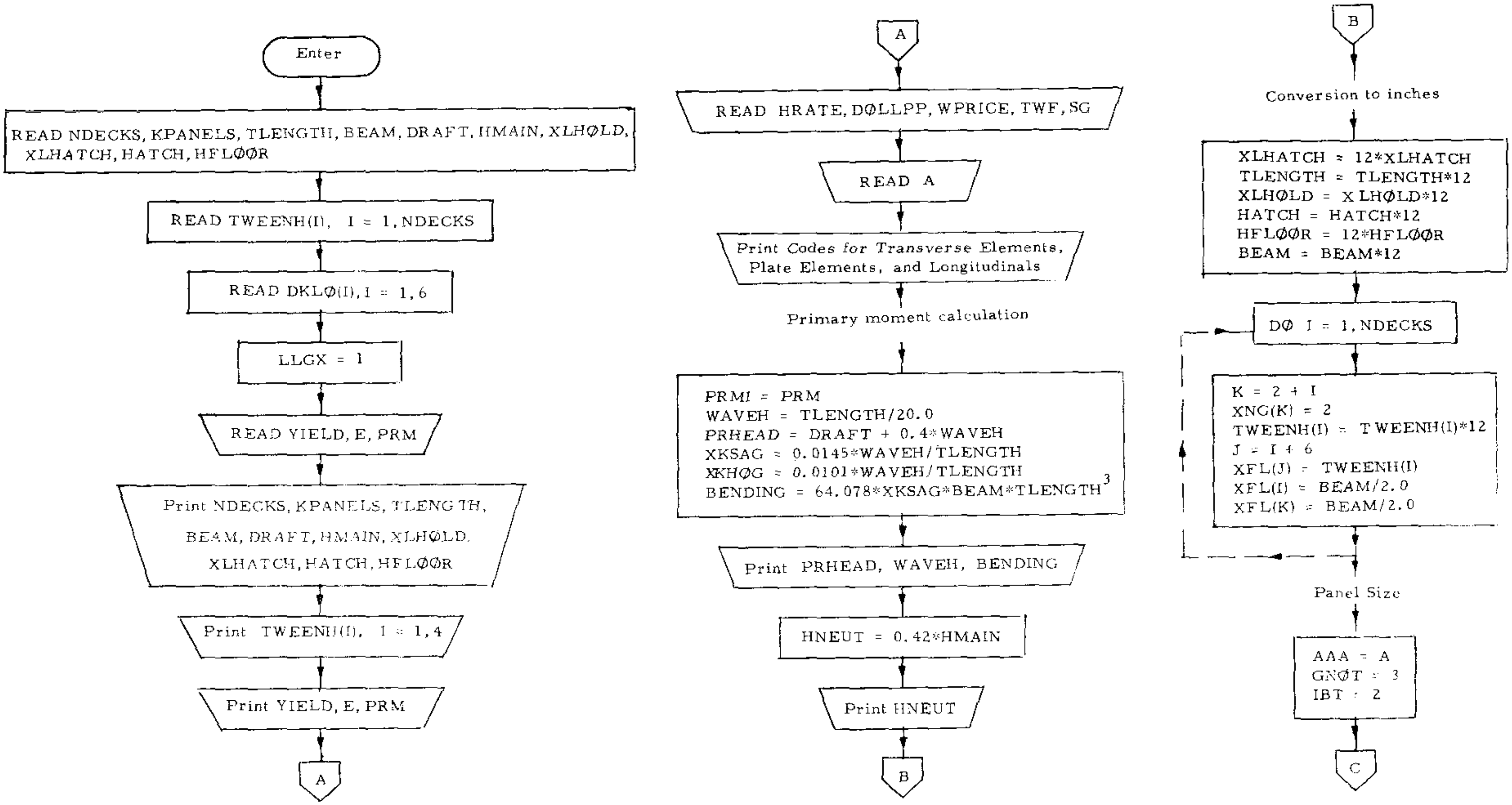
b) Terms specific to this subroutine:

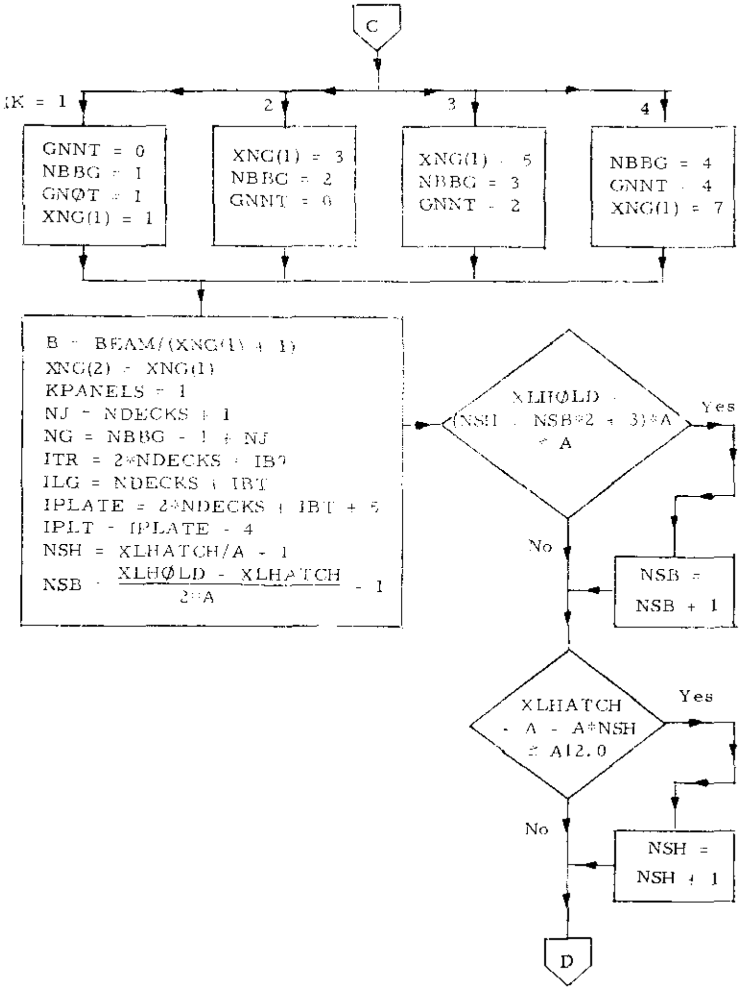
FORTRAN Term	Defintion
BENDING	Ship bending moment for sag
XMOME	Bending moment

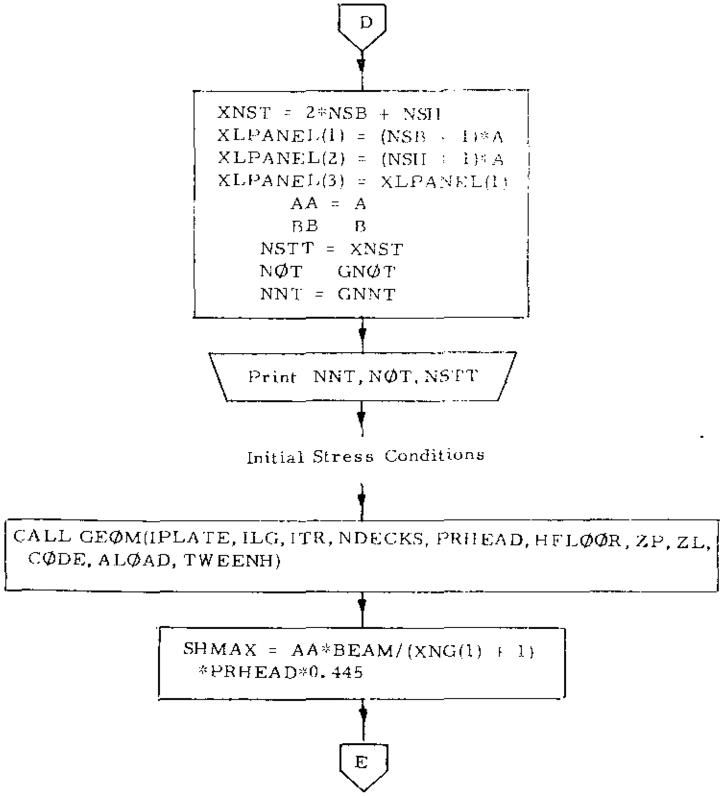
```
SUBROUTINE INPUT([illegible],NBBG, IK ,NJ,XFL,XIII,NSH,NSB,NG)
COMMON/TLEN/  TLENGTH    ,IHOG ,RABI,XKSAG,XKHOG
DIMENSION XIII(40)
COMMON /E/ IBT,E
COMMON/DK/ DKLO(10)
COMMON/LABEL/ TRANLAB(10),PLATLAB(15),LONGLAB(6)
COMMON/SH/   SHMAX
COMMON/COST/ HRATE,DOLLPP,WPRICE,TWF  ,SG
COMMON/LO/FSECLO(12),FSESL(17),FSESL1(17),SMODL2(17,2,3),PSLL(17)
COMMON/TR/ FSECTR(17),FSEST(17),FSEST1(17),SMODT2(17,2,3),SMODT3(
1 17,2,5)
 COMMON/H/ XMMT(17)
DIMENSION  SEST1(17),SESL1(17)
COMMON /B/ A,B,NDECKS,IPLATE,PSL,PRM,HMAIN,HNEUT,TESL,TEST,SESL,
1     SEST,          KPANELS,BEAM,XNG,HATCH,THIKK,THIKK1,EFFBR,EFFB1,
2     EFFW,EFFW1,HFLOOR,TWEENH,THIKX,EFBR,WEB,WEB1,XIPFX,CHI,
3     XTS, AREAT1,SMAL1,SMODL1,WEBL1,SUMAREAP,SUMAREAL,SUMMP,SUMML,
4     AREA,SUMAREA,SUMMOM,WPL,XPANEL,WBAYLP,XNST,NST,ITR,
5     WTR,BAYTR,SUMWT,SUMWTR,LLGX,  YIELD,WBAYTR,XLABOR,PLCOST,
6     COST,COSTMIN,GNOT,GNNT,KNEUT,SUMSMAP,SUMSMAL,SUMSMA,CODE,
7WAVEH,PRHEAD,DRAFT,ALOAD,XLHOLD,XLPANEL,AAA,BBB,AA,BB,DELTESL,
8     SUMSMA1,ZP,ZL,ILG,AREAL1,ASX,  SMAT1,SMODT1
DIMENSION XFL(15),PPP(7),DL(17)
DIMENSION ZP(17),SESL(17),SEST(17),TESL(17),TEST(17),XNG(7),THIKK(
117),      THIKK1(17,2,5),EFFBR(17),EFFW(17),EFFB1(17,2,5),EFFW1(17
2,2,5),AREA(17,5),TWEENH(5),SMAT1(12,2,5),SMODT1(12,2,5),THIKX(12,2
3),EFBR(12,2),AREAT1(12,2,5),WEB1(12,2,5),SMAL1(17,2,3),SMODL1(17,2
4,3),WEBL1(7,2,3),AREAL1(7,2,3),SUMAREAP(5),SUMAREAL(5),SUMMP(5),SU
5MML(5),SUMAREA(5),SUMMOM(5),WPL(5),XLPANEL(5),SUMWTR(5),WTR(12),ZL
6(7),TIKK1(17,5),CODE(17),ALOAD(17),SUMSMAP(5),SUMSMAL(5),SUMSMA(5)
READ 8100,NDECKS,KPANELS,TLENGTH,BEAM,DRAFT,HMAIN,XLHOLD,XLHATCH,
1 HATCH,HFLOOR
READ 8101,(TWEENH(I),I = 1,NDECKS)
READ 8101, (DKLO(I),I=1,6)
LLGX=1
READ 8102, YIELD, E, PRM
PRINT8103,NDECKS,KPANELS,TLENGTH,BEAM,DRAFT,HMAIN,XLHOLD,XLHATCH,
1 HATCH,HFLOOR
PRINT 8104, TWEENH(1), [illegible],TWEENH(3),TWEENH(4)
PRINT 8105, YIELD, E, PRM
READ 8101,HRATE,DOLLPP,WPRICE,TWF,SG
READ 9219,A
     PRINT 9153
     PRINT 9154
     PRINT 9155

      PRIMARY MOMENT CALCULATION
PRMI=PRM
WAVEH = TLENGTH/ 20.0
PRHEAD = DRAFT + 0.4*WAVEH
XKSAG  = 0.0145 *WAVEH / TLENGTH
XKHOG  = 0.0101 *WAVEH / TLENGTH
BENDING = 64.078 * XKSAG * BEAM * TLENGTH ** 3
PRINT 412,PRHEAD,WAVEH,BENDING
HNEUT  = 0.42* HMAIN
PRINT 8153, HNEUT
C
C     CONVERSION TO INCHES
      XLHATCH =12*XLHATCH
      TLENGTH =TLENGTH*12.0
       XLHOLD=XLHOLD*12.
      HATCH=HATCH*12.0
      HFLOOR = 12.0 * HFLOOR
      BEAM =BEAM*12.0
      DO 505 I = 1,NDECKS
      K=2+1
      XNG(K)=2.
      TWEENH(I) = TWEENH(I) * 12.0
      J=I+6
      XFL(J)=TWEENH(I)
       XFL(I)=BEAM/2.0
       XFL(K)=BEAM/2.0
505   CONTINUE

       PANEL SIZE
      AAA=A
      GNOT = 3.0
      IBT=2
      GO TO ( 499,501,502,503) IK
499   GNNT=0.
      NBBG=1
      GNOT=1.
      XNG(1)=1.0
      GO TO 504
501   XNG(1)= 3.0
      NBBG=2
      GNNT=0.
      GO TO 504
502   XNG(1)=5.0
      NBBG=3
      GNNT =2.0
      GO TJ 504
503   NBBG=4
      GNNT=4.
      XNG(1)=7.
504   B=BEAM/(XNG(1)+1.0)
      XNG(2)=XNG(1)
      KPANELS = 1
      NJ=NDECKS+1
      NG=NBBG+1+NJ
      ITR = 2*NDECKS + IBT
      ILG=NDECKS+IBT
      IPLATE=2*NDECKS+IBT+5
      IPLT=IPLATE-4
      NSH=XLHATCH/A -1
      NSB=((XLHOLD-XLHATCH )/2.)/A-1
      IF((XLHOLD-(NSH+NSB*2+3)*A).GE.A)      NSB=NSB+1
      IF((XLHATCH-A-A*NSH).GE.A/2.)     NSH=NSH+1
      XNST=2*NSB+NSH
      XLPANEL(1)=(NSB+1)*A
      XLPANEL(2)=(NSH+1)*A
      XLPANEL(3) = XLPANE_(1)
```

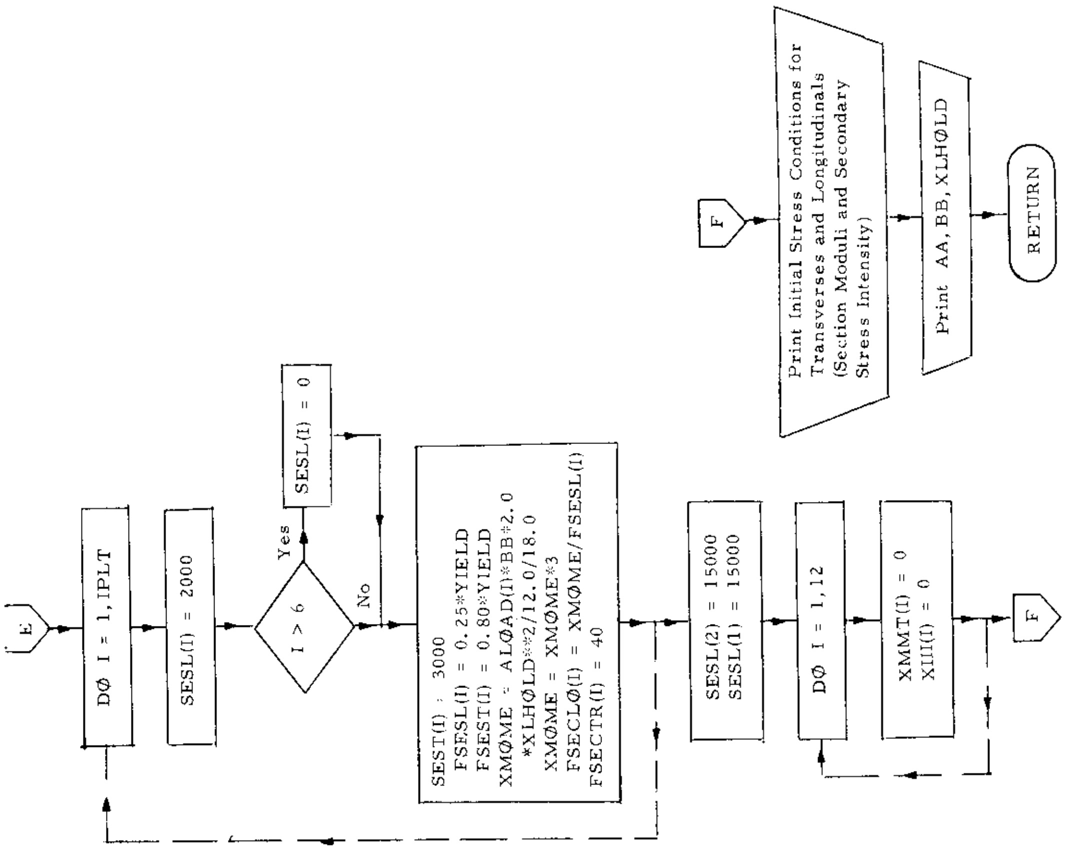
```
      AA=A
      BB = B
      NSTT=XNST
      NOT=GNOT
      NNT=GNNT
      PRINT 3256 ,NNT,NOT,NSIT
C
C     INITIAL STRESS CONDITIONS
      CALL        GEOM(IPLATE,ILG,ITR,NDECKS,PRHEAD,HFLOOR,ZP,ZL,CODE,
     1 ALOAD,TWEENH)
      SHMAX =     AA *BEAM/(XNG(I)+1)* PRHEAD* 0.445
       DO 9687  I=1,IPLT
      SESL(I) =  2000.
      IF(I.GT.6)  SESL(I)=0.
      SEST(I) = 3000.
       FSESL(I)= .25*YIELD
       FSEST(I)= .80*YIELD
      XMOME=ALOAD(I)*   BB*2.0 *        XLHOLD**2/12.0 /18.0
        XMOME=XMOME*3.
        FSECLO(I)=XMOME/  FSESL(I)
      FSECTR(I)=40.
 9687 CONTINUE
      SESL(2)=15000.
      SESL(1)=15000.
       DO 5678  I=1,12
      XMMT(I)=0.
 5678 XIII(I)=0.
      PRINT 1122
 1122 FORMAT( * INITIAL STRESS CONDITIONS *,//,* TRANVERSES *,/,
     1 *                SECTION MODULUS  SECONDARY STRESS*,/)
      DO 4569,I=1,10
 4569 PRINT 8012 ,TRANLAB(I) ,FSECTR(I),SESI(I)
 8012 FORMAT(*  *,A8,4F13.1)
      PRINT 1112
 1112 FORMAT(/* LONGITUDINALS *,/,
     1 *                SECTION MODULUS  SECONDARY STRESS*,/)
       DO 4568 ,I=1,6
 4568 PRINT 8012 ,LONGLAB(I) ,FSECLO(I),SESL(I)
      PRINT 5219
 5219 FORMAT(/,* MAIN PARAMETERS *,//)
      PRINT 510,AA,BB,XLHOLD
      PRINT 2398
 2398 FORMAT( 1H1, //     )
C
C     FORMAT STATEMENTS
C
  412 FORMAT(/  ,* PRESSURE HEAD UNDER WAVE CRESTS *,F15.1,* LB/SQ, INCH
     1* ,/,        * WAVE HEIGHT                    = *,F15.1,* FEET *,/,
     2             * BENDING MOMENT                = *,F15.1,* LB-FEET*,/)
  510 FORMAT(  /,* FRAME SPACING                 =*,F10.2,* INCHES*,/,
     1* SPACING BOTTOM LONGITUDINALS =*,F10.2 ,* INCHES*,/,
     2* HOLD LENGTH                  =*,F10.2,* INCHES*,/)
 3256 FORMAT(* DOUBLE BOTTOM MEMBERS *,/,
     1          * NUMBER OF N.T. LONG. = *,I3/,
     2          * NUMBER OF O.T. LONG. = *,I3/,
     3          * NUMBER OF N.T. FLOORS= *,I3/,
     4          * NUMBER OF O.T. FLOORS=   2 *,/)
 8100 FORMAT(2I2,8F8.2)
 8101 FORMAT ( 7F10.3)
 8102 FORMAT(3F15.2)
 8103 FORMAT(  /,*       NUMBER OF DECKS          =*,I10/
     1          *       NUMBER OF BAYS           =*,I10/
     2          *       SHIP LENGTH              =*,F10.2,* FEET*,/,
     3          *       BEAM                     =*,F10.2,* FEET*,/ ,
     4          *       DRAFT                    =*,F10.2,* FEET*,/  ,
     5          *       DEPTH                    =*,F10.2,* FEET*,/  ,
     6          *       LENGTH OF HOLD           =*,F10.2,* FEET*,/   ,
     7          *       LENGTH OF HATCH          =*,F10.2,* FEET*,/,
     8          *       WIDTH OF HATCH           =*,F10.2,* FEET*,/,
     9          *       HEIGHT OF FLOOR          =*,F10.2,* FEET*,/)
 8104 FORMAT(*    TWEENDECK HEIGHT 4   =*,F10.3,* FEET*,/,
     1      *     TWEENDECK HEIGHT 3   =*,F10.3,* FEET*,/,
     2      *     TWEENDECK HEIGHT 2   =*,F10.3,* FEET*,/,
     3      *     TWEENDECK HEIGHT 1   =*,F10.3,* FEET*,/)
 8153 FORMAT(1H0,* ESTIMATE OF NEUTRAL AXIS =*,F10.2/)
 8105 FORMAT(*     YIELD STRESS                =*,F15.1,* LB/SQ.IN,*,/,
     1*    YOUNGS MODULUS               =*,F15.1,* LB/SQ.IN,*,/,
     2*    PRIMARY STRESS CRITERION =*,F15.1,* LB/SQ.IN,*,/)
 9153 FORMAT(/,* CODE FOR TRANSVERSE ELEMENTS *,/,* 1= O.T. FLOOR *,/,
     1   * 2= N.T. FLOOR *,/,* 3= FOURTH DECK  DECK BEAM *,/,
     2   * 4= THIRD DECK DECK BEAM *,/,* 5= SECOND DECK DECK BEAM*,/,
     3   * 6= MAIN DECK DECK BEAM *,/,* 7= I.B. TO FOURTH DECK FRAME*
     4   ,/,* 8= FOURTH DECK TO THIRD DECK FRAME*,/,* 9= THIRD DECK TO
     5SECOND DECK FRAME*,/,* 10= SECOND DECK TO MAIN DECK FRAME *,/)
 9154 FORMAT (/,* CODE FOR PLATE ELEMENTS *,/,* 1= DOUBLE BOTTOM *,/,
     1   * 2= INNER BOTTOM*,/,* 3= FOURTH DECK *,/,* 4= THIRD DECK*,/
     2   ,* 5= SECOND DECK *,/,* 6 = MAIN DECK *,/,* 7= FIRST SIDE STRA
     3KE*,/,* 8= SECOND SIDE STRAKE *,/,* 9= THIRD SIDE STRAKE*,
     4/,* 10= TOP SIDE STRAKE    *,/,* 11= BILGE *,/,* 12= O.T. FLOOR *
     5,/,* 13= N.T. FLOOR *,/,* 14= O.T. LONGITUDINAL*,/,* 15= N.T. LONG
     6ITUNIAL *//)
 9155 FORMAT( * CODE FOR LONGITUDINALS *,/,* 1= KEEL *,/,* 2 = SIDE GIRD
     1ERS *,/,* 3= FOURTH DECK *,/,* 4= THIRD DECK *,/,* 5 = SECOND DECK
     2          *,/,* 6 = MAIN DECK *,/)
 9219 FORMAT(F10.2)
 9999 CONTINUE
      RETURN
      END
```

Subroutine INPUT(BENDING, NBBG, IK, NI, XFL, XIH, NSH, NSB, NG)




C
K = 1
2
3
4
GNNT = 0
NBBG = 1
GNØT = 1
XNG(1) = 1
XNG(1) = 3
NBBG = 2
GNNT = 0
XNG(1) = 5
NBBG = 3
GNNT = 2
NBBG = 4
GNNT = 4
XNG(1) = 7
B = BEAM/(XNG(1) + 1)
XNG(2) = XNG(1)
KPANELS = 1
NJ = NDECKS + 1
NG = NBBG - 1 + NJ
ITR = 2*NDECKS + IBT
ILG = NDECKS + IBT
IPLATE = 2*NDECKS + IBT + 5
IPLT = IPLATE - 4
NSH = XLHATCH/A - 1
NSB = XLHØLD - XLHATCH / 2*A - 1
XLHØLD - (NSH + NSB*2 + 3)*A = A
Yes
No
NSB = NSB + 1
XLHATCH - A - A*NSH ≥ A12.0
Yes
No
NSH = NSH + 1
D


D
XNST = 2*NSB + NSH
XLPANEL(1) = (NSB + 1)*A
XLPANEL(2) = (NSH + 1)*A
XLPANEL(3) = XLPANEL(1)
AA = A
BB = B
NSTT = XNST
NØT = GNØT
NNT = GNNT
Print NNT, NØT, NSTT
Initial Stress Conditions
CALL GEØM(IPLATE, ILG, ITR, NDECKS, PRHEAD, HFLØØR, ZP, ZL, CØDE, ALØAD, TWEENH)
SHMAX = AA*BEAM/(XNG(1) + 1) *PRHEAD*0.445
E


E
DØ I = 1, IPLT
SESL(I) = 2000
I > 6
Yes
SESL(I) = 0
No
SEST(I) = 3000
FSESL(I) = 0.25*YIELD
FSEST(I) = 0.80*YIELD
XMØME = ALØAD(I)*BB*2.0
*XLHØLD**2/12.0/18.0
XMØME = XMØME*3
FSECLØ(I) = XMØME/FSESL(I)
FSECTR(I) = 40
SESL(2) = 15000
SESL(1) = 15000
DØ I = 1,12
XMMT(I) = 0
XII(I) = 0
F
F
Print Initial Stress Conditions for Transverses and Longitudinals (Section Moduli and Secondary Stress Intensity)
Print AA, BB, XLHØLD
RETURN

11. Subroutine INTERMED

a) Abstract:

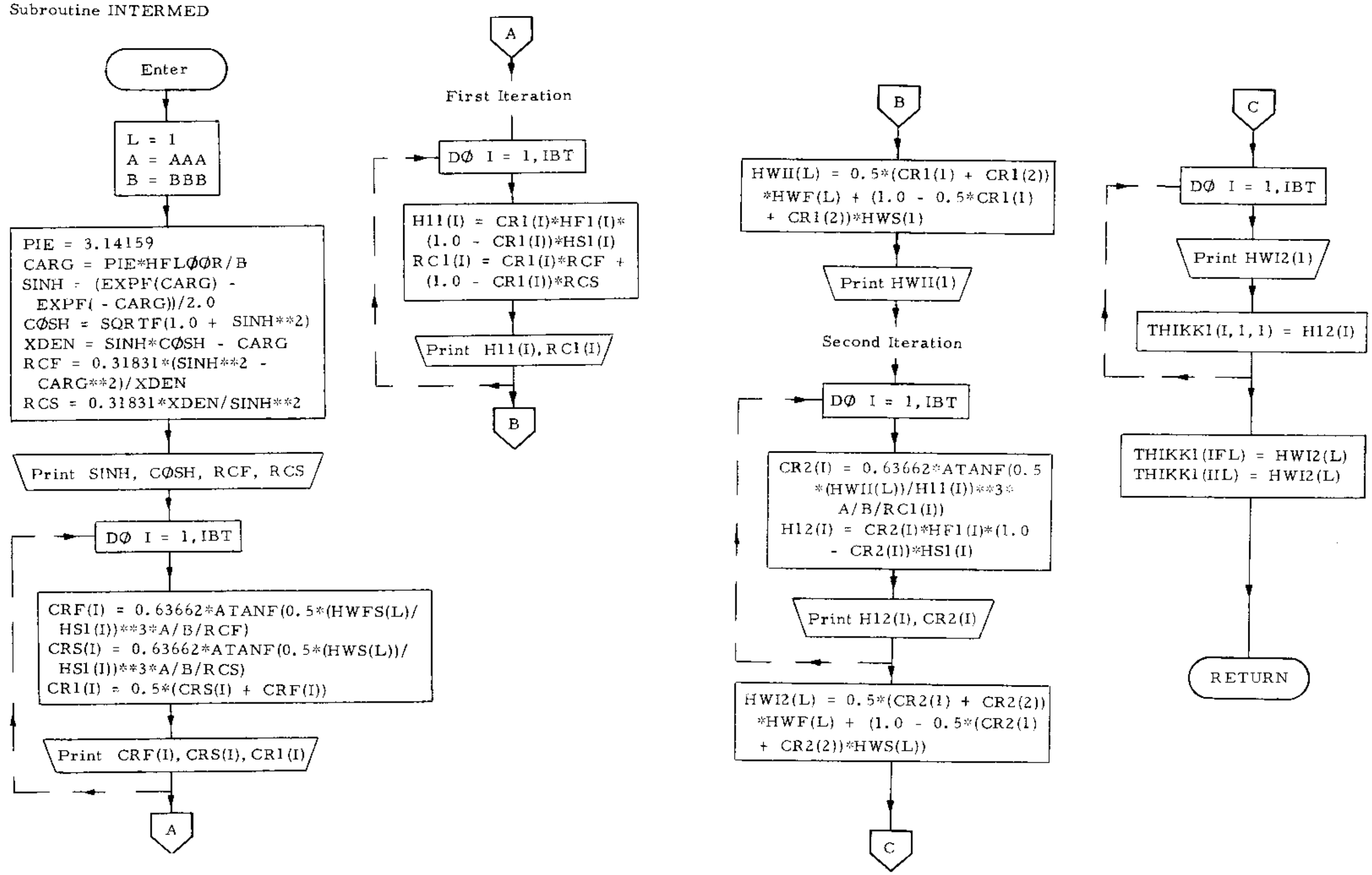
Subroutine INTERMED is called from TRANSHIP. It calculates the plate thicknesses of the double bottom structure for intermediate degrees of fixity. The plate thicknesses for both fixed and simple support are obtained from TRANSHIP. On the basis of these the degree of restraint is calculated in a two step iteration. The plate thicknesses for elastic restraint are then obtained by linear interpolation.

b) Terms specific to this subroutine:

FORTRAN Term	Definition
CRF(I)	Coefficient of restraint for fixed plate (I = 1 indicates bottom shell, I = 2 indicates inner bottom)
CRS(I)	Coefficient of restraint for simply supported plate (I = 1 indicates bottom shell, I = 2 indicates inner bottom)
RC1(I)	Restraint coefficient (r) for intermediate degree of fixity (I = 1 indicates bottom shell, I = 2 indicates inner bottom)
RCF	Restraint coefficient r (fixed case)
RCS	Restraint coefficient r (simply supported)

```
      SUBROUTINE INTERMED
      COMMON/F/ HF1,HS1,HWF,HWS,HWFS,H12,HWI2,IFL,IIL
      COMMON/G/ FIXCODE
      COMMON /B/ A,B,NDECKS,IPLATE,PSL,PRM,HMAIN,HNEUT,TESL,TEST,SESL,
     1     SEST,         KPANELS,BEAM,XNG,HATCH,THIKK,THIKK1,EFFBR,EFFB1,
     2     EFFW,EFFW1,HFLOOR,TWEENH,THIKX,EFBP,WBB,WBB1,XIPEX,CHI,
     3     XIS, AREAT1,SMAL1,SMODL1,WEBL1,SUMAREAP,SUMAREAL,SUMMP,SUMML,
     4     AREA,SUMAREA,SUMMOM,WPL,XPANEL,WRAYLP,XNST,NST,ITR,
     5     WTR,RAYTR,SUMWT,SUMWTR,LLGX,  YIELD,WBAYTR,XLABOR,PLCOST,
     6     COST,COSTMIN,GNOT,GNNT,KNEUT,SUMSMAP,SUMSMAL,SUMSMA,CODE,
     7WAVEH,PHHEAD,DRAFT,ALOAD,XLHOLD,XLPANEL,AAA,BBB,AA,BB,DELTESL,
     8      SUMSMA1,ZP,ZL,ILG,AREAL1,ASX,  SMAT1,SMODT1
      COMMON /E/ IBT,E
      DIMENSION HF1(2),HS1(2),HWF(2),HWS(2),CRF(2),CRS(2),HWFS(2),
     1CR1(2),H11(2),RC1(2),HWI1(2),H12(2),CR2(2),HWI2(2)
      DIMENSION ZP(17),SESL(17),SEST(17),TESL(17),TEST(17),XNG(7),THIKK(
     117),       THIKK1(17,2,5),EFFBR(17),EFFW(17),EFFB1(17,2,5),EFFW1(17
     2,2,5),AREA(17,5),TWEENH(5),SMAT1(12,2,5),SMODT1(12,2,5),THIKX(12,2
     3),EFBR(12,2),AREAT1(12,2,5),WEB1(12,2,5),SMAL1(17,2,3),SMODL1(17,2
     4,3),WEBL1(7,2,3),AREAL1(7,2,3),SUMAREAP(5),SUMAREAL(5),SUMMP(5),SU
     5MML(5),SUMAREA(5),SUMMOM(5),WPL(5),XLPANEL(5),SUMWTR(5),WTR(12),ZL
     6(7),TIKK1(17,5)
      L = 1
      A = AAA
      B = BBB
C     TWO TEMP CARDS
      HFLOOR=50.
      B=214.5
      PIE=3.14159
      CARG = PIE     * HFLOOR/B
      SINH = (EXPF(CARG)-EXPF(-CARG))/2.0
      COSH = SQRTF(1.0+SINH**2)
      XDEN = SINH*COSH - CARG
      RCF = 0.31831*(SINH**2-CARG**2)/XDEN
      RCS = 0.31831* XDEN/SINH**2
      PRINT 6,SINH,COSH,RCF,RCS
    6 FORMAT(1H0,*SINH=*,F14.4,*COSH=*,F14.4,*RCF=*,F14.4,*RCS=*,F14.4)
      DO 1 I = 1,IBT
      CRF (I) = 0.63662*ATANF(0.5*(HWFS(L)/HS1(I))**3*A/B/RCF)
      CRS(I) = 0.63662*ATANF(0.5*(HWS(L)/HS1(I))**3*A/B/RCS)
      CR1(I)  =0.5*(CRS(I) + CRF(I))
      PRINT 7,CRF(I),CRS(I),CR1(I)
    7 FORMAT(* CRF=*,F10.4,*CRS=*,F10.4,*CR1=*,F10.4)
    1 CONTINUE
C     FIRST ITERATION
      DO 2 I=1,IBT
      H11(I) = CR1(I)*HF1(I) + ( 1.0-CR1(I))*HS1(I)
      RC1(I) = CR1(I)*RCF + (1.0-CR1(I))*RCS
      PRINT 8,H11(I),RC1(I)
    8 FORMAT(* FIRST ITERATION     H11=*,F10.4,*RC1=*,F10.4)
    2 CONTINUE
      HWI1(L) = 0.5*(CR1(1)+CR1(2))*HWF(L) +(1.0-0.5*(CR1(1)+CR1(2)))*
     1HWS(L)
      PRINT 9,HWI1(L)
    9 FORMAT(* HWI1=*,F10.4)
C     SECOND ITERATION
      DO 3  I = 1,IBT
      CR2(I) = 0.63662*ATANF(0.5*(HWI1(L)/H11(I))**3*A/B/RC1(I) )
      H12(I) = CR2(I)*HF1(I)+(1.0-CR2(I))*HS1(I)
      PRINT 10,H12(I),CR2(I)
   10 FORMAT(* SECOND ITERATION     H12=*,F10.4,*CR2 =*,F10.4)
    3 CONTINUE
      HWI2(L) = 0.5*(CR2(1)+CR2(2))*HWF(L)+(1.0-0.5*(CR2(1)+CR2(2)))*
     1HWS(L))
      DO 4 I = 1,IBT
      PRINT 11,HWI2(I)
   11 FORMAT(* HWI2=*,F10.4/)
    4 THIKK1(I,1,1) = H12(I)
      THIKK1(IFL) = HWI2(L)
      THIKK1(IIL) = HWI2(L)
      RETURN
      END
```

Subroutine INTERMED



12. Subroutine LONGIT

a) Abstract:

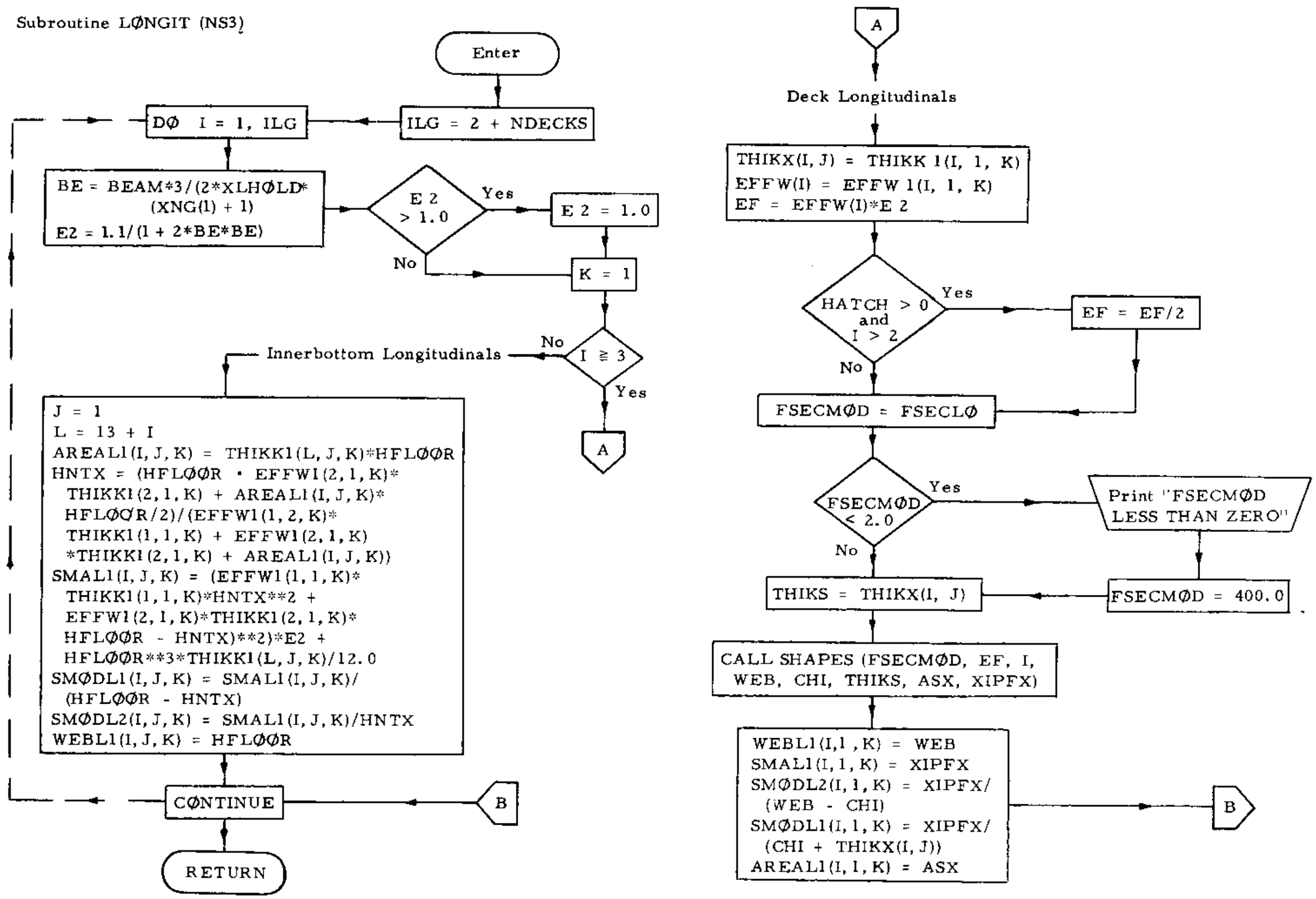
Subroutine LONGIT is called from subroutine SECTION. It calculates the section moduli of all longitudinals. The cross sectional area, the second central moment of area and the section modulus of O. T. and N. T. double bottom longitudinal girders are computed directly in subroutine LONGIT. For all other longitudinals the above quantities are obtained via subroutine SHAPES, which is called from subroutine LONGIT.

b) Terms specific to this subroutine:

FORTRAN Term	Definition	Mathematical Symbol
BE	Aspect ratio parameter	β
E2	Normalized effective breadth	λ
EF	Combined effective breadth-width	

```
      SUBROUTINE LONGIT (NS3)
      COMMON/LO/FSECLO(12),FSESL(17),FSESL1(17),SMODL2(17,2,3),[illegible](17)
C     LONGITUDINALS
      COMMON /E/ IBT,E
      COMMON /B/ A,B,NDECKS,IPLATE,PSL,PRM,HMAIN,HNEUT,[illegible],[illegible],[illegible],
     1     SEST,          KPANELS,BEAM,XNG,HATCH,THIKK,THIKK1,[illegible],[illegible]B1,
     2     EFFW,EFFW1,HFLOOR,TWEENH,THIKX,EFBR,WEB,WEB1,X[illegible],[illegible],
     3     XIS, AREAT1,SMAL1,SMODL1,WEBL1,SUMAREAP,SUMAREAL,[illegible],SUMML,
     4     AREA,SUMAREA,SUMMOM,WPL,XPANEL,WBAYLP,XNST,[illegible],[illegible],
     5     WTR,BAYTR,SUMWT,SUMWTR,LLGX,   YIELD,WBAYTR,XLAM[illegible],[illegible],
     6     COST,COSTMIN,GNOT,GNNT,KNEUT,SUMSMAP,SUMSMAL,[illegible],[illegible],
     7WAVEH,PRHEAD,DRAFT,ALOAD,XLHOLD,XLPANEL,AAA,RBB,AA,[illegible],[illegible],
     8      SUMSMA1,ZP,ZL,ILG,AREAL1,ASX,   SMAT1,SMODT1
      DIMENSION ZP(17),SESL(17),SEST(17),TESL(17),TEST(17),[illegible],THIKK(
     117),        THIKK1(17,2,5),EFFBR(17),EFFW(17),EFFB1(17,[illegible],[illegible]1(17
     2,2,5),AREA(17,5),TWEENH(5),SMAT1(12,2,5),SMODT1(12,2,[illegible]),[illegible](12,2
     3),EFBR(12,2),AREAT1(12,2,5),WEB1(12,2,5),SMAL1(17,2,[illegible]),[illegible](17,2
     4,3),WEBL1(7,2,3),AREAL1(7,2,3),SUMAREAP(5),SUMAREAL(5),[illegible](5),SU
     5MML(5),SUMAREA(5),SUMMOM(5),WPL(5),XLPANEL(5),SUMWTR(5),[illegible](12),ZL
     6(7),TIKK1(17,5),CODE(17),ALOAD(17),SUMSMAP(5),SUMSMAL(5),SUMSMA(5)
      ILG  = 2 + NDECKS
      DO 130 I =1,ILG
      BE=      BEAM   *3/2/XLHOLD /(XNG(I)+1)
      E2=1.1/(1.+2.*BE*BE)
      IF(E2.GT.1.0)  E2=1.0
      K=1
      GO TO (131,131,133) I
C     INNERBOTTOM LONGITUDINALS
  131 J=1
      L=13+I
      AREAL1(I,J,K) = THIKK1(L,J,K) * HFLOOR
      HNTX=(HFLOOR*EFFW1(2,1,K)*THIKK1(2,1,K)+AREAL1(I,J,K)*HFLOOR/2.)/(
     1EFFW1(1,2,K)*THIKK1(1,1,K)+ EFFW1(2,1,K)* THIKK1(2,1,K)
     3 +AREAL1(I,J,K)   )
      SMAL1(I,J,K) =(EFFW1(1,1,K)*THIKK1(1,1,K) * HNTX ** 2
     1            + EFFW1(2,1,K)*THIKK1(2,1,K)*(HFLOOR- HNTX) **2 )*E2
     2  +HFLOOR**3*THIKK1(L,J,K)/12.
      SMODL1(I,J,K) = SMAL1(I,J,K)/ (HFLOOR - HNTX)
        SMODL2(I,J,K)=SMAL1(I,J,K)/HNTX
      WEBL1(I,J,K) = HFLOOR
      GO TO 130
  133 CONTINUE
C     DECK LONGITUDINALS
      THIKX(I,J) = THIKK1(I,1,K)
      EFFW(I) = EFFW1(I,1,K)
      EF=EFFW(I) *E2
      IF(HATCH.GT.0.0.AND.I.GT.2)  EF=E
      FSECMOD=FSECLO(I)
          IF(FSECMOD.LT. 2.)  PRINT 554
 5541 FORMAT( /,*  FSECMOD LESS THAN ZE
          IF(FSECMOD.LT. 2.)    FSECMOD
 8995 FORMAT(/,*  SECMOD = *,E13.5, *I
       THIKS=THIKX(I,J)
      CALL          SHAPES  (FSECMOD,EF ,I
      WEBL1(I,1,K) = WEB
      SMAL1(I,1,K) = XIPFX
      SMODL2(I,1,K)= XIPFX /(WEB-CHI)
      SMODL1(I,1,K)= XIPFX /(CHI+THIKX(
      AREAL1(I,1,K)=ASX
  130 CONTINUE
      RETURN
      END
```

Subroutine LØNGIT (NS3)



13. Subroutine LONGMAT

a) Abstract

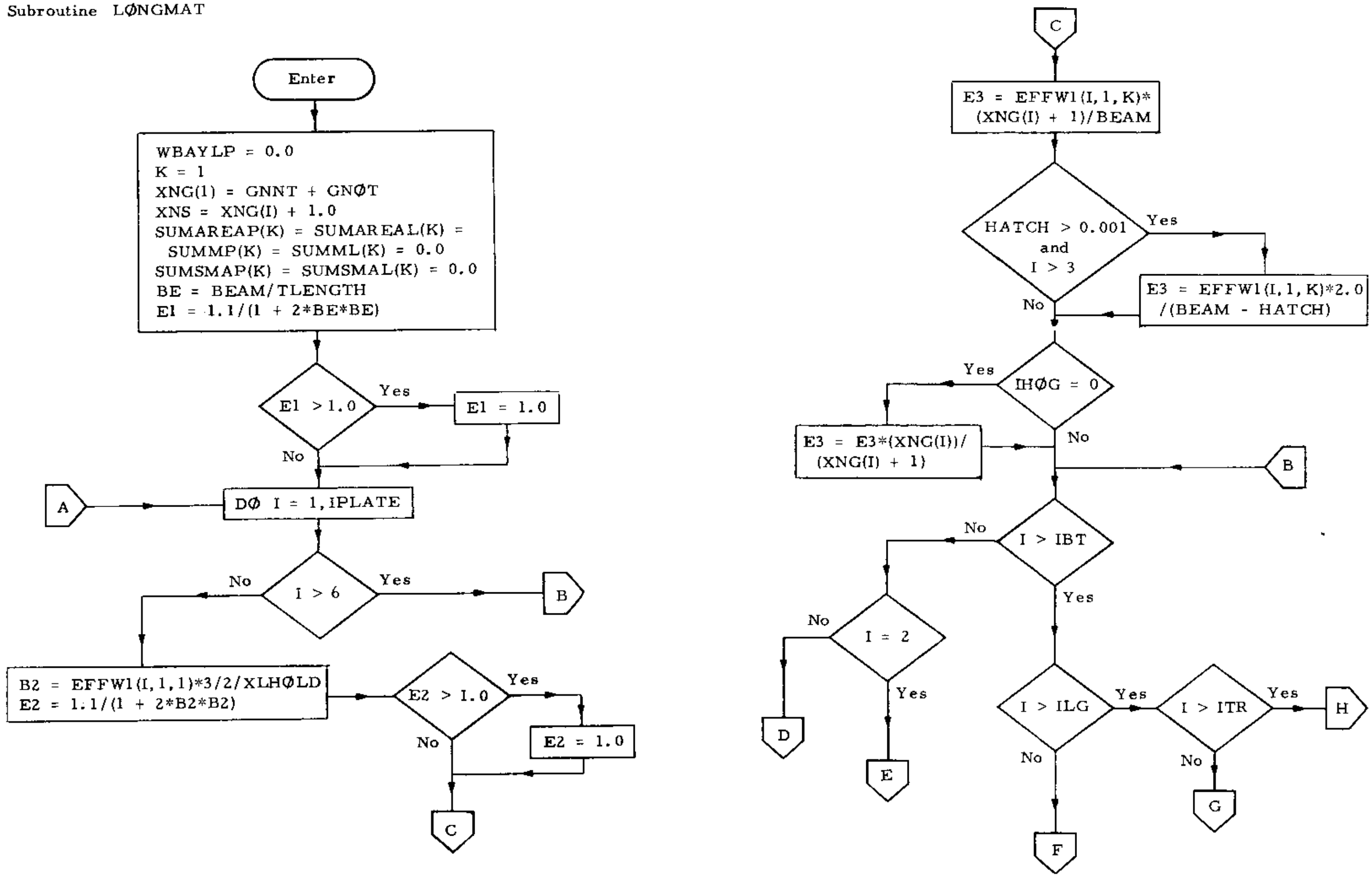
This subroutine is called from SECTION. It calculates the cross sectional area as well as first and second central moment of area of all longitudinal members.

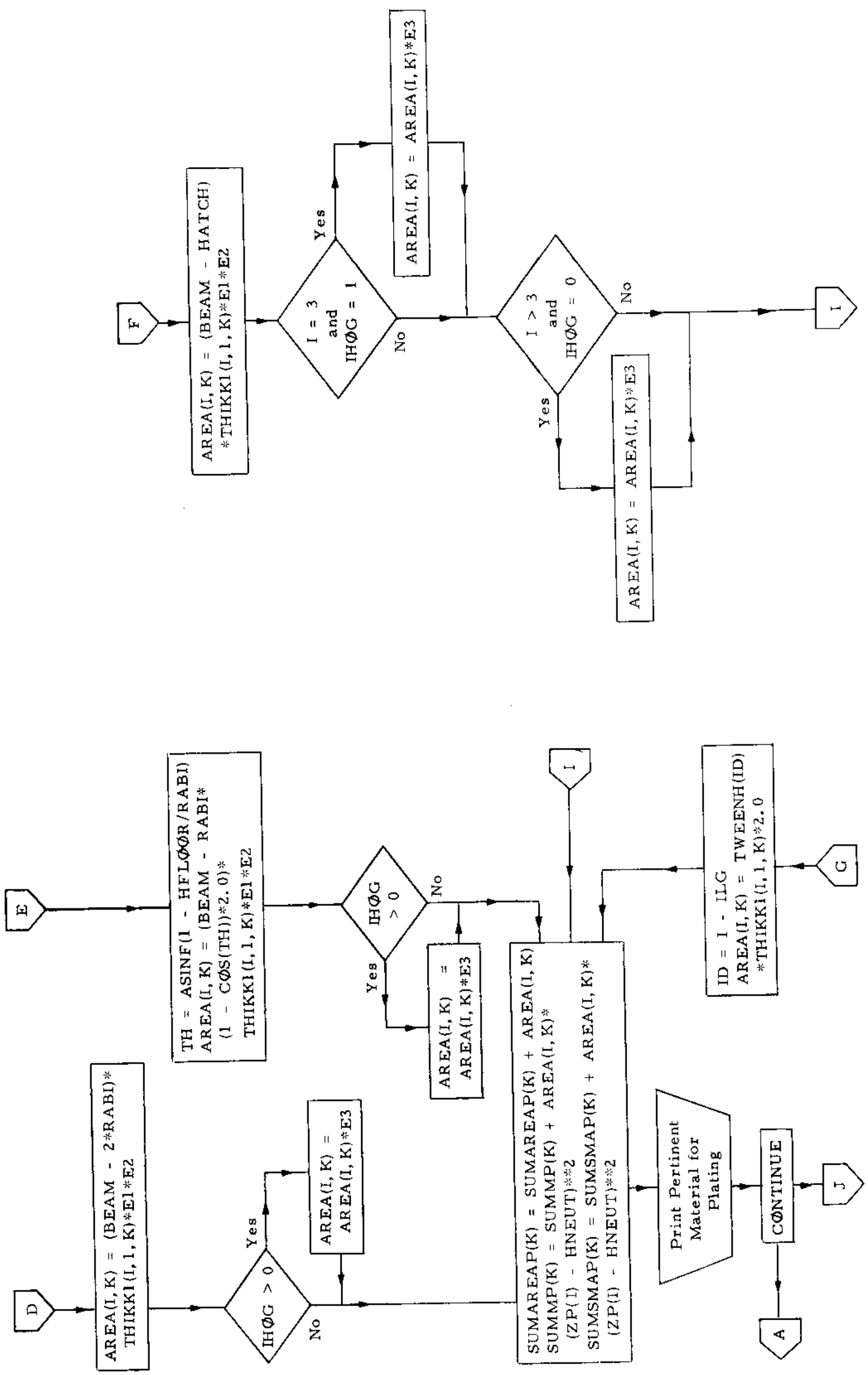
b) Terms specific to this subroutine:

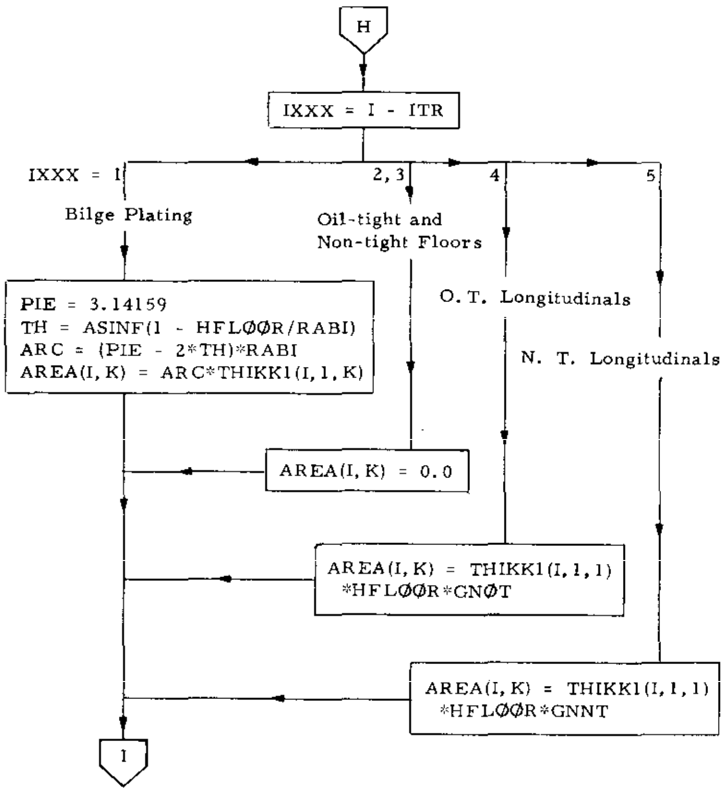
FORTRAN Term	Definition	Mathematical Symbol
ARC	Arc length of bilge plating	
AREA	Cross-sectional area of plating	
B2	Aspect ratio parameter	β
BE	Aspect ratio parameter	β
E1	Normalized effective breadth	λ
E2	Normalized effective breadth	λ
WBAYLP	Weight of longitudinals and plating per hold	

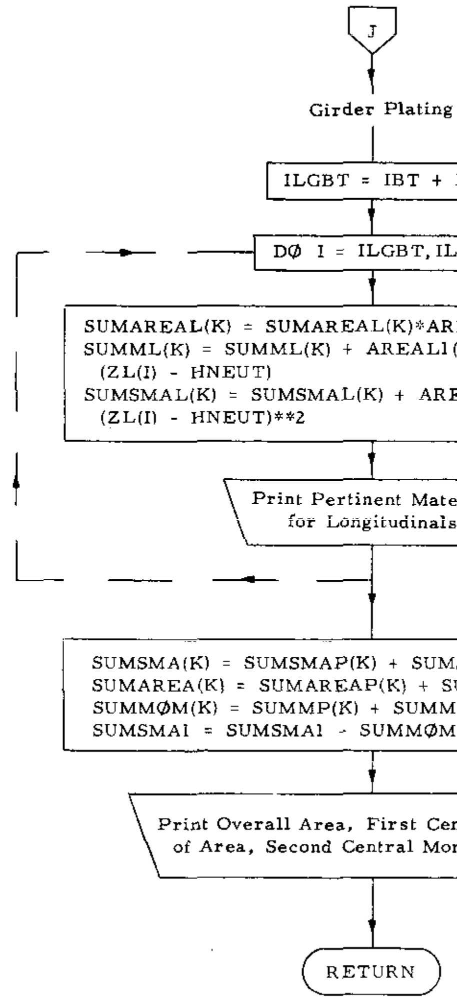
```
      SUBROUTINE LONGMAT
C     LONGITUDINAL MATERIAL
      COMMON/TLEN/  TLENGTH     ,IHOG ,RABI
      COMMON/LABEL/ TRANLAB(10),PLATLAB(12),LONGLAB(8)
      COMMON /B/ A,B,NBLOCKS,IPLATE,PSL,PRM,HMAIN,HNEUT,TESL,TEST,SESL,
     1    SEST,        KPANELS,BEAM,XNG,HATCH,THIKK,THIKK1,EFFBR,EFFB1,
     2    EFFW,EFFW1,HFLOOR,TWEENH,THIKX,EFBR,WEB,WEB1,XIPIX,CHI,
     3    XIS, AREAT1,SMAL1,SMODL1,WEBL1,SUMAREAP,SUMAREAL,SUMMP,SUMML,
     4    AREA,SUMAREA,SUMMOM,WPL,XPANEL,WBAYLP,XNST,NST,ITR,
     5    WTR,BAYTR,SUMWT,SUMWTR,LLGX,  YIELD,WBAYTR,XLABOR,PLCOST,
     6    COST,COSTMIN,GNOT,GNNT,KNEUT,SUMSMAP,SUMSMAL,SUMSMA,CODE,
     7WAVEH,PRHEAD,DRAFT,ALOAD,XLHOLD,XLPANEL,AAA,BBB,AA,BB,DELTESL,
     8     SUMSMA1,ZP,ZL,ILG,AREAL1,ASX,  SHAT1,SMODT1
      COMMON /E/ IBT,E
      DIMENSION ZP(17),SESL(17),SEST(17),TESL(17),TEST(17),XNG(7),THIKK(
     117),       THIKK1(17,2,5),EFFBR(17),EFFW(17),EFFB1(17,2,5),EFFW1(17
     2,2,5),AREA(17,5),TWEENH(5),SHAT1(12,2,5),SMODT1(12,2,5),THIKX(12,2
     3),EFBR(12,2),AREAT1(12,2,5),WEB1(12,2,5),SMAL1(17,2,3),SMODL1(17,2
     4,3),WEBL1(7,2,3),AREAL1(7,2,3),SUMAREAP(5),SUMAREAL(5),SUMMP(5),SU
     5MML(5),SUMAREA(5),SUMMOM(5),WPL(5),XLPANEL(5),SUMWTR(5),WTR(12),ZL
     6(7),TIKK1(17,5),CODE(17),ALOAD(17),SUMSMAP(5),SUMSMAL(5),SUMSMA(5)
      WBAYLP =0,0
        K=1
      XNG( 1)=GNNT+GNOT
      XNS=XNG(I)+1,0
      SUMAREAP(K)=SUMAREAL(K)=SUMMP(K)= SUMML(K)=0,0
      SUMSMAP(K) - SUMSMAL(K) = 0,0
      BE=BEAM/TLENGTH
      E1=1.1/(1,+2,*BE*BE)
      IF(E1,GT,1,0) E1=1,0
      DO 140  I =1,IPLATE
      IF(I,GT,6)  GO TO  898
      B2=      EFFW1(I,1,1)*3/2/XLHOLD
      E2=1,1/(1,+2,*B2*B2)
      IF(E2,GT,1,0) E2=1,0
      E3=EFFW1(I,1,K)*(XNG(I)+1,)/BEAM
      IF(HATCH,GT, 0,001,AND,I,GE,3) E3=EFFW1(I,1,K)*2,0/(BEAM-HATCH)
      IF(IHOG,EQ,0)  E3 =E3*(XNG(I) )/(XNG(I)+1)
  898 IF(I,GT,IBT) GO TO 142
      IF(I,EQ,2)  GO TO 141
      AREA(I,K)=(BEAM-2,0*RABI)*THIKK1(I,1,K)    *E1*E2
      IF(IHOG,GT,0)   AREA(I,K)= AREA(I,K)*E3
      GO TO 143
  141 TH=ASINF(  1,-HFLOOR/RABI)
      AREA(I,K)=(BEAM-RABI*(1-COS(TH))*2,0) *THIKK1(I,1,K)    *E1*E2
      IF(IHOG,GT,0)   AREA(I,K)= AREA(I,K)*E3
      GO TO 143
  142 IF(I,GT,ILG) GO TO 145
  144 AREA(I,K)=(BEAM-HATCH)*THIKK1(I,1,K)*E1*E2
      IF(I ,EQ.3,AND,IHOG,EQ,1)  AREA(I,K) =AREA(I,K)*E3
      IF(I ,GT,3,AND,IHOG,EQ,0)  AREA(I,K) =AREA(I,K)*E3
      GO TO 143
  145 IF(I,GT,ITR) GO TO 147
  146 ID= I-ILG
      AREA(I,K)=TWEENH(ID)* THIKK1(I,1,K) *2,0
      GO TO 143
  147 IXXX = I - ITR
      GO TO ( 148,149,149,161,162)IXXX
C     BILGEPLATING
  148 PIE=3,14159
      TH=ASINF(  1,-HFLOOR/RABI)
      ARC=(PIE-2,0*TH)*RABI
      AREA(I,K) =   ARC*THIKK1(I,1,K)
      GO TO 143
  149 AREA(I,K) = 0,0
      GO TO 143
  161 AREA(I,K) = THIKK1(I,1,1)* HFLOOR * GNOT
      GO TO 143
  162 AREA(I,K) = THIKK1(I,1,1)* HFLOOR * GNNT
  143 SUMAREAP(K) = SUMAREAP(K) + AREA(I,K)
      SUMMP(K)= SUMMP(K)+ AREA(I,K) *(ZP(I)-HNEUT)
      SUMSMAP(K) =SUMSMAP(K) +AREA(I,K) *(ZP(I)- HNEUT) **2
 9053 PRINT 9163 ,PLATLAB(I),THIKK1(I,1,1),AREA(I,1),SUMAREAP(1),ZP(I),
     1SUMMP(1), SUMSMAP(1),E2,E3
  140 CONTINUE
C     GIRDER PLATING
      ILGBT = IBT + 1
      DO 150 I = ILGBT,ILG
      SUMAREAL(K)=SUMAREAL(K)+AREAL1(I,1,K)*XNG(I)
      SUMML(K) = SUMML(K) + AREAL1(I,1,K) *XNG(I) * (ZL(I)- HNEUT)
      SUMSMAL(K) =SUMSMAL(K) + AREAL1(I,1,K)*XNG(I) *(ZL(I)-HNEUT)**2
 9353 PRINT 9163,LONGLAB(I),AREAL1(I,1,1),SUMAREAL(1),ZL(I),SUMML(1) ,
     1SUMSMAL(1)
 9163 FORMAT( * *,A8,8F10,2/ )
  150 CONTINUE
 8153 FORMAT(1H0,* HNEUT =*,F10,2)
      SUMSMA (K)= SUMSMAP(K) +SUMSMAL(K)
      SUMAREA(K) = SUMAREAP(K) + SUMAREAL(K)
      SUMMOM(K) = SUMMP(K) +  SUMML(K)
  160 CONTINUE
      SUMSMA1  =  SUMSMA(1) - SUMMOM(1)* *2 / SUMAREA(1)
      PRINT 9355 ,SUMAREA(1),SUMMOM(1),SUMSMA(1) ,E1
 9355 FORMAT( 4F10,2)
      RETURN
      END
```

Subroutine LØNGMAT




F
AREA(I, K) = (BEAM - HATCH)
*THIKK1(I, 1, K)*E1*E2
I = 3
and
IHØG = 1
Yes
No
AREA(I, K) = AREA(I, K)*E3
I > 3
and
IHØG = 0
Yes
No
AREA(I, K) = AREA(I, K)*E3
I
D
AREA(I, K) = (BEAM - 2*RABI)*
THIKK1(I, 1, K)*E1*E2
IHØG > 0
Yes
No
AREA(I, K) =
AREA(I, K)*E3
E
TH = ASINF(1 - HFLØØR/RABI)
AREA(I, K) = (BEAM - RABI*
(1 - CØS(TH))*2.0)*
THIKK1(I, 1, K)*E1*E2
IHØG
> 0
Yes
No
AREA(I, K) =
AREA(I, K)*E3
I
G
ID = I - ILG
AREA(I, K) = TWEENH(ID)
*THIKK1(I, 1, K)*2.0
SUMAREAP(K) = SUMAREAP(K) + AREA(I, K)
SUMMP(K) = SUMMP(K) + AREA(I, K)*
(ZP(I) - HNEUT)**2
SUMSMAP(K) = SUMSMAP(K) + AREA(I, K)*
(ZP(I) - HNEUT)**2
Print Pertinent
Material for
Plating
CØNTINUE
J
A


H
IXXX = I - ITR
IXXX = 1
2, 3
4
5
Bilge Plating
Oil-tight and
Non-tight Floors
O. T. Longitudinals
N. T. Longitudinals
PIE = 3.14159
TH = ASINF(1 - HFLØØR/RABI)
ARC = (PIE - 2*TH)*RABI
AREA(I, K) = ARC*THIKK1(I, 1, K)
AREA(I, K) = 0.0
AREA(I, K) = THIKK1(I, 1, 1)
*HFLØØR*GNØT
AREA(I, K) = THIKK1(I, 1, 1)
*HFLØØR*GNNT
I


J
Girder Plating
RETURN

14. Subroutine MATINV

a) Abstract:

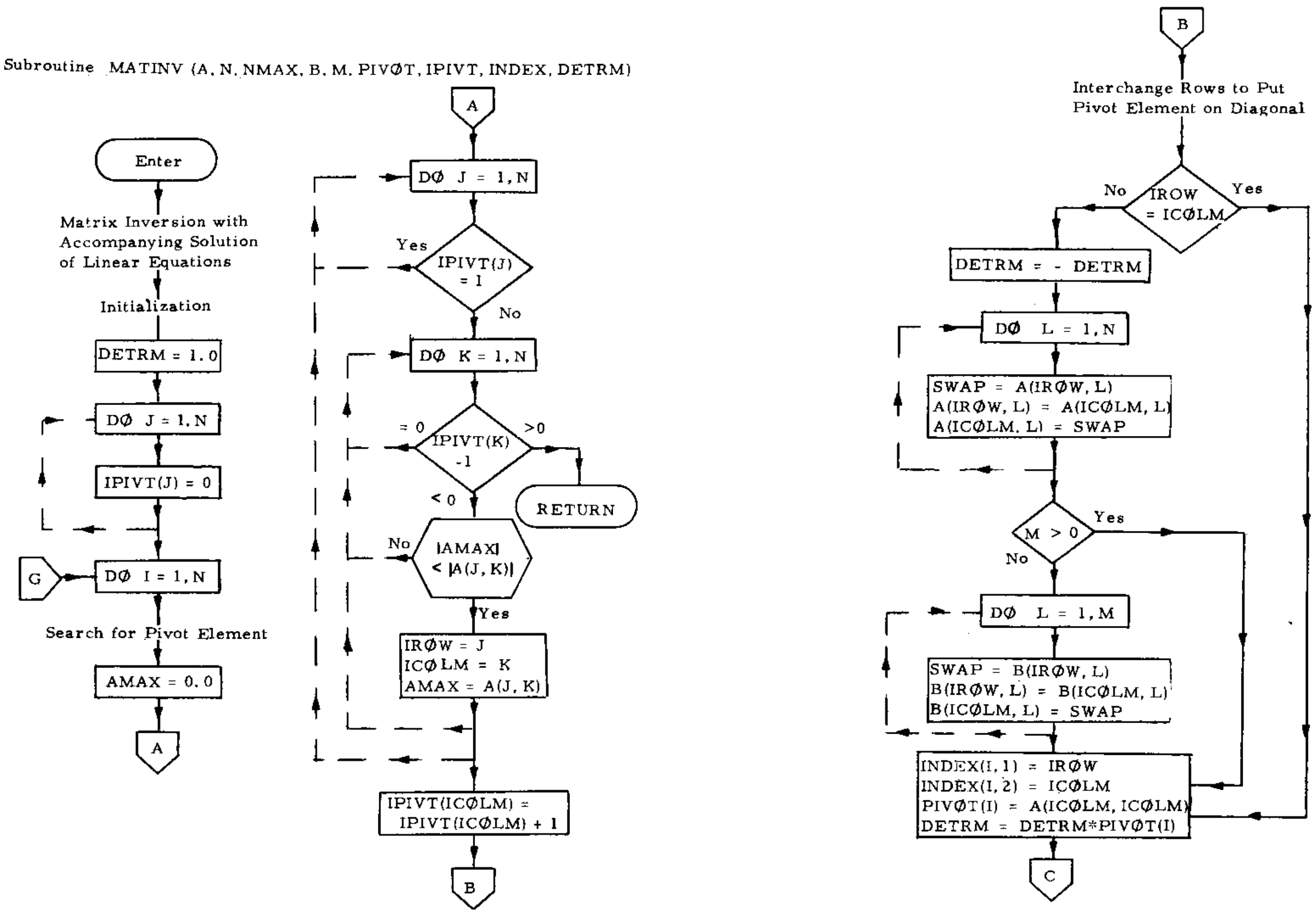
This subroutine calculates the inverse matrix of a square matrix, with accompanying solution of linear equations.

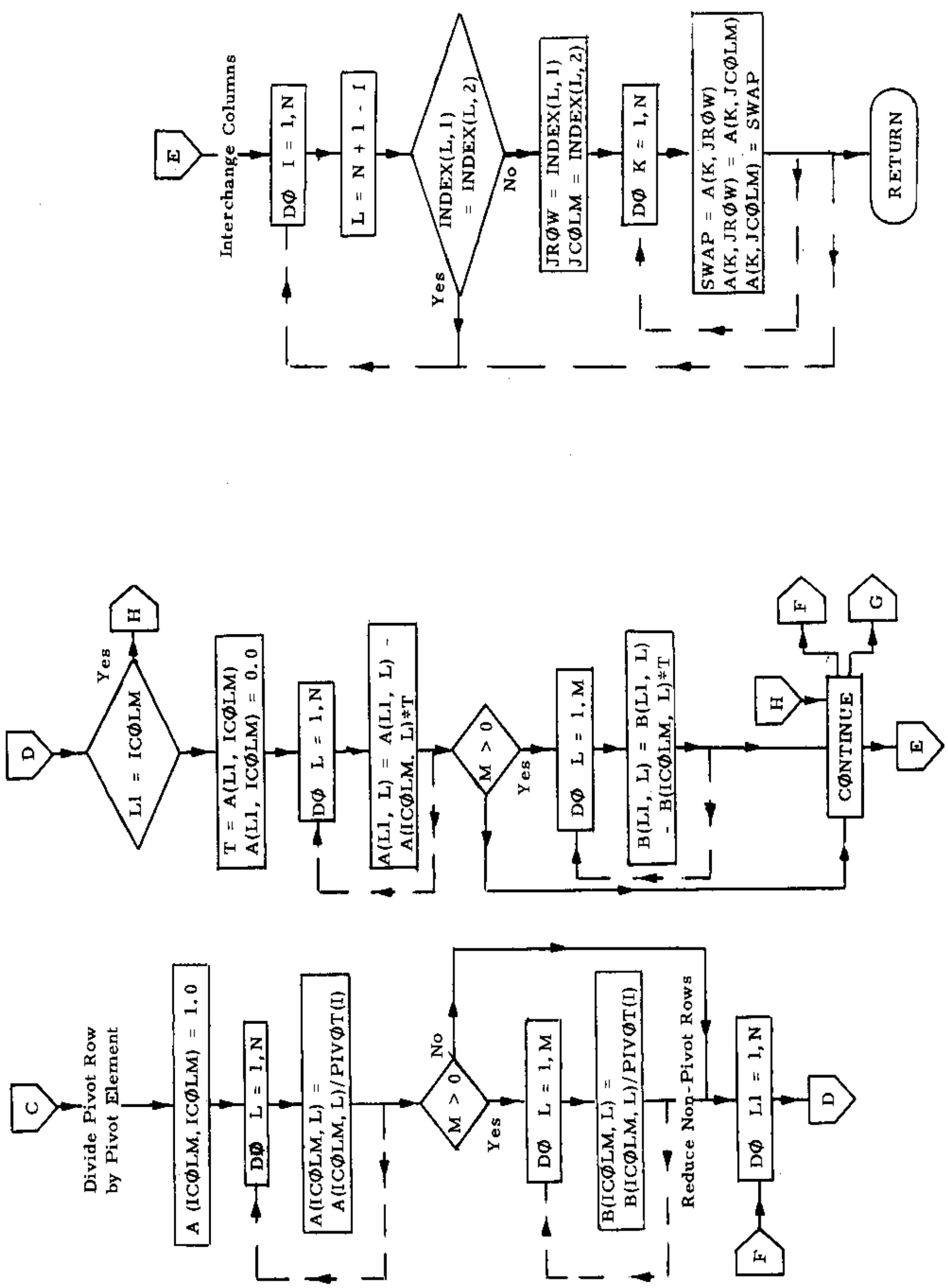
b) Terms specific to this subroutine:

FORTRAN Term	Definition
A	Input array
B	Solution matrix for linear equations
DETRM	Out put argument for determinant of A
N	Degree of input array

```
      SUBROUTINE MATINV (A,N,NMAX,B,M,PIVOT,IPIVT ,INDEX,DETRM)
      DIMENSION IPIVT(N),A(NMAX,NMAX),B(NMAX,1),INDEX(N,2),PIVOT(N)
      F1 CODA MATIN   MATRIX INVERSION
C     MATRIX INVERSION WITH ACCOMPANYING SOLUTION OF LINEAR EQUATIONS
C
C
C
C     INITIALIZATION
C
   10 DETRM =1.0
   15 DO 20 J=1,N
   20 IPIVT (J)=0
   30 DO 550 I=1,N
C
C     SEARCH FOR PIVOT ELEMENT
C
   40 AMAX=0.0
   45 DO 105 J=1,N
   50 IF (IPIVT (J)-1) 60, 105, 60
   60 DO 100 K=1,N
   70 IF (IPIVT (K)-1) 80, 100, 740
   80 IF (ABS (AMAX)-ABS (A(J,K))) 85, 100, 100
   85 IROW=J
   90 ICOLM =K
   95 AMAX=A(J,K)
  100 CONTINUE
  105 CONTINUE
  110 IPIVT (ICOLM) =IPIVT (ICOLM) +1
C
C     INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL
C
  130 IF (IROW-ICOLM)  140, 260, 140
  140 DETRM =-DETRM
  150 DO 200 L=1,N
  160 SWAP=A(IROW,L)
  170 A(IROW,L)=A(ICOLM, L)
  200 A(ICOLM, L)=SWAP
  205 IF(M) 260, 260, 210
  210 DO 250 L=1, M
  220 SWAP=B(IROW,L)
  230 B(IROW,L)=B(ICOLM, L)
  250 B(ICOLM, L)=SWAP
  260 INDEX(I,1)=IROW
  270 INDEX(I,2)=ICOLM
  310 PIVOT(I)=A(ICOLM, ICOLM)
  320 DETRM =DETRM *PIVOT(I)
C
C     DIVIDE PIVOT ROW BY PIVOT ELEMENT
C
  330 A(ICOLM, ICOLM) =1.0
  340 DO 350 L=1,N
  350 A(ICOLM, L)=A(ICOLM, L)/PIVOT(I)
  355 IF(M) 380, 380, 360
  360 DO 370 L=1,M
  370 B(ICOLM, L)=B(ICOLM, L)/PIVOT(I)
C
C     REDUCE NON-PIVOT ROWS
C
  380 DO 550 L1=1,N
  390 IF(L1-ICOLM)  400, 550, 40
  400 T=A(L1,ICOLM)
  420 A(L1,ICOLM) =0.0
  430 DO 450 L=1,N
  450 A(L1,L)=A(L1,L)-A(ICOLM, L
  455 IF(M) 550, 550, 460
  460 DO 500 L=1,M
  500 B(L1,L)=B(L1,L)-B(ICOLM, L
  550 CONTINUE
C
C     INTERCHANGE COLUMNS
C
  600 DO 710 I=1,N
  610 L=N+1-I
  620 IF (INDEX(L,1)-INDEX(L,2))
  630 JROW=INDEX(L,1)
  640 JCOLM =INDEX(L,2)
  650 DO 705 K=1,N
  660 SWAP=A(K,JROW)
  670 A(K,JROW)=A(K,JCOLM)
  700 A(K,JCOLM) =SWAP
  705 CONTINUE
  710 CONTINUE
  740 RETURN
  750 END
```

Subroutine MATINV (A, N, NMAX, B, M, PIVØT, IPIVT, INDEX, DETRM)




E
Interchange Columns
DØ I = 1, N
L = N + 1 - I
INDEX(L, 1) = INDEX(L, 2)
Yes
No
JRØW = INDEX(L, 1)
JCØLM = INDEX(L, 2)
DØ K = 1, N
SWAP = A(K, JRØW)
A(K, JRØW) = A(K, JCØLM)
A(K, JCØLM) = SWAP
RETURN
D
L1 = ICØLM
Yes
H
T = A(L1, ICØLM)
A(L1, ICØLM) = 0.0
DØ L = 1, N
A(L1, L) = A(L1, L) - A(ICØLM, L)*T
M > 0
Yes
DØ L = 1, M
B(L1, L) = B(L1, L) - B(ICØLM, L)*T
H
CØNTINUE
F
G
E
C
Divide Pivot Row by Pivot Element
A (ICØLM, ICØLM) = 1.0
DØ L = 1, N
A(ICØLM, L) = A(ICØLM, L)/PIVØT(I)
M > 0
No
Yes
DØ L = 1, M
B(ICØLM, L) = B(ICØLM, L)/PIVØT(I)
Reduce Non-Pivot Rows
DØ L1 = 1, N
D
F

15. Subroutine NTPLATE

a) Abstract:

Subroutine NTPLATE is called from subroutine ASPECT. It calculates the thicknesses of the non-tight floors and non-tight longitudinals.

The non-tight floors are designed to the critical buckling shear stress or to the maximum allowable shear stress intensity in the floor web, depending upon which will yield the greater floor thickness.

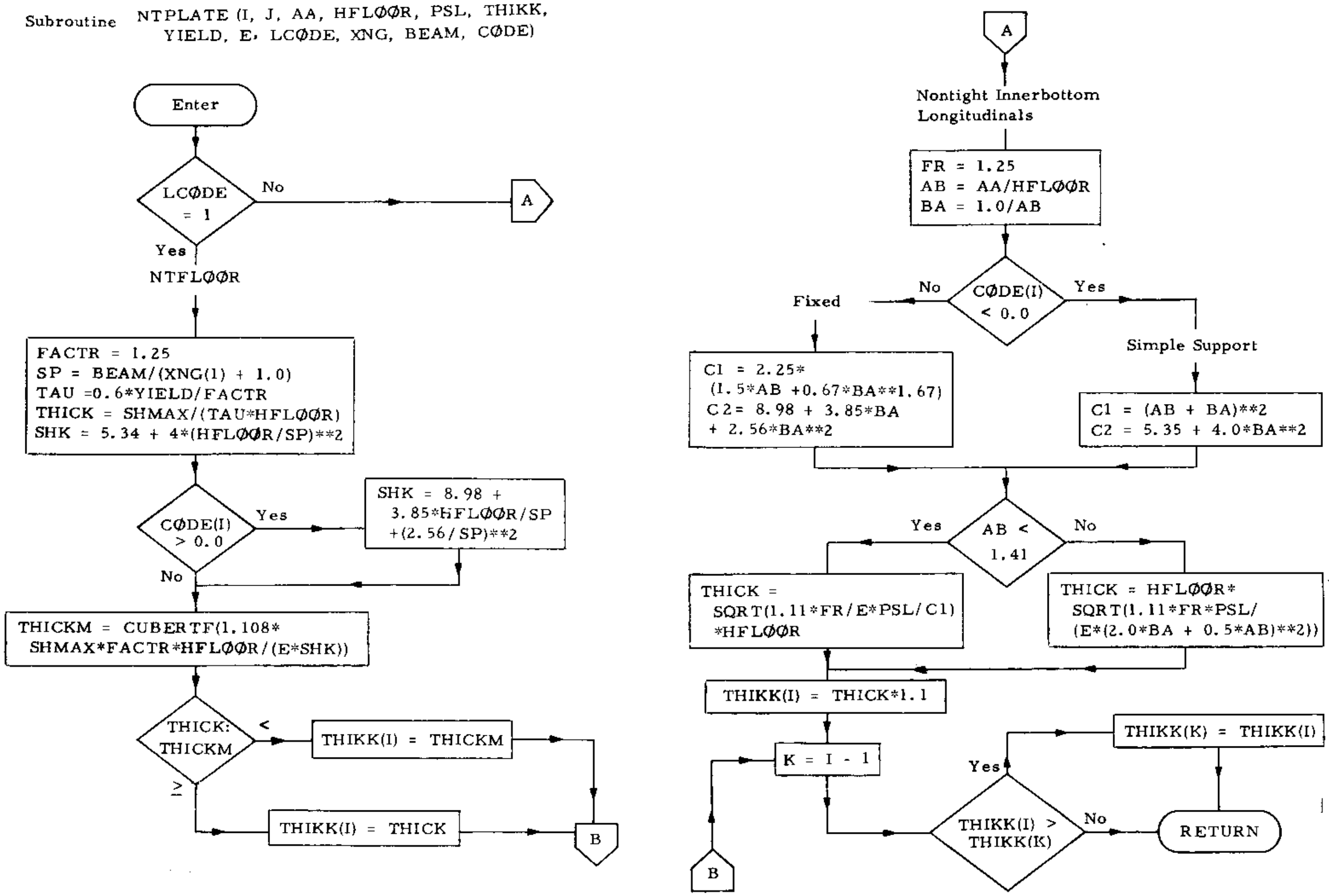
The non-tight longitudinals do not carry any lateral load, hence they are only subject to in-plane loading, and are designed to the critical buckling compressive load. The critical buckling load depends on the aspect ratio of the plate and separate equations are used depending upon whether the aspect ratio is greater or smaller than $\sqrt{2}$.

b) Terms specific to this subroutine:

FORTRAN Term	Definition
AB	Aspect ratio
SP	Span of floor (between longitudinals)
TAU	Shear stress intensity
THICKM	Minimum plate thickness with respect to buckling stability

```
      SUBROUTINE NTPLATE(I,J,AA,HFLOOR,PSL,THIKK,YIELD,E,LCODE,XNG,BEAM,
     1  CODE)
      DIMENSION THIKK(17) ,XNG(7)
      DIMENSION CODE(17)
      COMMON/SH/   SHMAX
      IF(LCODE.EQ.1)  GO TO 78
C     NONTIGHT INNERBOTTOM LONGITUDINALS
      FR=1.25
      AB =AA/HFLOOR
      BA = 1.0/AB
      IF(CODE(I).LT.0.0)  GO TO  22
C    FIXED
      C1=2.25*(1.5*AB+0.67* BA **1.67 )
      C2=8.98+3.85*BA+2.56*BA**2
      GO TO 23
C    SIMPLE SUPPORT
 22   C1=AB+BA**2
      C2=5.35+4.0*BA**2
 23   IF(AB- 1.41) 1,2,2
    2 THICK =  HFLOOR *SQRTF(1.11 *   FR* PSL/(E *(2.0*BA + 0.5*AB)**2))
      GO TO 3
  1   THICK= SQRT(1.11*FR/E*  PSL/C1)*HFLOOR
    3 THIKK(I) = THICK*  1.1
      GO TO 99
 78    CONTINUE
C   NTFLOOR
 9889 FORMAT(* SHMAX *,F10.2)
      FACTR = 1.25
      SP=BEAM/(XNG(1)+1.0    )
      TAU = 0.6 * YIELD    /   FACTR
      THICK = SHMAX/ (TAU * HFLOOR)
      SHK = 5.34 +4.0*(HFLOOR/SP)**2
      IF(CODE(I).GT.0.0)SHK=8.98+ 3.85*HFLOOR/SP+(2.56/SP)**2
      THICKM = CUBERTF(1.108 * SHMAX*FACTR*HFLOOR /(E*SHK))
      IF(THICK - THICKM) 11,12,12
   11 THIKK(I) = THICKM
      GO TO 13
   12 THIKK(I) =THICK
 13    CONTINUE
 99   K=I+1
      IF( THIKK(I). GT. THIKK(K) )   THIKK(K)=THIKK(I)
      RETURN
      END
```

Subroutine NTPLATE (I, J, AA, HFLØØR, PSL, THIKK, YIELD, E, LCØDE, XNG, BEAM, CØDE)



16. Subroutine PLATING

a) Abstract:

Subroutine PLATING is called from subroutine ASPECT. Thicknesses of platings are determined by the method of Bengston.

b) Description:

A first approximation to the thickness of plating is obtained by linear theory and upon neglect of in-plane loading. Then the deflection at the center of the plate rectangle is determined by solving the deflection equation of Bengston. This is done using the Bolzano Bisection Method. Knowing the deflection (WO), the maximum bending stress intensities with and without compressive stresses are calculated as functions of the deflection (WO). These stresses are compared with the tertiary design stress intensities and if they are larger than the latter, the plate thickness is increased and the process of computation repeated, replacing the initially obtained plate thicknesses by the newly calculated one.

When the solution has converged on a plate thickness which satisfies the design stress intensity, the plating is checked for stability, and if necessary, the plate thickness is further increased. Having established the correct thickness, the effective width of the plating is calculated.

c) Terms specific to this subroutine:

FORTRAN Term	Definition	Mathematical Symbol
AB	Aspect ratio	
ACTB	Actual buckling stress intensity	
BUCR	Buckling ratio	
CRITB	Critical buckling stress intensity	σ_{cr}
DEF	Deflection of plating	
EBRV	Effective breadth according to Schade's formula	
EFFBR	Effective breadth of plating	b_e
EFFW	Effective width of plating	b
FLEXR	Flexural rigidity of plating	D
PHIL	Factor depending on aspect ratio in equation for SXBP	φ_x
PHIS	Factor depending on aspect ratio in equation for SYBP	φ_y

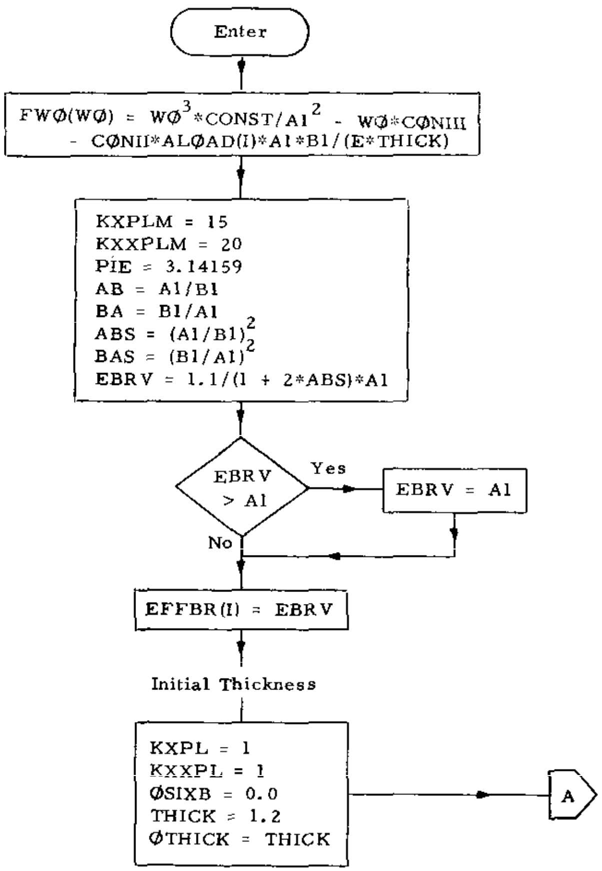
PLATING continued

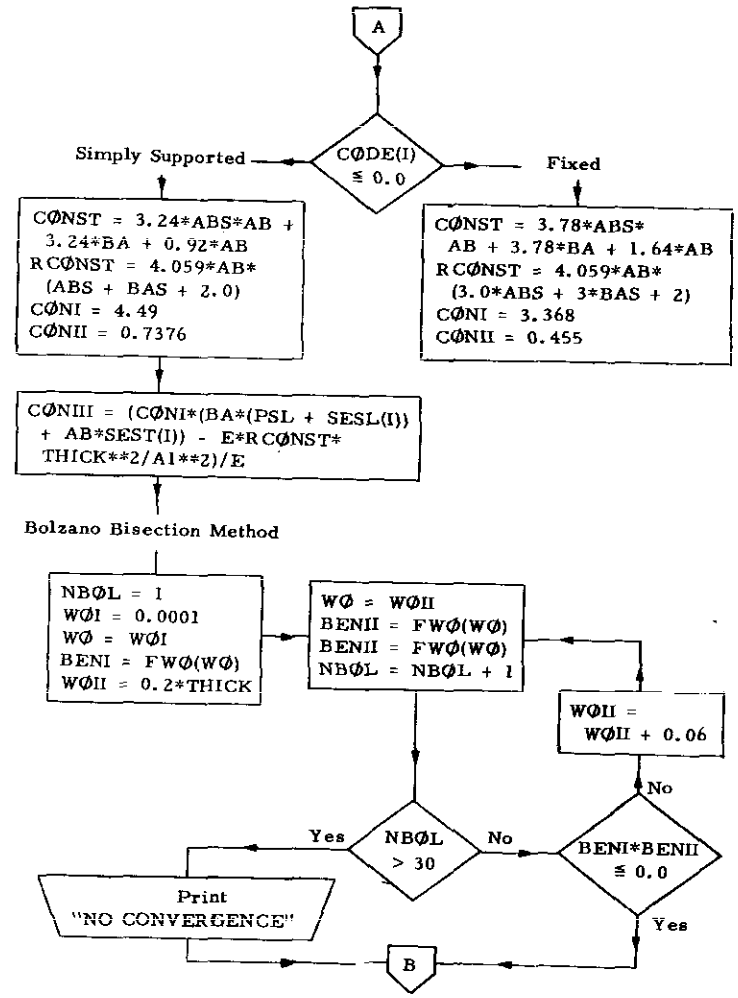
FORTRAN Term	Definition	Mathematical Symbol
SIXB	Maximum bending stress intensities in the longitudinal direction (at the middle of the sides) when the compressive stresses are of such magnitude as to cause buckling	σ'_{xb} σ_{xb}
SIYB	As SIXB, but in transverse direction	σ'_{yb} σ_{yb}
SXBP	Maximum bending stress intensities in the longitudinal direction when there is no compressive stress	σ''_{xb}
SYBP	Maximum bending stress intensities in the transverse direction when there is no compressive stress	σ''_{yb}
THIKK	Plate thickness	h
WØ	Deflection at center of plate rectangle	
WØI	Lower limiting value of the deflection	
WØII	Upper limiting value of the deflection (varying)	

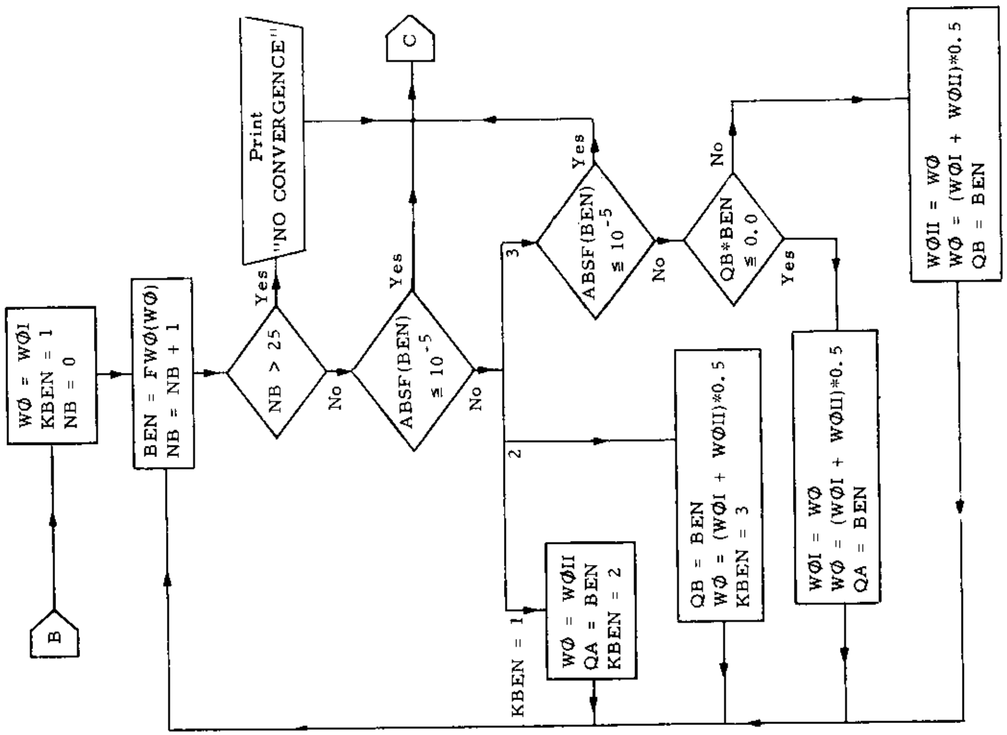
```
      SUBROUTINE PLATING   (A1,B1,I,J)
      COMMON/DE/ DEF(20)
      COMMON /E/ IRT,E
      COMMON /B/ A,B,NDECKS,IPLATE,PSL,PRM,HMAIN,HNEUT,TESL,TEST,SESL,
     1    SEST,       KPANELS,BEAM,XNG,HATCH,THIKK,THIKK1,EFFBR,EFFR1,
     2    EFFW,EFFW1,HFLOOR,TREENH,THIKX,EFBR,WEB,WEB1,XIPEX,CHI,
     3    XIS, AREAT1,SMAL1,SMODL1,WEBL1,SUMAREAP,SUMAREAL,SUMMP,SUMML,
     4    AREA,SUMAREA,SUMMOM,WPL,XPANEL,WBAYLP,XNST,NST,ITR,
     5    WTR,BAYTR,SUMWT,SUMWTR,LLGX,   YIELD,WBAYTR,XLABOR,PLCOST,
     6    COST,COSTMIN,GNOT,GNNT,KNEUT,SUMSMAP,SUMSMAL,SUMSMA,CODE,
     7WAVEH,PRHEAD,DRAFT,ALOAD,XLHOLD,XLPANEL,AAA,BBB,AA,BB,DELTESL,
     8     SUMSMA1,ZP,ZL,ILG,AREAL1,ASX,   SMAT1,SMODT1
      DIMENSION ZP(17),SESL(17),SEST(17),TESL(17),TEST(17),XNG(7),THIKK(
     117),      THIKK1(17,2,5),EFFBR(17),EFFW(17),EFFR1(17,2,5),EFFW1(17
     2,2,5),AREA(17,5),TREENH(5),SMAT1(12,2,5),SMODT1(12,2,5),THIKX(12,2
     3),EFBR(12,2),AREAT1(12,2,5),WEB1(12,2,5),SMAL1(17,2,3),SMODL1(17,2
     4,3),WEBL1(7,2,3),AREAL1(7,2,3),SUMAREAP(5),SUMAREAL(5),SUMMP(5),SU
     5MML(5),SUMAREA(5),SUMMOM(5),WPL(5),XLPANEL(5),SUMWTR(5),WTR(12),ZL
     6(7),TIKK1(17,5),CODE(17),ALOAD(17),SUMSMAP(5),SUMSMAL(5),SUMSMA(5)
      FWO(WO)  =   WO **3 * CONST /A1 ** 2 - WO * CONIII
     2  - CONII * ALOAD(I) *A1*B1 / ( E * THICK )
      KXPLM=15
      KXXPLM = 20
      PIE=3.14159
       AB=A1/B1
      BA=B1/A1
      ABS = (A1/B1) **2
      BAS = (B1/A1)** 2
       EBRV = 1.1/(1.0 + 2.0* ABS)*A1
      IF( EBRV.LE.A1) GO TO 101
      EBRV = A1
  101 EFFBR(I) =EBRV
C     INITIAL THICKNESS
 9113 KXPL= 1
      KXXPL = 1
      OSIXB = 0.0
      THICK = 1.2
      OTHICK = THICK
  112 CONTINUE
      IF(CODE(I))120,120,130
C     SIMPLY SUPPORTED
  120  CONST = 3.24 * ABS * AB + 3.24 * BA + 0.92 * AB
       RCONST = 4.059 * AB * ( ABS * BAS + 2.0 )
       CONI = 4.49
       CONII = 0.7376
       GO  TO 9140
C     FIXED
  130  CONST = 3.78 * ABS * AB + 3.78 * BA + 1.64 * AB
       RCONST = 4.059 * AB * ( 3.0 * ABS + 3.0 * BAS + 2.0 )
       CONI = 3.368
       CONII = 0.455
       GO  TO 9140
C
C
 9140 CONIII = ( CONI * ( BA * ( PSL + SESL(I))+ AB * SEST(I) )
     2  - E * RCONST * THICK ** 2 / A1** 2  ) / E
C     BOLZANO BISECTION METHOD
      NBOL =1
      WOI = 0.0001
      WO = WOI
      BENI      = FWO(WO)
       WOII = 0.2*THICK
   14 WO = WOII
      BENII     =FWO(WO)
      NBOL = NBOL + 1
      IF(NBOL.GT.50) GO TO 16
      IF(BENI*BENII)13,13,12
   12  WOII = WOII + 0.06
      GO TO 14
   16 PRINT 9145
   13 WO = WOI
       KBEN = 1
      NB = 0
    4 BEN       = FWO(WO)
       NB = NB + 1
      IF(NB.GT.25) GO TO 9
      IF(ABSF(BEN).LE.1.E-5) GO TO 5
      GO TO (1,2,3) KBEN
    1 WO = WOII
      QA = BEN
       KBEN = 2
      GO TO 4
    2 QB = BEN
      WO = (WOI + WOII) * 0.5
      KBEN = 3
      GO TO 4
    3 CONTINUE
      IF(ABSF(BEN)- 1.0E-5)5,5,6
    6  IF(QB*BEN)7,7,8
    7 WOI = WO
      WO = (WOI+WOII)*0.5
      QA = BEN
      GO TO 4
    8 WOII = WO
      WO = (WOI+WOII)*0.5
      QB = BEN
      GO TO 4
    9 PRINT 9145
 9145 FORMAT(*NO CONVERGENCE*)
    5 CONTINUE
  144 CONTINUE
       FLEXR = E * THICK ** 3.0/10.92
      DEF(I)=WO
  147  IF ( CODE(I) )  150, 150, 151
C      CLAMPED
  151  SIXB = 118.44 * FLEXR * WO/( A1* THICK ) ** 2
       SIYB = ABS * SIXB
      IF(BA.LT.1.) GO TO  431
       PHIL = 192.0 + 51.0 * EXPF(-2.5*(BA-1))
       PHIS =  243. *EXPF(-.59*(BA-1))
       GO TO  432
  431  PHIL =  243.
```

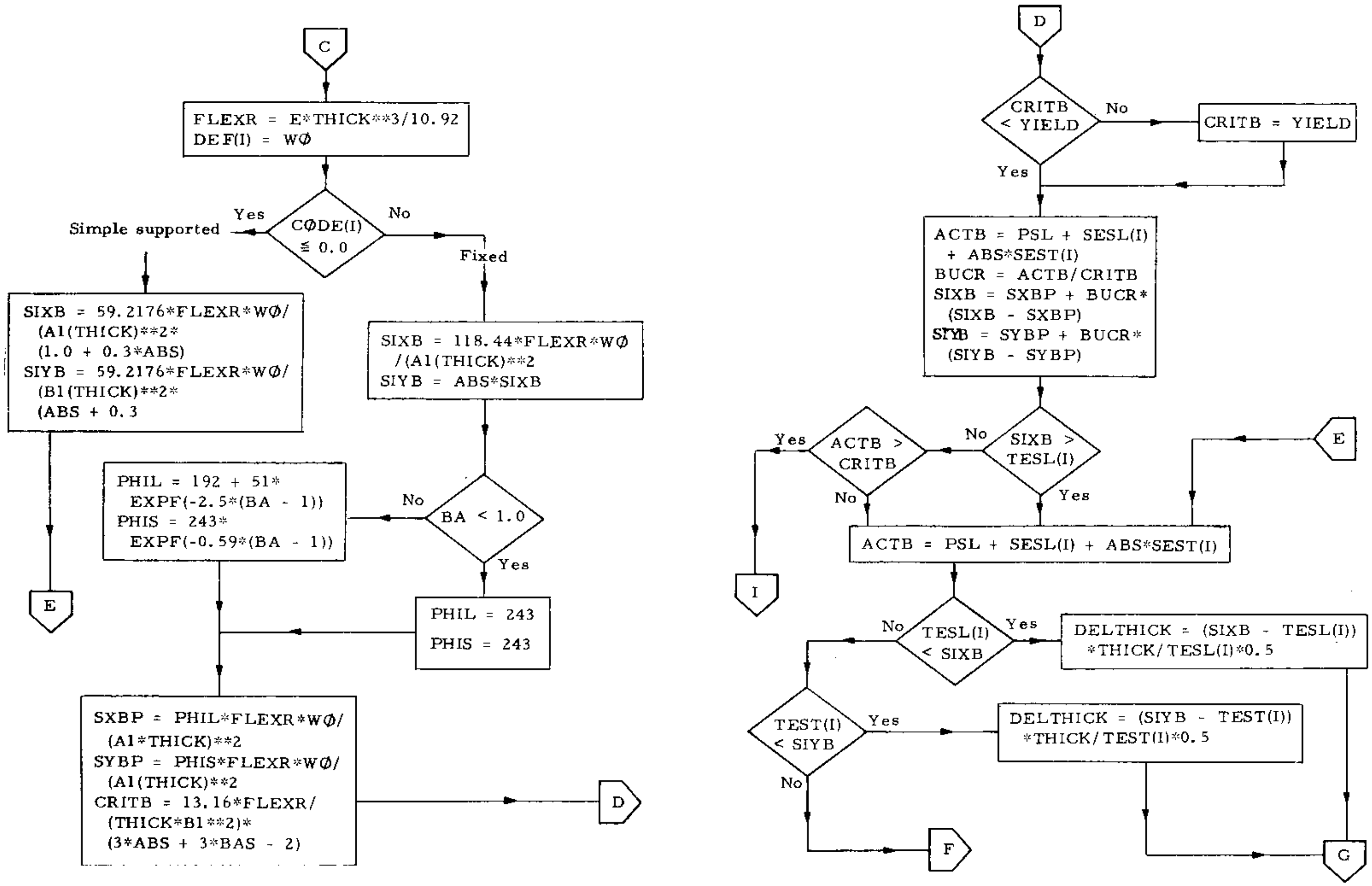
```
      PHIS =  243.
 432  SXBP = PHIL * FLEXR * WO/(A1*THICK) ** 2
      SYBP = PHIS  * FLEXR * WO/( A1* THICK) ** 2
      CRITB = 13.16 * FLEXR/(THICK *B1** 2)*(3.0*ABS + 3.0 * BAS-2.)
      IF(CRITB - YIELD )8181,8182,8182
 8182 CRITB = YIELD
 8181 ACTB = PSL + SESL(I) + ABS *SEST(I)
      BUCR = ACTB/CRITB
      SIXB = SXBP + BUCR * ( SIXB - SXBP )
      SIYB = SYBP + BUCR * ( SIYB - SYBP)
      IF(SIXB.GT.TESL(I))       GO TO 152
      IF(ACTB.GT.CRITB)         GO TO 159
      GO TO 152
C     SIMPLE SUPPORTED
 150  SIXB=  59.2176 * FLEXR * WO / (A1*THICK ) ** 2* (1.0+0.3*ABS)
      SIYB = 59.2176 * FLEXR * WO / (B1*THICK) **2 * ( ABS + 0.3 )
  152 ACTB=PSL+SESL(I)+ABS*SEST(I)
      IF(TESL(I)-SIXB)440,153,153
 153  IF( TEST(I) - SIYB) 441, 154, 154
 154    IF ( (TESL(I) - SIXB ) - ( TEST(I) - SIYB ) ) 155, 155, 156
 155   IF ((TESL(I) - SIXB ) - 0.04 * TESL(I) ) 159,159, 442
 156   IF (( TEST(I) - SIYB ) - 0.04 * TEST(I) ) 159,159,443
 442  DELTHICK = (SIXB - TESL(I) ) * THICK/TESL(I)  *0.25
      GO TO 157
 443  DELTHICK = (SIYB - TEST(I) ) * THICK/TEST(I)     *0.25
        GO TO 157
 440  DELTHICK = (SIXB - TESL(I) ) * THICK/TESL(I)  *0.5
      GO TO 157
 441  DELTHICK = (SIYB - TEST(I) ) * THICK/TEST(I) * 0.5
  157 THICK=THICK+DELTHICK
      IF(THICK.GT.0.1)   GO TO 158
      THICK=.25
      PRINT 959,I
 959  FORMAT(/,* THICK= .25 FOR PLATE I = *,I3,/)
  158 CONTINUE
      IF(CODE(I).EQ. 1.0) GO TO 421
       CRITB = 9.87 * FLEXR * ( AB + BA ) **2 /(THICK * B1** 2)
      IF((SIXB-OSIXB)/(THICK-OTHICK)) 421,421,159
  421 THIKK(I)= THICK
      IF(ABSF(DELTHICK).LE.0.009) GO TO 159
      KXPL = 1 +KXPL
      OSIXB = SIXB
      OTHICK = THICK
      IF(KXPL-KXPLM)112,112,159
 159   FLEXR = E * THICK ** 3.0/10.92
      THIKK(I) = THICK
 1003 FORMAT(1H0,7F11.2)
      IF (CODE(I).EQ.1) GO TO 165
       ACTB = ( PSL + SESL(I) ) + ABS * SEST(I)
       CRITB = 9.87 * FLEXR * ( AB + BA ) **2 /(THICK * B1** 2)
      GO TO 502
  165  CRITB = 13.16 * FLEXR/(THICK*B1 ** 2)*(3.0*ABS + 3.0 * BAS+2.)
 502  CONTINUE
      BUCR=ACTB/CRITB
      IF(BUCR.LT.1.0) GO TO 160
 605   FORMAT(I5)
      IF(CRITB.GT.YIELD)GO TO 501
      GO TO 163
 501  CRITB = YIELD
 163  BUCR = ACTB / CRITB
      KXXPL = KXXPL + 1
      IF (KXXPL.EQ.KXXPLM) GO TO 160
 164   IF ( BUCR - 1.0 ) 160,161,161
  161 THICK = THICK + .02
      THIKK(I)=THICK
      GO TO 159
 160   THIKK(I) = THICK
       IF ( CODE(I) ) 410, 400,400
C      EFFECTIVE WIDTH
C     FIXED
 400  CONI=9.0/256.0*PIE**4
      GO TO 420
  410 CONI=PIE**4/16.0
 420   CONII = CONI * ( 1.0 + 0.3 * ABS ) * BA
      CBX  = CRITB / ( PSL + SESL(I))
      BBX=ABS*SEST(I)/(PSL+SESL(I))
       EFFW(I) = B1/ ( 1.0 + CONII / CONST * ( 1.0 + BBX-CBX))
      IF(1+BBX.LT.CBX)     EFFW(I)=B1
      RETURN
      END
```

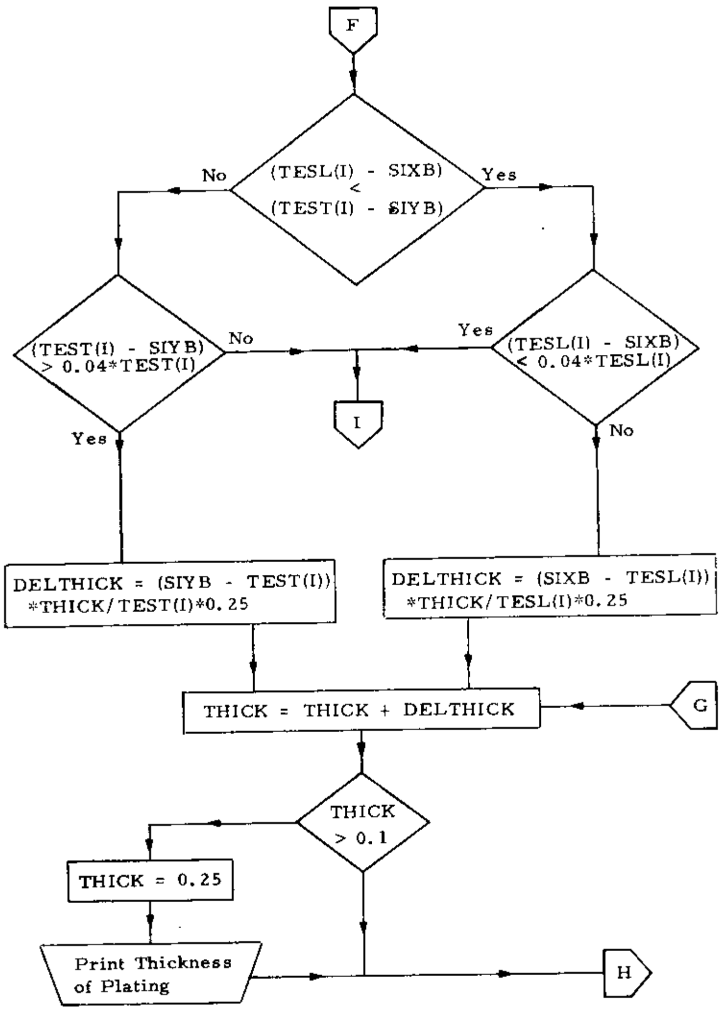
Subroutine PLATING (A1, B1, I, J)

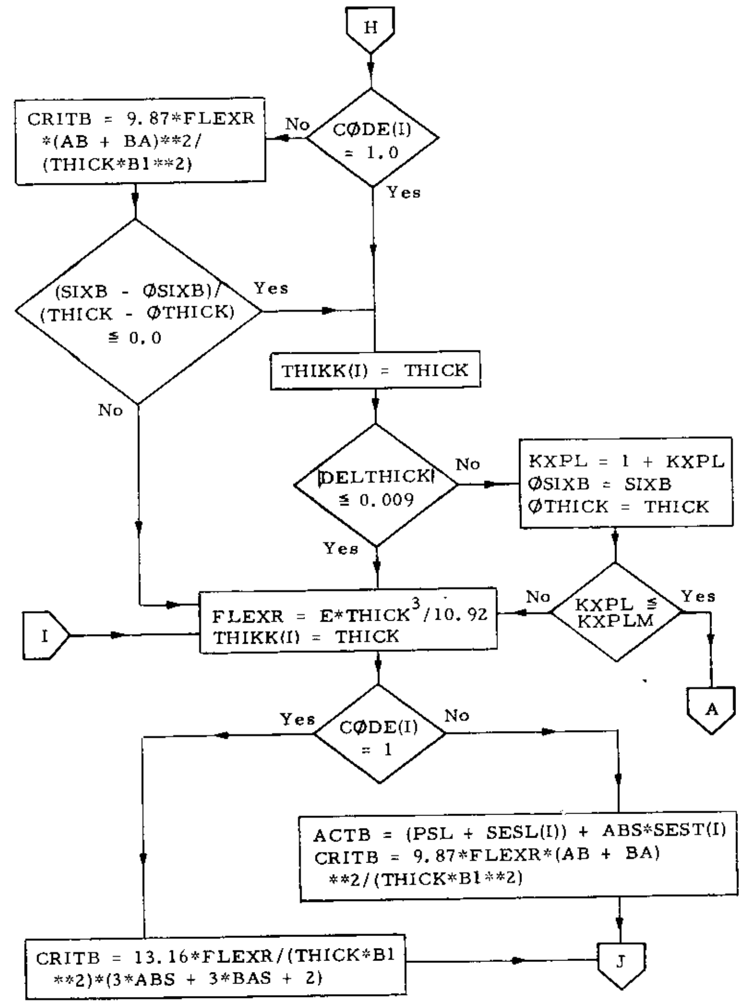


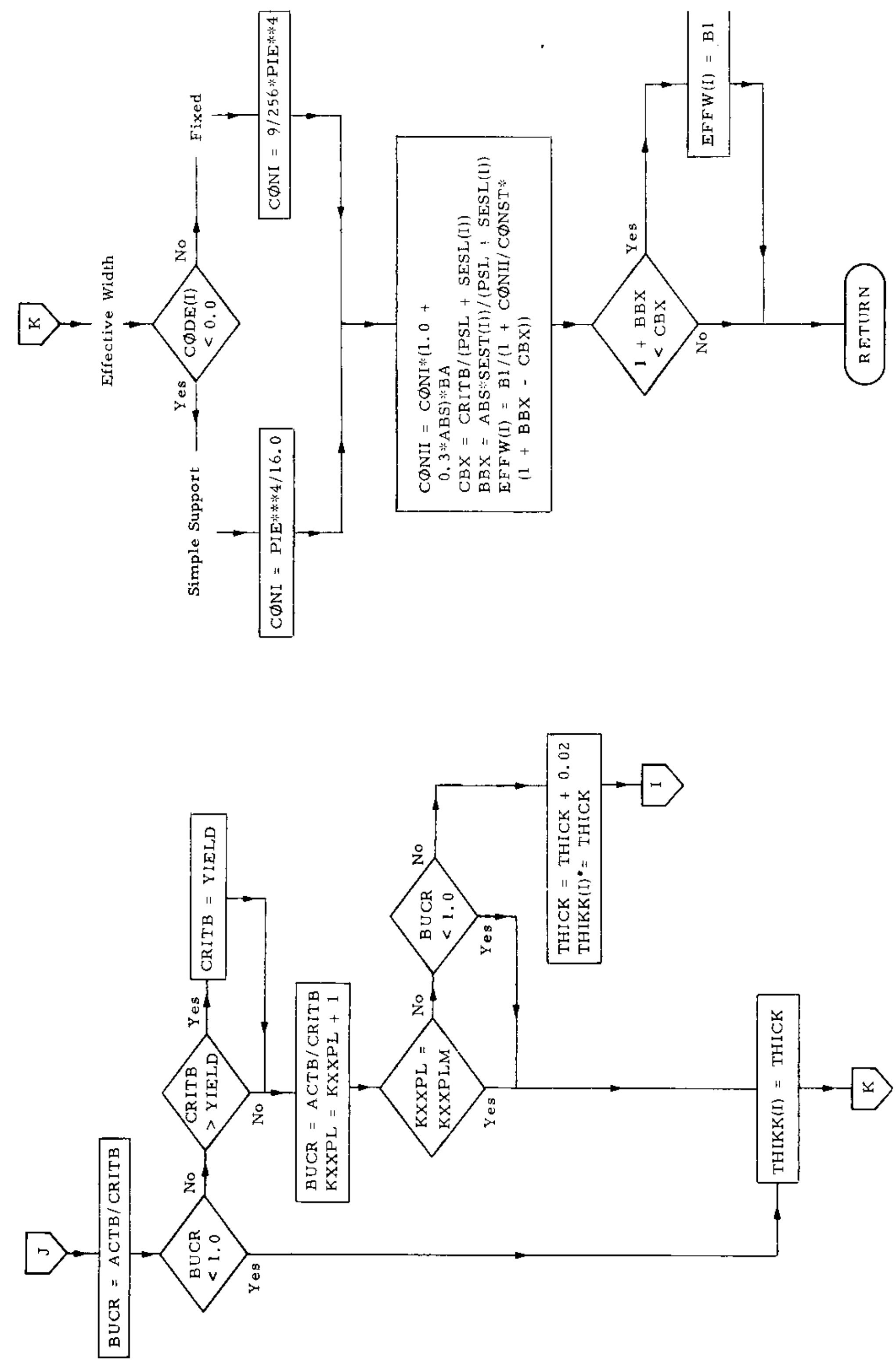



B
WØ = WØI
KBEN = 1
NB = 0
BEN = FWØ(WØ)
NB = NB + 1
NB > 25
Yes
No
Print
"NO CONVERGENCE"
ABSF(BEN)
$\le 10^{-5}$
Yes
No
C
KBEN = 1
2
3
WØ = WØII
QA = BEN
KBEN = 2
QB = BEN
WØ = (WØI + WØII)*0.5
KBEN = 3
ABSF(BEN)
$\le 10^{-5}$
Yes
No
QB*BEN
≤ 0.0
Yes
No
WØI = WØ
WØ = (WØI + WØII)*0.5
QA = BEN
WØII = WØ
WØ = (WØI + WØII)*0.5
QB = BEN


C
FLEXR = E*THICK**3/10.92
DEF(I) = WØ
CØDE(I) ≦ 0.0
Yes
Simple supported
No
Fixed
SIXB = 59.2176*FLEXR*WØ/
(A1(THICK)**2*
(1.0 + 0.3*ABS)
SIYB = 59.2176*FLEXR*WØ/
(B1(THICK)**2*
(ABS + 0.3
SIXB = 118.44*FLEXR*WØ
/(A1(THICK)**2
SIYB = ABS*SIXB
E
BA < 1.0
No
Yes
PHIL = 192 + 51*
EXPF(-2.5*(BA - 1))
PHIS = 243*
EXPF(-0.59*(BA - 1))
PHIL = 243
PHIS = 243
SXBP = PHIL*FLEXR*WØ/
(A1*THICK)**2
SYBP = PHIS*FLEXR*WØ/
(A1(THICK)**2
CRITB = 13.16*FLEXR/
(THICK*B1**2)*
(3*ABS + 3*BAS - 2)
D
D
CRITB < YIELD
No
CRITB = YIELD
Yes
ACTB = PSL + SESL(I)
+ ABS*SEST(I)
BUCR = ACTB/CRITB
SIXB = SXBP + BUCR*
(SIXB - SXBP)
SIYB = SYBP + BUCR*
(SIYB - SYBP)
SIXB > TESL(I)
No
Yes
ACTB > CRITB
Yes
No
E
I
ACTB = PSL + SESL(I) + ABS*SEST(I)
TESL(I) < SIXB
No
Yes
DELTHICK = (SIXB - TESL(I))
*THICK/TESL(I)*0.5
TEST(I) < SIYB
Yes
No
DELTHICK = (SIYB - TEST(I))
*THICK/TEST(I)*0.5
F
G


F
(TESL(I) - SIXB) < (TEST(I) - SIYB)
No
Yes
(TEST(I) - SIYB) > 0.04*TEST(I)
No
Yes
(TESL(I) - SIXB) < 0.04*TESL(I)
Yes
No
I
DELTHICK = (SIYB - TEST(I)) *THICK/TEST(I)*0.25
DELTHICK = (SIXB - TESL(I)) *THICK/TESL(I)*0.25
THICK = THICK + DELTHICK
G
THICK > 0.1
THICK = 0.25
Print Thickness of Plating
H


H
CØDE(I) = 1.0
No
Yes
CRITB = 9.87*FLEXR *(AB + BA)**2/ (THICK*B1**2)
(SIXB - ØSIXB)/ (THICK - ØTHICK) ≦ 0.0
Yes
No
THIKK(I) = THICK
|DELTHICK| ≦ 0.009
No
Yes
KXPL = 1 + KXPL
ØSIXB = SIXB
ØTHICK = THICK
KXPL ≦ KXPLM
No
Yes
A
I
FLEXR = E*THICK3/10.92
THIKK(I) = THICK
CØDE(I) = 1
Yes
No
ACTB = (PSL + SESL(I)) + ABS*SEST(I)
CRITB = 9.87*FLEXR*(AB + BA) **2/(THICK*B1**2)
CRITB = 13.16*FLEXR/(THICK*B1 **2)*(3*ABS + 3*BAS + 2)
J



J
BUCR = ACTB/CRITB
BUCR
< 1.0
Yes
No
CRITB
> YIELD
Yes
No
CRITB = YIELD
BUCR = ACTB/CRITB
KXXPL = KXXPL + 1
KXXPL =
KXXPLM
Yes
No
BUCR
< 1.0
Yes
No
THICK = THICK + 0.02
THIKK(I) = THICK
I
THIKK(I) = THICK
K
K
Effective Width
CØDE(I)
< 0.0
Yes
No
Simple Support
Fixed
CØNI = PIE**4/16.0
CØNI = 9/256*PIE**4
CØNII = CØNI*(1.0 +
0.3*ABS)*BA
CBX = CRITB/(PSL + SESL(I))
BBX = ABS*SEST(I)/(PSL + SESL(I))
EFFW(I) = B1/(1 + CØNII/CØNST*
(1 + BBX - CBX))
1 + BBX
< CBX
Yes
No
EFFW(I) = B1
RETURN

17. Subroutine REACT

a) Abstract:

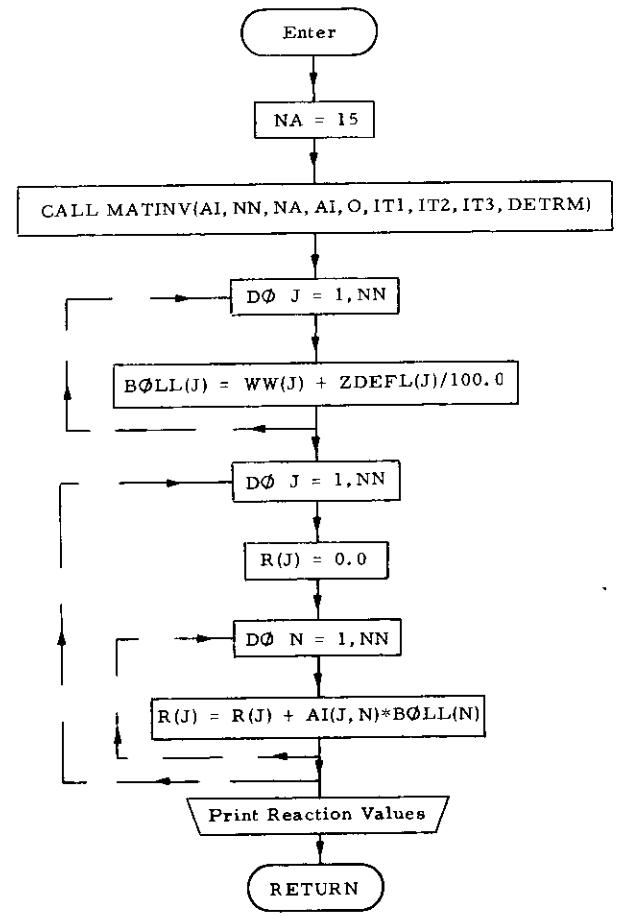
This subroutine calculates the reactions at the intersections of frames and longitudinal girders. The reactions are obtained by application of the influence matrix to the difference of the released frame deflections obtained by subroutine FRAME and the grillage deflections obtained by subroutine STEP.

b) Terms specific to this subroutine:

FORTRAN Term	Definition
AI	Array of influence coefficients
BØLL	Difference between released frame deflection and grillage deflection
NN	Number of longitudinals
R	Reaction
WW	Unrestrained deflection
ZDEFL	Restrained deflection

```
      SUBROUTINE REACT(NN,AI,R,WW,ZDEFL)
     DIMENSION R(9),AI(15,15),BOLL(9 ),ZDEFL(9),WW(9)
     DIMENSION IT1(9),IT2(9),IT3(9,2)
      NA=15
     CALL MATINV (AI,NN,NA,AI,0,IT1,IT2,IT3,DETRM)
     DO 1000  J = 1,NN
     BOLL(J) = WW(J)+ZDEFL(J) / 100,0
1000 CONTINUE
     DO 1800  J = 1,NN
     R(J) = 0,0
     DO 1800  N = 1,NN
     R(J) =  R(J) + AI(J,N)*BOLL(N)
1800 CONTINUE
     PRINT 4015,(R(I),I=1,NN)
4015 FORMAT (1H0, * REACTIONS     *,/9E13,5)
     RETURN
     END
```

Subroutine REACT (NN, AI, R, WW, ZDEFL)



18. Subroutine RTPLSUB

a) Abstract:

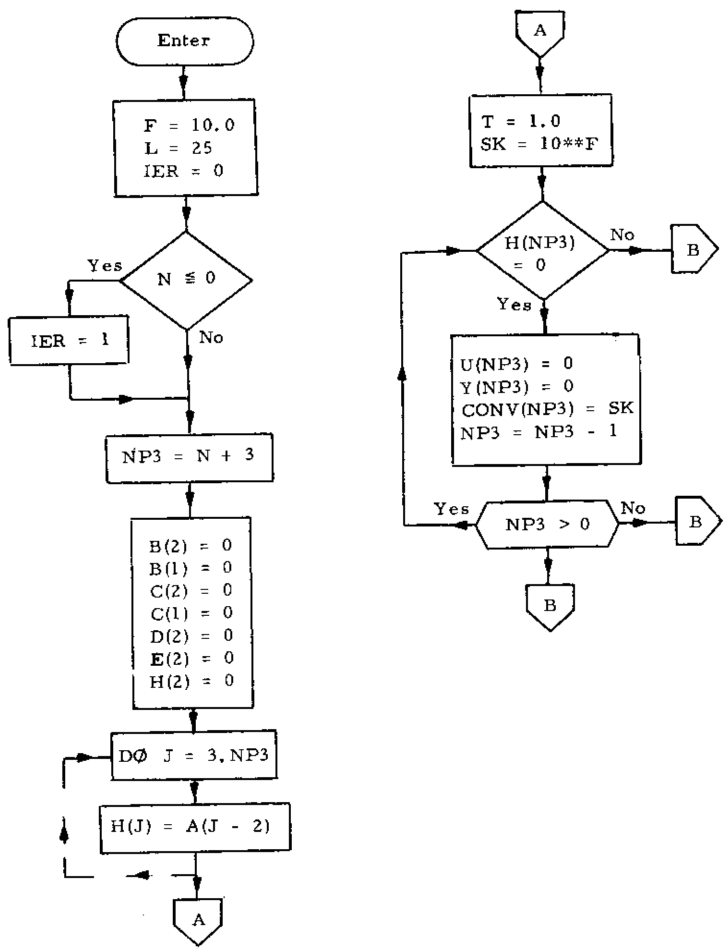
Subroutine RTPLSUB calculates the roots of the characteristic equation.

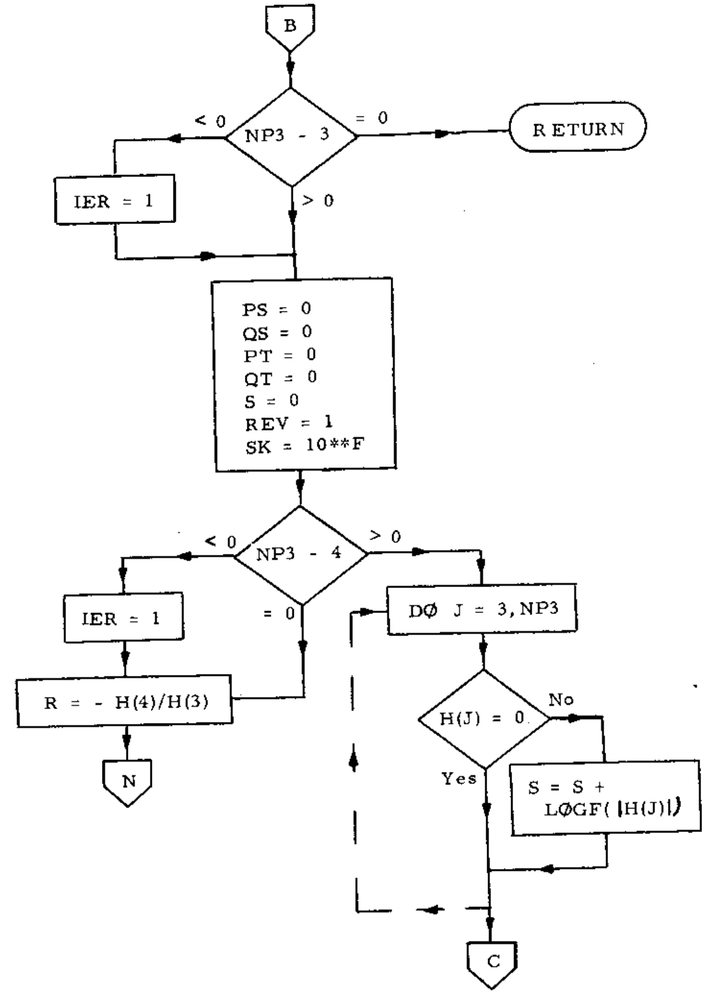
b) Terms specific to this subroutine:

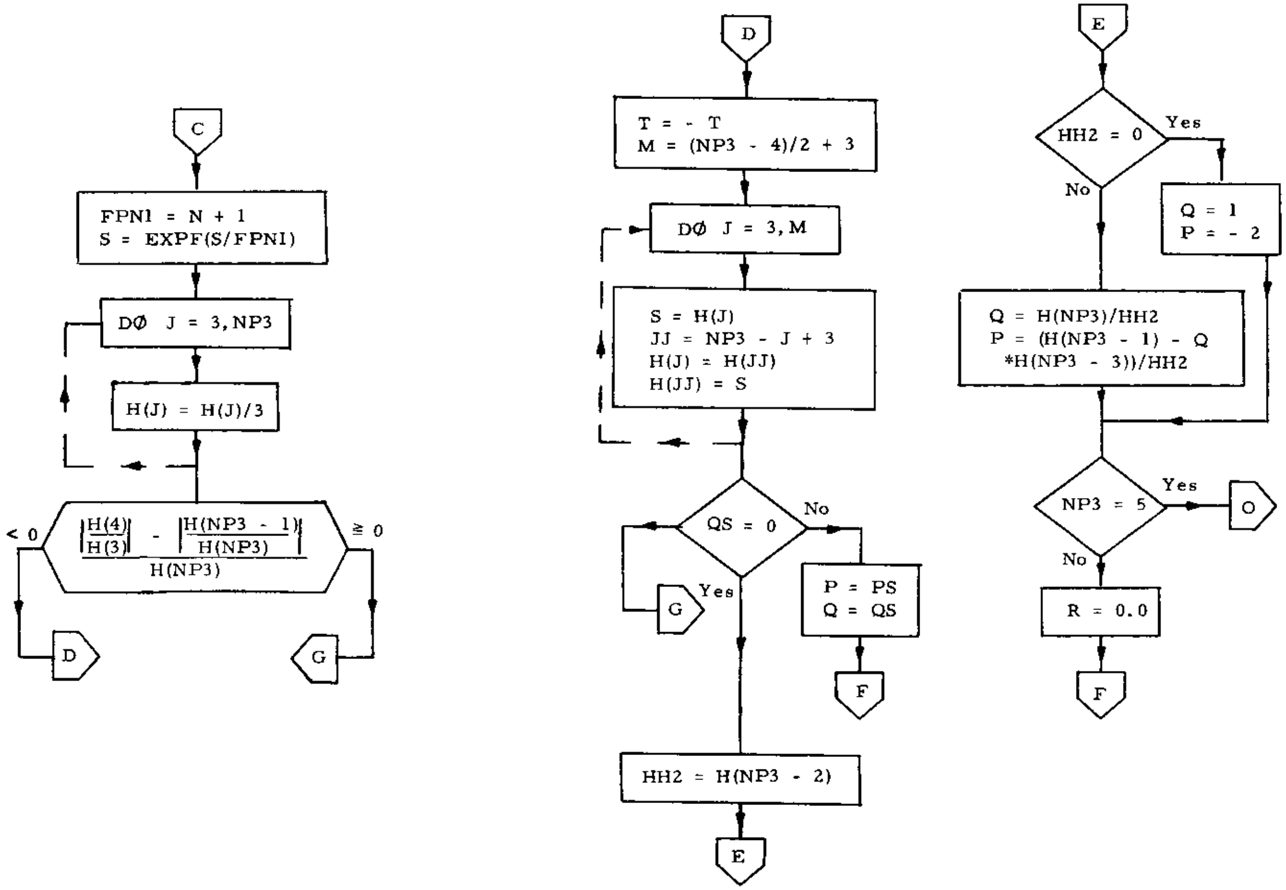
FORTRAN Term	Definition
A	Input array of coefficients
CONV	Degree of convergence
IER	Error code
N	Order of matrix
U	Output array of real roots
V	Output array of imaginary roots

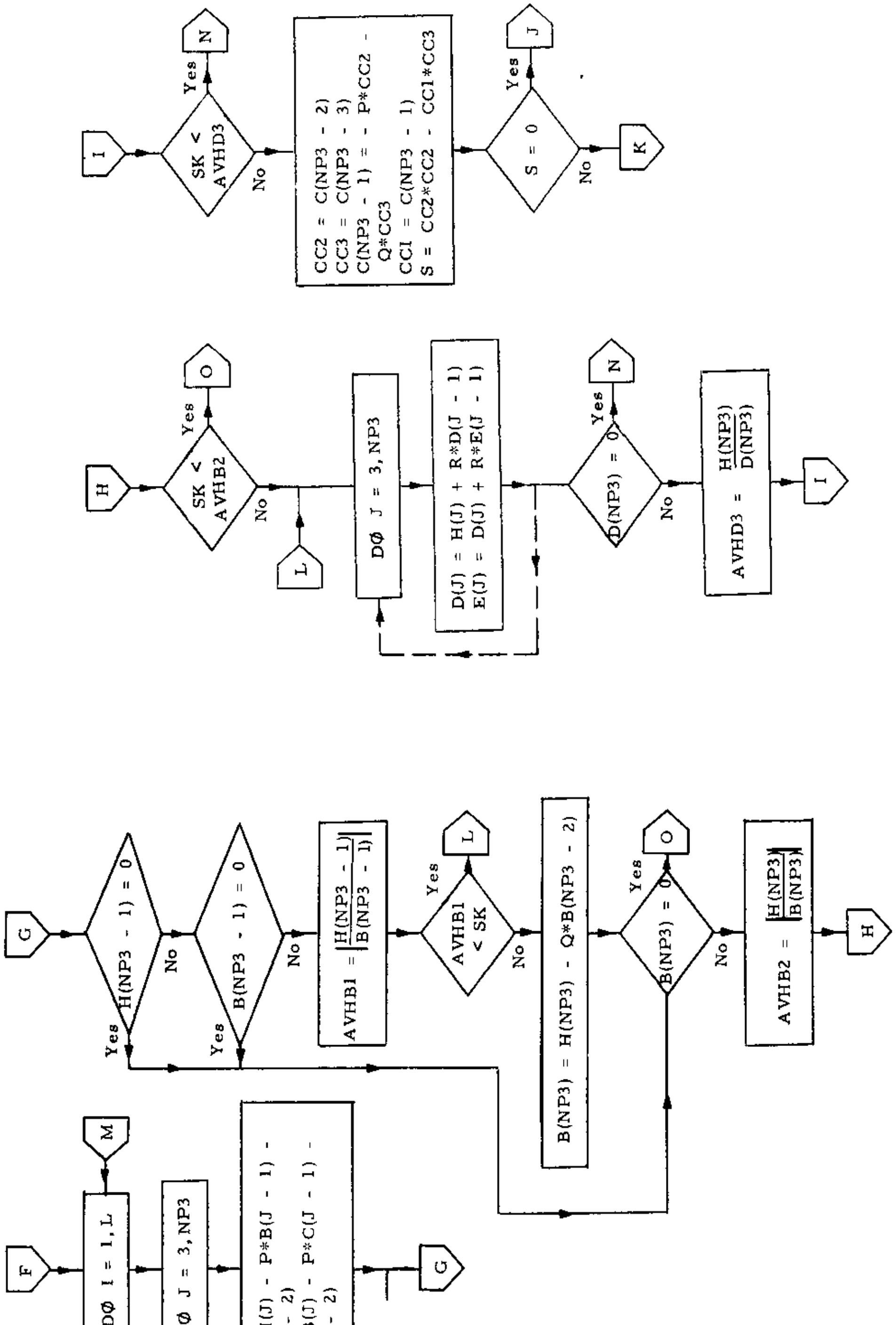
```
      SUBROUTINE RTPLSUB(N,A,U,V,CONV,IER)
      DIMENSION A(10),H(50),B(50),C(50),D(50),E(50),U( 9),V( 9),CONV(50)
      F=10,0
      L=25
      IER=0
      IF(N) 54,54,52
   54 IER=1
   52 NP3=N+3
  100 B(2)=0,0
      B(1)=0,0
      C(2)=0,0
      C(1)=0,0
      D(2)=0,0
      E(2)=0,0
      H(2)=0,0
      DO 101 J=3,NP3
  101 H(J)=A(J-2)
      T=1,0
      SK=10,0**F
  150 IF(H(NP3)) 200,151,200
  151 U(NP3)=0,0
      V(NP3)=0,0
      CONV(NP3)=SK
      NP3=NP3-1
      IF(NP3)152,152,150
  152 IER=1
  200 IF(NP3-3)205,51,201
  205 IER=1
  201 PS=0,0
      QS=0,0
      PT=0,0
      QT=0,0
      S=0,0
      REV=1,0
      SK=10,0**F
      IF(NP3-4)206,202,203
  206 IER=1
  202 R=-H(4)/H(3)
      GO TO 500
  203 DO 207 J=3,NP3
      IF(H(J))204,207,204
  204 S=S+LOGF(ABSF(H(J)))
  207 CONTINUE
      FPN1=N+1
      S=EXPF(S/FPN1)
      DO 208 J=3,NP3
  208 H(J)=H(J)/S
  210 IF(ABSF(H(4)/H(3))-ABSF(H(NP3-1)/H(NP3)))250,252,252
  250 T=-T
      M=(NP3-4)/2 + 3
      DO 251 J=3,M
      S=H(J)
      JJ=NP3-J+3
      H(J)=H(JJ)
  251 H(JJ)=S
  252 IF(QS) 253,254,253
  253 P=PS
      Q=QS
      GO TO 300
  254 HH2=H(NP3-2)
      IF(HH2) 256,255,256
  255 Q=1,0
      P=-2,0
      GO TO 257
  256 Q=H(NP3)/HH2
      P=(H(NP3-1)-Q*H(NP3-3))/HH2
  257 IF(NP3-5)258,550,258
  258 R=0,0
  300 DO 490 I=1,L
  350 DO 351 J=3,NP3
      B(J)=H(J)-P*B(J-1)-Q*B(J-2)
  351 C(J)=B(J)-P*C(J-1)-Q*C(J-2)
      IF(H(NP3-1))352,400,352
  352 IF(B(NP3-1))353,400,353
  353 AVHB1=ABSF(H(NP3-1)/B(NP3-1))
  356 IF(AVHB1-SK)450,354,354
  354 B(NP3)=H(NP3)-Q*B(NP3-2)
  400 IF(B(NP3))401,550,401
  401 AVHB2=ABSF(H(NP3)/B(NP3))
  403 IF(SK-AVHB2)550,450,450
  450 DO 451 J=3,NP3
      D(J)=H(J)+R*D(J-1)
  451 E(J)=D(J)+R*E(J-1)
      IF(D(NP3))452,500,452
  452 AVHD3=ABSF(H(NP3)/D(NP3))
  460 IF(SK-AVHD3)500,453,453
  453 CC2=C(NP3-2)
      CC3=C(NP3-3)
      C(NP3-1)=-P*CC2-Q*CC3
      CC1=C(NP3-1)
      S=CC2*CC2-CC1*CC3
      IF(S)455,454,455
  454 P=P-2,0
      Q=Q*(Q+1,0)
      GO TO 456
  455 P=P+(B(NP3-1)*CC2-B(NP3)*CC3)/S
      Q=Q+(-B(NP3-1)*CC1+B(NP3)*CC2)/S
  456 IF(E(NP3-1))458,457,458
  457 R=R-1,0
      GO TO 490
  458 R=R-D(NP3)/E(NP3-1)
  490 CONTINUE
      PS=PT
      QS=QT
      PT=P
      QT=Q
      IF(REV)491,492,492
  491 SK=SK/10,0
  492 REV=-REV
      GO TO 250
  500 IF(T)501,502,502
  501 R=1,0/R
  502 NP=NP3-3
      U(NP)=R
      V(NP)=0,0
      CONV(NP)=SK
      NP3=NP3-1
      DO 503 J=3,NP3
  503 H(J)=D(J)
      IF(NP3-3)300,51,300
  550 IF(T)551,552,552
  551 P=P/Q
      Q=1,0/Q
  552 PP2=P/2,0
      QMPSQ=Q-PP2*PP2
  560 IF(QMPSQ)554,554,553
  553 NP=NP3-3
      U(NP)=-PP2
      U(NP-1)=-PP2
      S=SQRTF(QMPSQ)
      V(NP)=S
      V(NP-1)=-S
      GO TO 561
  554 S=SQRTF(-QMPSQ)
      NP=NP3-3
      IF(P)555,556,556
  555 U(NP)=-PP2+S
      GO TO 557
  556 U(NP)=-PP2-S
  557 U(NP-1)=Q/U(NP)
      V(NP)=0,0
      V(NP-1)=0,0
  561 CONV(NP)=SK
      CONV(NP-1)=SK
      NP3=NP3-2
      DO 558 J=3,NP3
  558 H(J)=B(J)
      GO TO 200
   51 RETURN
      END
```

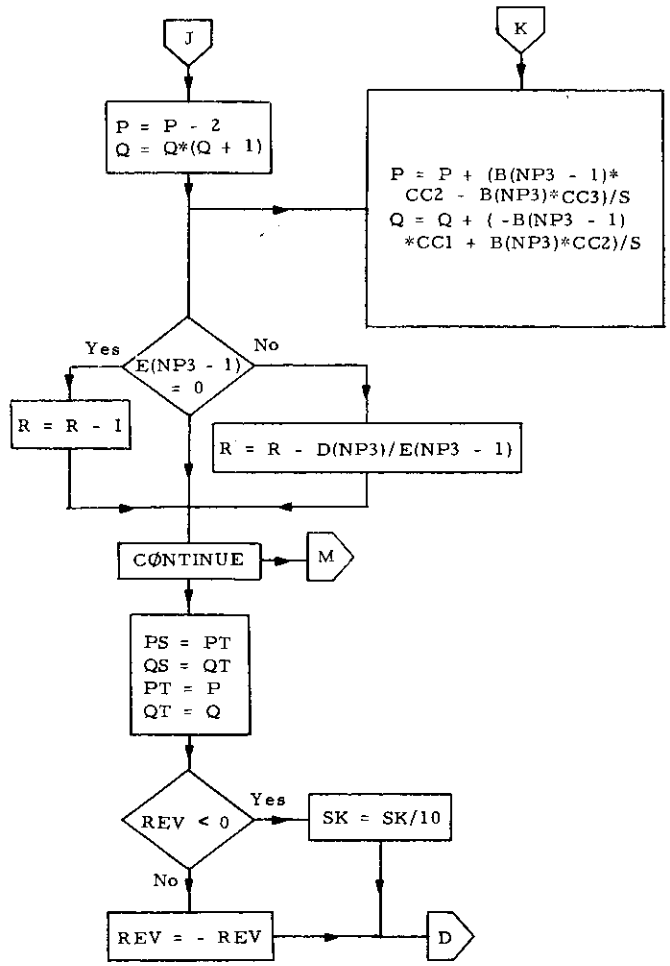
Subroutine RTPLSUB (N, A, U, V, CONV, IER)

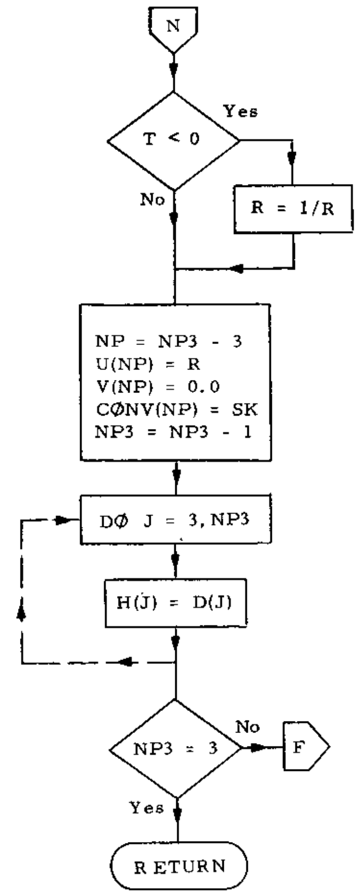


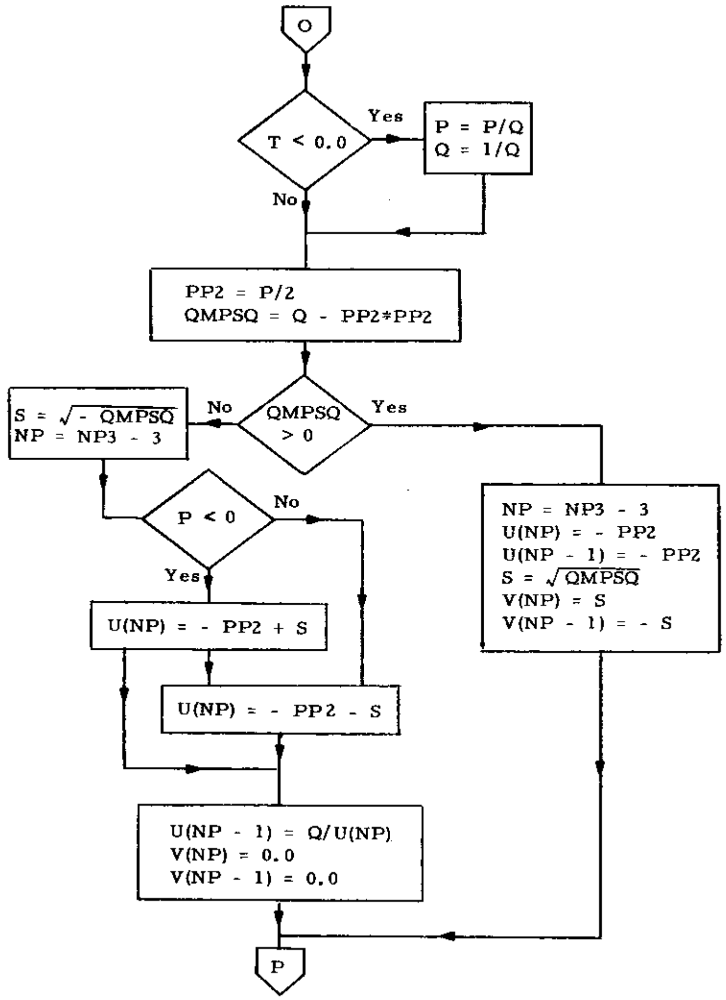


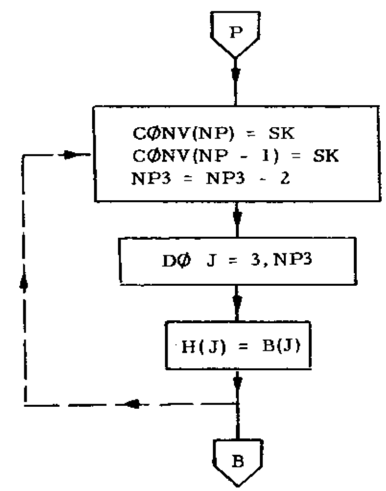

C
FPN1 = N + 1
S = EXPF(S/FPN1)
DØ J = 3,NP3
H(J) = H(J)/3
|H(4)/H(3)| - |H(NP3 - 1)/H(NP3)| / H(NP3)
< 0
≧ 0
D
G
D
T = - T
M = (NP3 - 4)/2 + 3
DØ J = 3,M
S = H(J)
JJ = NP3 - J + 3
H(J) = H(JJ)
H(JJ) = S
QS = 0
Yes
No
G
P = PS
Q = QS
F
HH2 = H(NP3 - 2)
E
E
HH2 = 0
Yes
No
Q = 1
P = - 2
Q = H(NP3)/HH2
P = (H(NP3 - 1) - Q
*H(NP3 - 3))/HH2
NP3 = 5
Yes
No
O
R = 0.0
F


I
SK < AVHD3
Yes
N
No
CC2 = C(NP3 - 2)
CC3 = C(NP3 - 3)
C(NP3 - 1) = - P*CC2 - Q*CC3
CC1 = C(NP3 - 1)
S = CC2*CC2 - CC1*CC3
S = 0
Yes
J
No
K
H
SK < AVHB2
Yes
O
No
L
DØ J = 3, NP3
D(J) = H(J) + R*D(J - 1)
E(J) = D(J) + R*E(J - 1)
D(NP3) = 0
Yes
N
No
AVHD3 = H(NP3)/D(NP3)
I
G
H(NP3 - 1) = 0
Yes
No
B(NP3 - 1) = 0
Yes
No
AVHB1 = |H(NP3 - 1)/B(NP3 - 1)|
AVHB1 < SK
Yes
L
No
B(NP3) = H(NP3) - Q*B(NP3 - 2)
B(NP3) = 0
Yes
O
No
AVHB2 = |H(NP3)/B(NP3)|
H
F
M
DØ I = 1, L
G


J
K
P = P - 2
Q = Q*(Q + 1)
P = P + (B(NP3 - 1)*
CC2 - B(NP3)*CC3)/S
Q = Q + (-B(NP3 - 1)
*CC1 + B(NP3)*CC2)/S
Yes
No
E(NP3 - 1)
= 0
R = R - I
R = R - D(NP3)/E(NP3 - 1)
CØNTINUE
M
PS = PT
QS = QT
PT = P
QT = Q
REV < 0
Yes
SK = SK/10
No
REV = - REV
D


N
T < 0
Yes
No
R = 1/R
NP = NP3 - 3
U(NP) = R
V(NP) = 0.0
CØNV(NP) = SK
NP3 = NP3 - 1
DØ J = 3, NP3
H(J) = D(J)
NP3 = 3
No
F
Yes
RETURN


O
T < 0.0
Yes
No
P = P/Q
Q = 1/Q
PP2 = P/2
QMPSQ = Q - PP2*PP2
QMPSQ
> 0
No
Yes
S = √- QMPSQ
NP = NP3 - 3
P < 0
No
Yes
U(NP) = - PP2 + S
U(NP) = - PP2 - S
U(NP - 1) = Q/U(NP)
V(NP) = 0.0
V(NP - 1) = 0.0
NP = NP3 - 3
U(NP) = - PP2
U(NP - 1) = - PP2
S = √QMPSQ
V(NP) = S
V(NP - 1) = - S
P


P
CØNV(NP) = SK
CØNV(NP - 1) = SK
NP3 = NP3 - 2
DØ J = 3, NP3
H(J) = B(J)
B

19. Subroutine SECTION

a) Abstract:

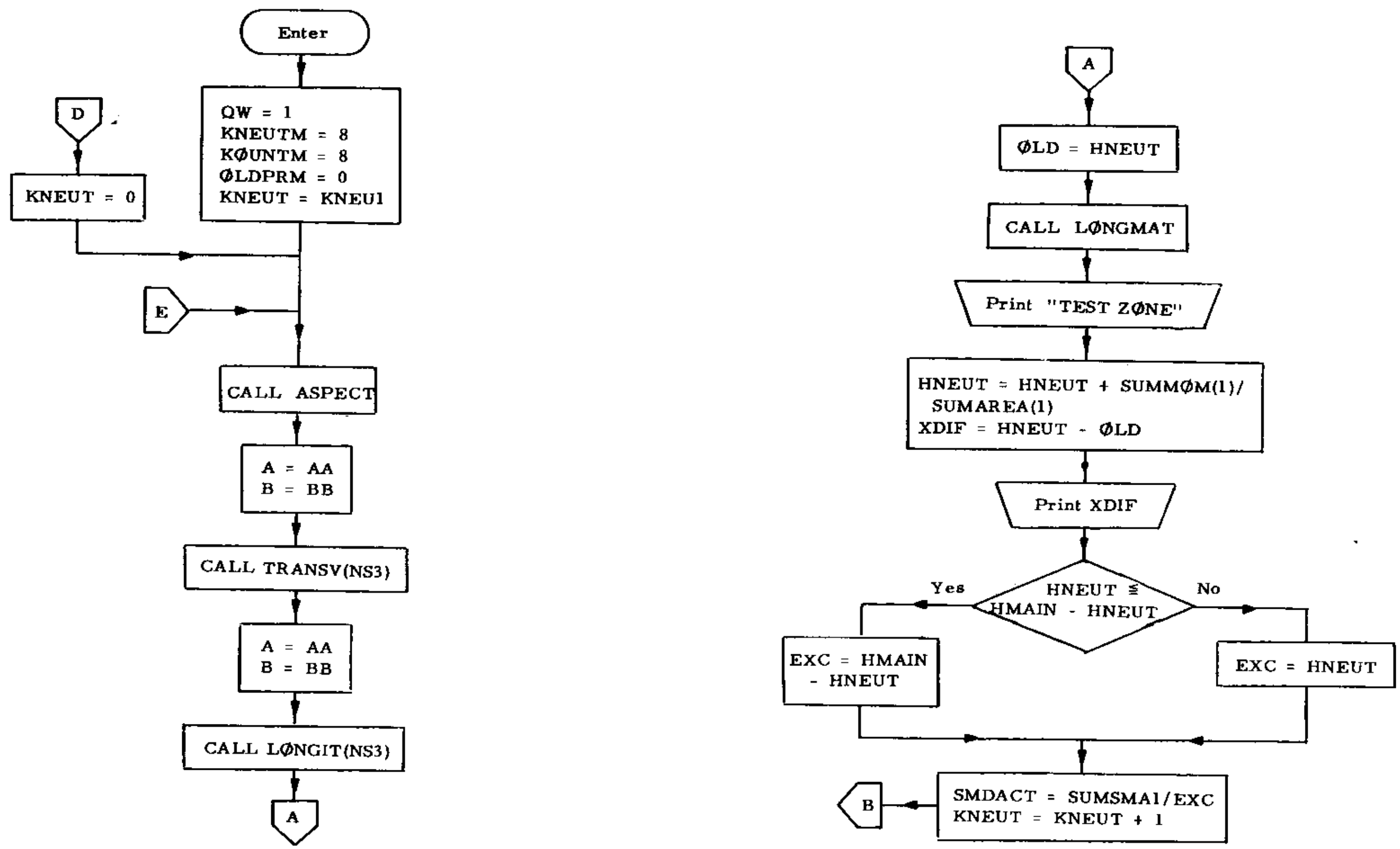
Subroutine SECTION is called from program TRANSHIP. It calculates the section modulus of the midship section. The neutral axis and section modulus are solved by iteration, the first starting from an assumed value (based on empirical data). To accomplish this, the subroutine makes use of subroutines ASPECT, TRANSV, LONGIT and LONGMAT.

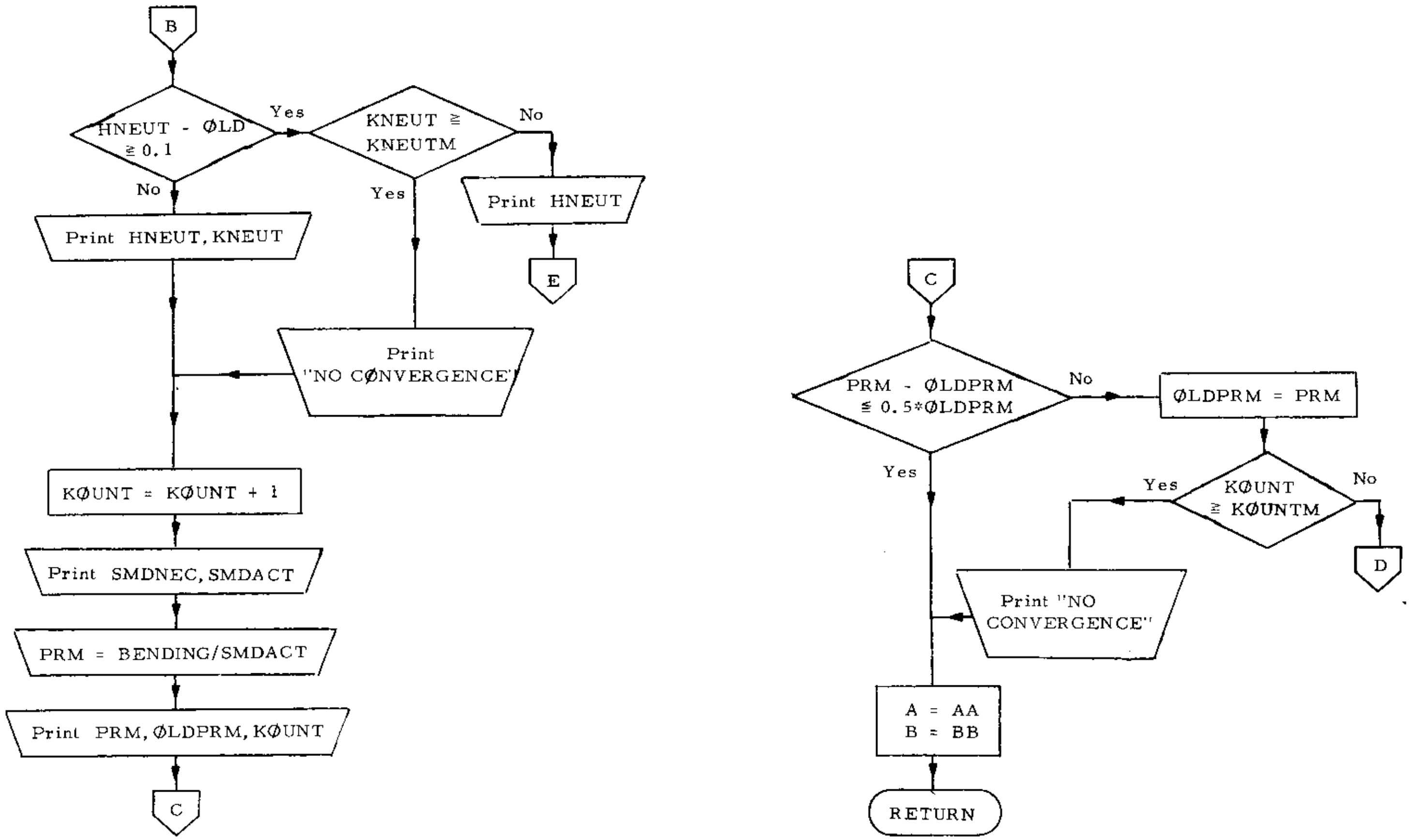
b) Terms specific to this subroutine:

FORTRAN Term	Definition
KNEUT	Number of iterations in determination of neutral axis of ship section
KOUNT	Number of iterations in determination of correct ship section modulus
SMDNEC	Necessary section modulus

```
      SUBROUTINE SECTION (KNEU1, SMDNEC,KOUNT,BENDING,NX2)
      COMMON/DE/ DEF(20)
      COMMON/DK/ DKLO(10)
      COMMON/LO/FSECLO(12),FSESL(17),FSESL1(17),SMODL2(17,2,3),PSLL(17)
      COMMON/TR/ FSECTR(17),FSEST(17),FSEST1(17),SMODT2(17,2,3),SMODT3(
     1 17,2,5)
      COMMON /B/ A,B,NDECKS,IPLATE,PSL,PRM,HMAIN,HNEUT,TESL,TEST,SESL,
     1     SEST,        KPANELS,BEAM,XNG,HATCH,THIKK,THIKK1,EFFBR,EFFB1,
     2     EFFW,EFFW1,HFLOOR,TWEENH,THIKX,EFBR,WEB,WEB1,XIPFX,CHI,
     3     XIS, AREAT1,SMAL1,SMODL1,WEBL1,SUMAREAP,SUMAREAL,SUMMP,SUMML,
     4     AREA,SUMAREA,SUMMOM,WPL,XPANEL,WBAYLP,XNST,NST,ITR,
     5     WTR,BAYTR,SUMWT,SUMWTR,LLGX,  YIELD,WBAYTR,XLABOR,PLCOST,
     6     COST,COSTMIN,GNOT,GNNT,KNEUT,SUMSMAP,SUMSMAL,SUMSMA,CODE,
     7WAVEH,PRHEAD,DRAFT,ALOAD,XLHOLD,XLPANEL,AAA,BBB,AA,BB,DELTESL,
     8      SUMSMA1,ZP,ZL,ILG,AREAL1,ASX,  SMAT1,SMODT1
      DIMENSION ZP(17),SESL(17),SEST(17),TESL(17),TEST(17),XNG(7),THIKK(
     117),       THIKK1(17,2,5),EFFBR(17),EFFW(17),EFFB1(17,2,5),EFFW1(17
     2,2,5),AREA(17,5),TWEENH(5),SMAT1(12,2,5),SMODT1(12,2,5),THIKX(12,2
     3),EFBR(12,2),AREAT1(12,2,5),WEB1(12,2,5),SMAL1(17,2,3),SMODL1(17,2
     4,3),WEBL1(7,2,3),AREAL1(7,2,3),SUMAREAP(5),SUMAREAL(5),SUMMP(5),SU
     5MML(5),SUMAREA(5),SUMMOM(5),WPL(5),XLPANEL(5),SUMWTR(5),WTR(12),ZL
     6(7),TIKK1(17,5),CODE(17),ALOAD(17),SUMSMAP(5),SUMSMAL(5),SUMSMA(5)
 8012 FORMAT(/8E13,5,/)
       OW=1.
      KNEUTM =8
      KOUNTM=8
      OLDPRM=0.
      KNEUT=KNEU1
      GO TO 347
  348  KNEUT=0
  347 CONTINUE
       CALL ASPECT
      A=AA
      B=BB
      CALL TRANSV (NS3)
      A=AA
      B=BB
      CALL LONGIT (NS3)
      OLD = HNEUT
      CALL LONGMAT
      PRINT 9991 ,OW
 9991 FORMAT(*  TEST ZONE *, F10,1/)
      HNEUT = HNEUT + SUMMOM(1)/SUMAREA(1)
      XDIF=HNEUT - OLD
      PRINT 8154,XDIF
      IF(HNEUT.LE.  HMAIN -HNEUT) GO TO 410
      EXC = HNEUT
      GO TO 411
  410 EXC = HMAIN -HNEUT
  411 SMDACT = SUMSMA1/EXC
      KNEUT = KNEUT + 1
      IF(ABSF(HNEUT-OLD) ,GE, 0,1)  GO TO 8152
 8151 PRINT 8153,HNEUT
      PRINT 414, KNEUT
      GO TO 8154
 8152 IF(KNEUT.GE.KNEUTM)GO TO 8155
      PRINT 8153,HNEUT
      GO TO 347
 8155 PRINT 8156
 8156 FORMAT(* NO CONVERGENCE *)
       PRINT 8153, HNEUT
      GO TO 8157
 8154 CONTINUE
 8157 CONTINUE
      KOUNT = KOUNT + 1
      PRINT 415, SMDNEC, SMDACT
      PRM = BENDING/SMDACT
      PRINT 9004, PRM
      PRINT 402 , OLDPRM ,KOUNT
      IF(ABSF(PRM-OLDPRM  ),LE, 0.05* OLDPRM ) GO TO 404
      OLDPRM=PRM
      IF(KOUNT ,GE,KOUNTM) GO TO 403
      GO TO 348
  403 PRINT 8156
  404 CONTINUE
       A = AA
      B = BB
  402 FORMAT (1H0,10X,* OLDPRM  =*,F8,2/ 17X,I2///)
 9004 FORMAT (1H0,10X,* NEWPRM  =*,F8,2/)
  415 FORMAT(1H0,*NECESSARY SECTION MODULUS    =*,F20,0/
     1*ACTUAL SECTION MODULUS       =*,F20,0//)
  414 FORMAT(* KNEUT = *,I2)
 8153 FORMAT(1H0,* HNEUT =*,F10,2)
 9154 FORMAT(1H0,* DELHN =*,F10,2)
      RETURN
      END
```

Subroutine SECTIØN(KNEU1, SMDNEC, KØUNT, BENDING, NXZ)




B
HNEUT - ØLD ≧ 0.1
Yes
No
KNEUT ≧ KNEUTM
Yes
No
Print HNEUT
E
Print HNEUT, KNEUT
Print "NO CØNVERGENCE"
KØUNT = KØUNT + 1
Print SMDNEC, SMDACT
PRM = BENDING/SMDACT
Print PRM, ØLDPRM, KØUNT
C
C
PRM - ØLDPRM ≦ 0.5*ØLDPRM
No
ØLDPRM = PRM
Yes
KØUNT ≧ KØUNTM
Yes
No
D
Print "NO CONVERGENCE"
A = AA
B = BB
RETURN

20. Subroutine SHAPES

a) Abstract:

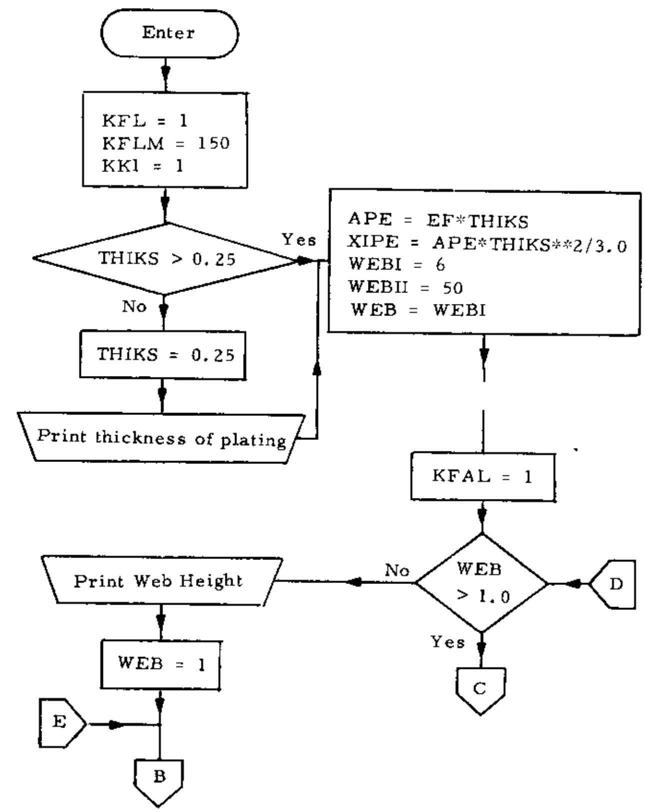
This subroutine is called from both subroutines LONGIT and TRANSV and is used to calculate the scantlings of shapes. Subroutine FSHAPE is called by this subroutine. The section is determined by its section modulus.

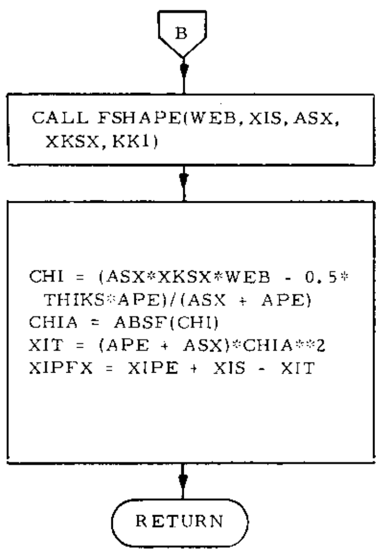
b) Terms specific to this subroutine:

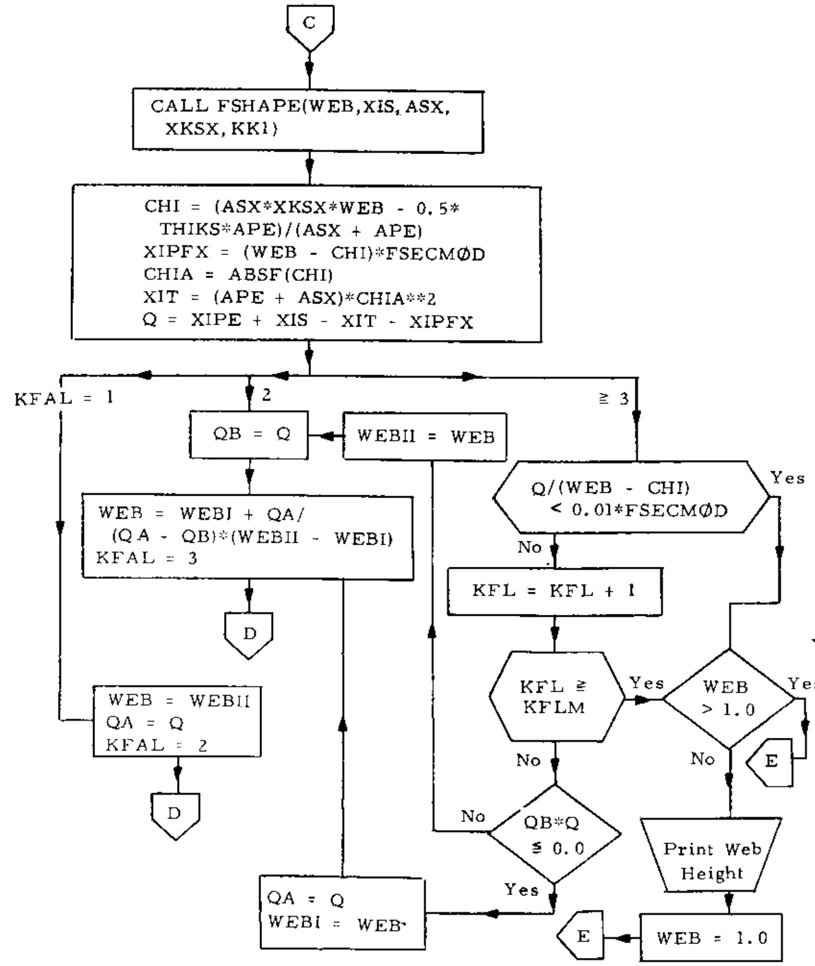
FORTRAN Term	Definition	Mathematical Symbol
APE	Effective area of plating	A_{pe}
CHI	Distance of neutral axis of plate-frame combination from center of plate	χ_{ps}
WEB	Web height	
XIPE	Second central moment of effective plating (about the faying surface)	I_{pe}
XIPFX	Second central moment of plate-frame combination	I_{pf}
XIT	Second central moment of transference	I_t

```
      SUBROUTINE SHAPES  (FSECMOD,EF ,I  ,WEB,CHI,THIKS,ASX,XIPFX)
       KFL=1
      KFLM=150
      KK1=1
      IF(THIKS      ,GT,,25)   GO TO 987
      THIKS=0,25
      PRINT 987,I
  987 FORMAT(/,* FOR I= *,I3 ,* THIKS =1,0 INCHES *,/)
  987 APE = EF*THIKS
      XIPE = APE*THIKS**2/3,0
      WEBI=6,
      WEBII = 50,
 12   WEB = WEBI
C
      KFAL=1
    4 IF(WEB.GT,1,0)  GO TO  44
      PRINT 900,WEB,I
  900 FORMAT(/,*  WEB LESS THAN  1,0 INCHES NOW  1,0 WEB = *,F10,4,*
     1  FOR ELEMENT I= *,I3,/)
      WEB=1,
      GO TO 55
   44 CALL FSHAPE(WEB,XIS,ASX,XKSX,KK1)
      CHI = (ASX*XKSX*WEB-0.5*THIKS      *APE)/(ASX+APE)
      XIPFX = (WEB - CHI)*FSECMOD
      CHIA=ABSF(CHI)
      XIT = (APE +ASX) * CHIA**2
      Q = XIPE +XIS -XIT - XIPFX
      GO TO (1,2,3)KFAL
    1 WEB = WEBII
      QA =Q
      KFAL=2
      GO TO 4
    2 QB = Q
    9 WEB= WEBI + QA/(QA-QB)*(WEBII - WEBI)
      KFAL = 3
      GO TO 4
    3 IF(ABSF(Q/(WEB-CHI)),LT, ,01*FSECMOD)  GO TO 5
    6 KFL=KFL+1
      IF(KFL.GE.KFLM)GO TO 5
      IF(QB*Q)7,7,8
    7 QA = Q
      WEBI = WEB
      GO TO 9
    8 WEBII = WEB
      GO TO 2
    5 IF(WEB.GT,1,)   GO TO 55
      PRINT 9988,I,WEB
 9988 FORMAT(/,* WEB SET TO  1,0 INCHES I = *,I3,* WEB WAS = *,F10,3,/)
      WEB=1,
   55 CALL FSHAPE(WEB,XIS,ASX,XKSX,KK1)
      CHI = (ASX*XKSX*WEB-0,5*THIKS      *APE)/(ASX+APE)
      CHIA=ABSF(CHI)
      XIT = (APE +ASX) * CHIA**2
      XIPFX = XIPE + XIS - XIT
 8995 FORMAT(/,*  SECMOD = *,E13,5, *I  =  *,I 2,/)
      RETURN
      END
```

Subroutine SHAPES(FSECMØD, EF, I, WEB, CHI, THIKS, ASX, XIPFX)




B
CALL FSHAPE(WEB, XIS, ASX, XKSX, KK1)
CHI = (ASX*XKSX*WEB - 0.5* THIKS*APE)/(ASX + APE)
CHIA = ABSF(CHI)
XIT = (APE + ASX)*CHIA**2
XIPFX = XIPE + XIS - XIT
RETURN


C
CALL FSHAPE(WEB,XIS, ASX, XKSX, KK1)
CHI = (ASX*XKSX*WEB - 0.5* THIKS*APE)/(ASX + APE)
XIPFX = (WEB - CHI)*FSECMØD
CHIA = ABSF(CHI)
XIT = (APE + ASX)*CHIA**2
Q = XIPE + XIS - XIT - XIPFX
KFAL = 1
2
≥ 3
QB = Q
WEBII = WEB
WEB = WEBI + QA/ (QA - QB)*(WEBII - WEBI)
KFAL = 3
D
WEB = WEBII
QA = Q
KFAL = 2
D
Q/(WEB - CHI) < 0.01*FSECMØD
Yes
No
KFL = KFL + 1
KFL ≥ KFLM
Yes
No
WEB > 1.0
Yes
No
E
QB*Q ≤ 0.0
No
Yes
Print Web Height
QA = Q
WEBI = WEB
E
WEB = 1.0

21. Subroutine SOLVE

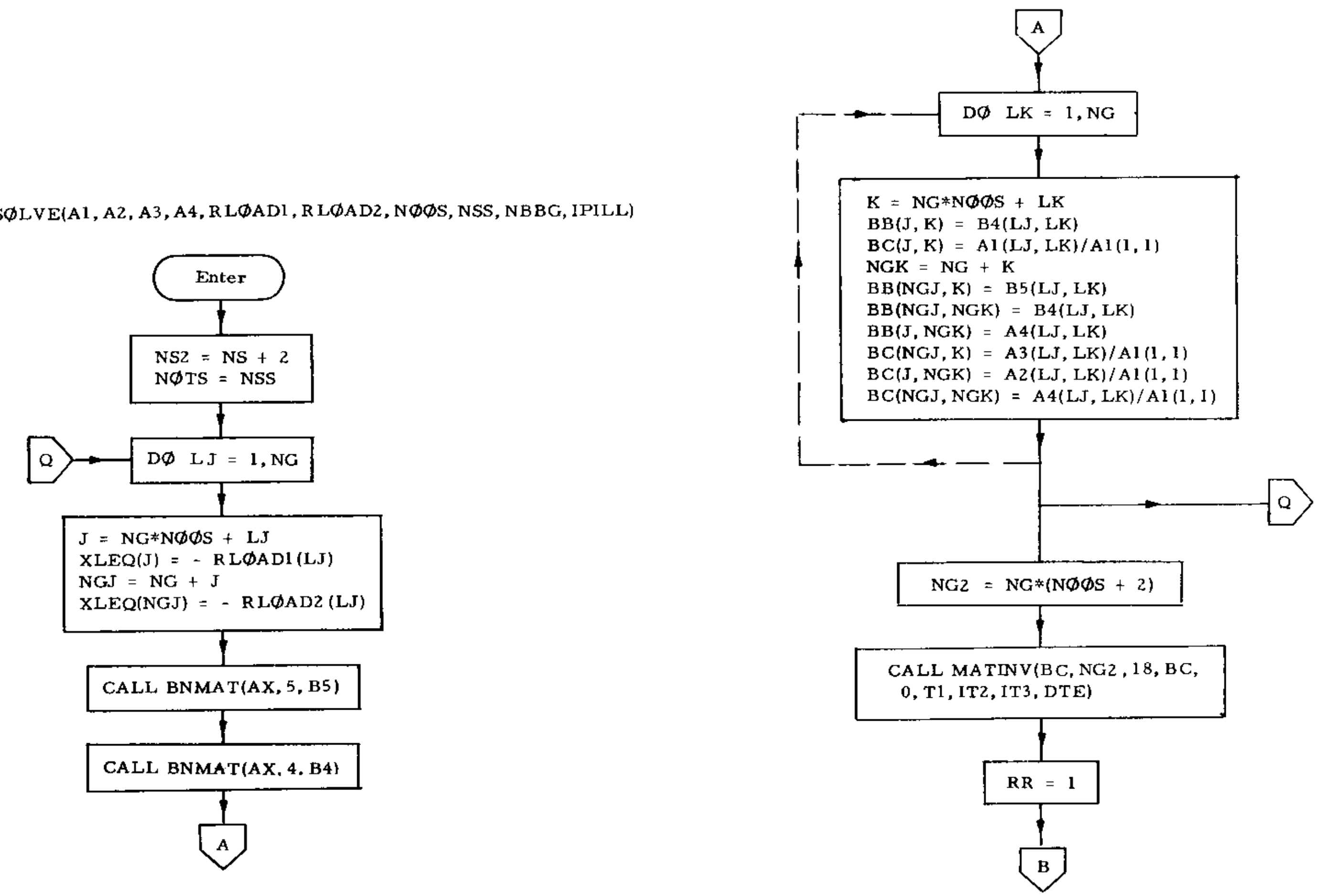
a) Abstract:

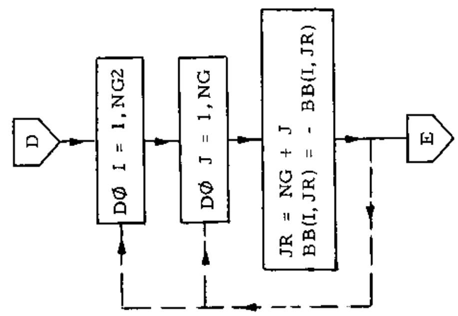
This subroutine sets up and solves the grillage slope-deflection equations for the joint deformation and forces with the aid of subroutine BNMAT. Subroutine MATINV is used in the calculation for the joint deformation and forces.

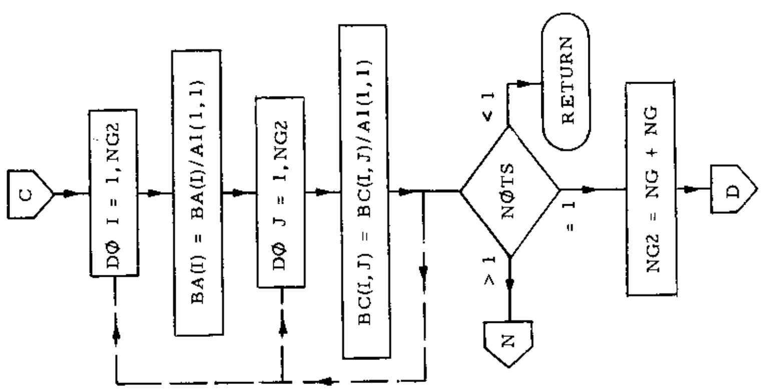
```
      SUBROUTINE SOLVE(A1,A2,A3,A4,RLOAD1,RLOAD2,NOOS,NSS,NBBG,IPILL)
      DIMENSION CH(18,18),CB(18,18)
      DIMENSION CX(18,18)
       DIMENSION A1(9,9),A2(9,9),A3(9,9),RLOAD1(9),RLOAD2(9)
     1,BA(18),XLEQ(18),A4(9,9),EE(18),RLL(6,9,9),XL1(9),NOS(6)   ,
     2            BC(18,18),II(18),CK(9,9),B1(9,9),Y(9),SMM(6,15,15)
     3,T1( 18), IT2( 18), IT3( 18,2)
     4,C(18,18),BB(18,18), BS(18),B4(9,9),B5(9,9),BD(18)
      COMMON   ALFA, AX, CC, CN, DEFL,     GP, H, JJ, K, NG, NS, PD,
     1ROOT, S, SS, V, VV, W, X, XID, XII, XL, XLL, XML, XMU, XX,RR
      COMMON /E/ IBT,E
       DIMENSION CHB(18)
      COMMON /D/  VS,VY
      DIMENSION  VS(15),VY(25)
      DIMENSION  BDD(18)
      DIMENSION  ALFA(15,15),AX( 9,9 ),CC(10),DEFL( 9),PD(15,40),
     1ROOT( 9),S(15),V(15),VV(15),W(15),XID(15),XML(15),XMU(15),XX(40),
     2XII(40)
      NS2=NS+2
      NOTS=NSS
  900 DO 1000 LJ=1,NG
      J=NG*NOOS+LJ
      XLEQ(J) = -RLOAD1(LJ)
      NGJ = NG+J
      XLEQ(NGJ) = -RLOAD2(LJ)
       CALL BNMAT(AX,5,B5)
       CALL BNMAT(AX,4,B4)
      DO 1000 LK=1,NG
      K = NG * NOOS + LK
       BB(J,K)=B4(LJ,LK)
      BC(J,K) = A1(LJ,LK)/A1(1,1)
      NGK = NG + K
       BB(NGJ,K)=B5(LJ,LK)
       BB(NGJ,NGK)=B4(LJ,LK)
       BB(J,NGK)=A4(LJ,LK)
      BC(NGJ,K) = A3(LJ,LK)/A1(1,1)
      BC(J,NGK) = A2(LJ,LK)/A1(1,1)
      BC(NGJ,NGK) = A4(LJ,LK)/A1(1,1)
 1000 CONTINUE
      NG2 = NG * (NOOS + 2)
      CALL MATINV ( BC, NG2, 18, BC, 0, T1, IT2,IT3,DTE)
         RR= 1
   22 DO 111 I=1,NG2
        BA(I)=0,0
        DO 111 J=1,NG2
       BA(I)=BA(I)+BC(I,J)*XLEQ(J)
  111  CONTINUE
      DO 2000 LJ = 1,NG
      J = NG * NOOS + LJ
      NGJ = NG+J
       V(LJ)=BA(J)/A1(1,1)
      VV(LJ)=BA(NGJ)/A1(1,1)
 2000 CONTINUE
      DO 2900 I=1,NG2
      BA(I)=BA(I)/A1(1,1)
      DO 2900 J=1,NG2
      BC(I,J)=BC(I,J)/A1(1,1)
 2900 CONTINUE
      IF(NOTS-1) 4000,5950,5951
 5951 NG2=NG+NG
       DO 6002 I=1,NG
       IM=I+NG
       BS(I)= BA(I)*XID(I)*E
       BS(IM)=BA(IM)*XID(I)*E
      DO 6002 J=1,NG2
       C(I,J)=0,0
      C(IM,J)=0,0
       DO 6002 IR=1,NG2
      C(I,J)=C(I,J)-BC(I,IR)*BB(IR,J)*XID(I)*E
       C(IM,J)=C(IM,J)-BC(IM,IR)*BB(IR,J)*XID(I)*E
 6002 CONTINUE
      DO 4999 I=1,NG2
      DO 4999 J=1,NG
      BC(I,J)=-BC(I,J)
 4999 CONTINUE
 6020 FORMAT(9E13,5)
      DO 6003 I=1,NG
      IM=NG+I
      DO 6003 J=1,NG2
      C(I,J)=C(I,J)-BC(I,J) *XID(I)*E
      C(IM,J)=C(IM,J)-BC(IM,J) *XID(I)*E
 6003 CONTINUE
      GO TO 4000
 5950 NG2=NG+NG
        DO 5999  I=1,NG2
       DO 5999 J=1,NG
      JR=NG+J
      BB(I,JR)=-BB(I,JR)
 5999 CONTINUE
       DO 6000 I=1,NG
       IM=I+NG
      BS(I)=BS(I)+BA(I)   *XID(I)*E
       BS(IM)=BS(IM)-BA(IM)*XID(I)*E
       DO 6000 J= 1,NG2
      CB(I,J)=0,0
      CB(IM,J)=0,0
       DO 6000 IR=1,NG2
       CB(IM,J)=CB(IM,J)+BC(IM,IR)*BB(IR,J)*XID(I)*E
      CB(I,J)=CB(I,J)-BC(I,IR)*BB(IR,J)*XID(I)*E
 6000 CONTINUE
      DO 6502 I=1,NG2
      DO 6502 J=1,NG2
      C(I,J)=CB(I,J)+C(I,J)
 6502 CONTINUE
      IF(IPILL,EQ,0)  GO TO 5963
 5962 N2=NG+2
      NB=NG+NBBG
      LN=NG+NBBG+1
      L2=NG+1
      DO 9661 I=1,NG2
      DO 9661 KK= 1,NG2
      CH(I,KK)=C(I,KK)
```

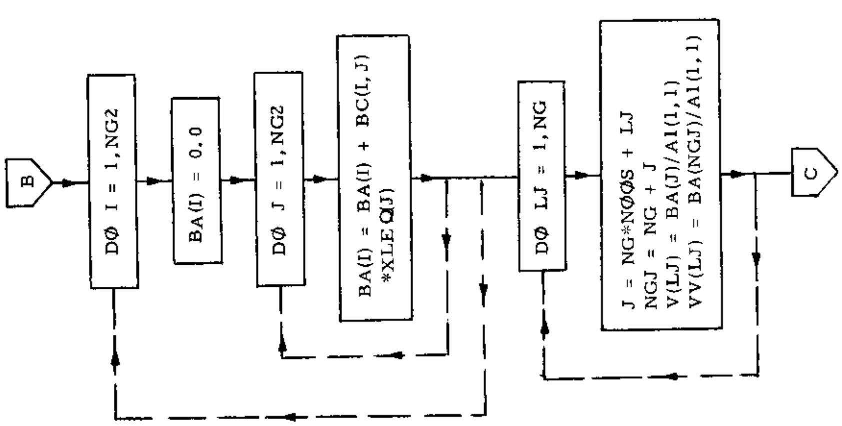
```
9661 CONTINUE
     DO 9662 I=1,NG
     DO 9662 J=1,NG
     KJ=NG+J
     KI=NG+I
     CH(I,J)=C(KI,KJ)
     CH(KI,KJ)=-C(I,J)
     CH(I,KJ)=-C(I,KJ)
9662 CONTINUE
     DO 9941,KK= L2,NG2
     KJ=KK-NG
     CHB(KJ)=BS(KK)
     CHB(KK)=-BS(KJ)
9941 CONTINUF
     DO 6393  I=1,NG2
6393 PRINT 6200,(C(I,J),J=1,NG2)
     PRINT 6000,( BS(I),I=1,NG2)
     DO 9961,  J=1,NB
     DO 9961,  JK=LN,NG2
     CH(N2,J )=CH(N2,J)+CH(JK,J)
      CH(J,N2)=CH(J,N2)+CH(J,JK)
9961 CONTINUE
     DO 9963,JK=LN,NG2
               CHB(N2)=CHB(N2)+CHB(JK)
9963 CONTINUE
     DO 9393  I=1,NB
9393 PRINT 6200,(CH(I,J),J=1,NB)
     PRINT 6000,(CHB(I),I=1,NB )
     CALL MATINV(CH, NB,18,CH,0,T1,IT2,IT3,DTE)
     DO 10 I=1,NB
  10 PRINT 6200,(CH(I,J),J=1,NB)
     DO 3399 I=1,NB
     BDD(I)=0,
     DO 3399 J=1,NB
     BDD(I)=BDD(I)-CH(I,J)*CHB(J)
3399 CONTINUE
     DO 4398 I=LN,NG2
4398 BDD(I)=BDD(N2)
      PRINT 6000,(BDD(I),I=1,NG2)
     DO 7  J=1,NG
     NGJ=NG+J
      BD(NGJ)=BDD(J)
     BD(J)=BDD(NGJ)
   7 CONTINUE
     GO TO 6096
5963 CONTINUE
     CALL MATINV(C,NG2,18,C,0,T1,IT2,IT3,DTE)
     DO 6006 I=1,NG2
     BD(I)=0,0
     DO 6006 J=1,NG2
     BD(I)=BD(I)-C(I,J)*BS(J)
6006 CONTINUE
6096 CONTINUE
     ITESTP=0
     IF(ITESTP,NE,1)  GO TO 6019
     VY(1)=  -,009047829
     VY(2)=  -,0068824396
     VY(3)=  -,0068824396
     VY(4)=  -,0068824396
     VS(1)=- ,0024218398
     VS(2)= -,0019010065
     VS(3)= -,0012228669
     VS(4)= -,00086023614
     DO 6 J=1,NG
     NGJ=NG+J
      BD(J)=VY(J)
     BD(NGJ)=VS(J)
   6 CONTINUE
6019 CONTINUE
     DO 6007 J=1,NG
     NGJ=NG+J
     VS(J)=BD(NGJ)
     VY(J)=BD(J)
6007 CONTINUE
9615 CONTINUE
     DO 6008 I=1,NG
     IM=I+NG
     V(I)= BA(I)+E*XID(I)
     VV(I )=-BA(IM)*E*XID(I)
     DO 6008 J=1,NG2
     VV(I )=VV(I )+CB(IM,J)*BD(J)
     V(I)=V(I)+CB(I,J)*BD(J)
6008 CONTINUE
4000 RETURN
 101 FORMAT (15I5)
6000 FORMAT (1H0,4H V= 9E12,4)
6010 FORMAT (1H0,4H VV= 9E12,4)
6011 FORMAT (1H0,I2,6H Y=   9E11,3)
6200  FORMAT (1H0,12E10,3/12E10,3)
     END
```

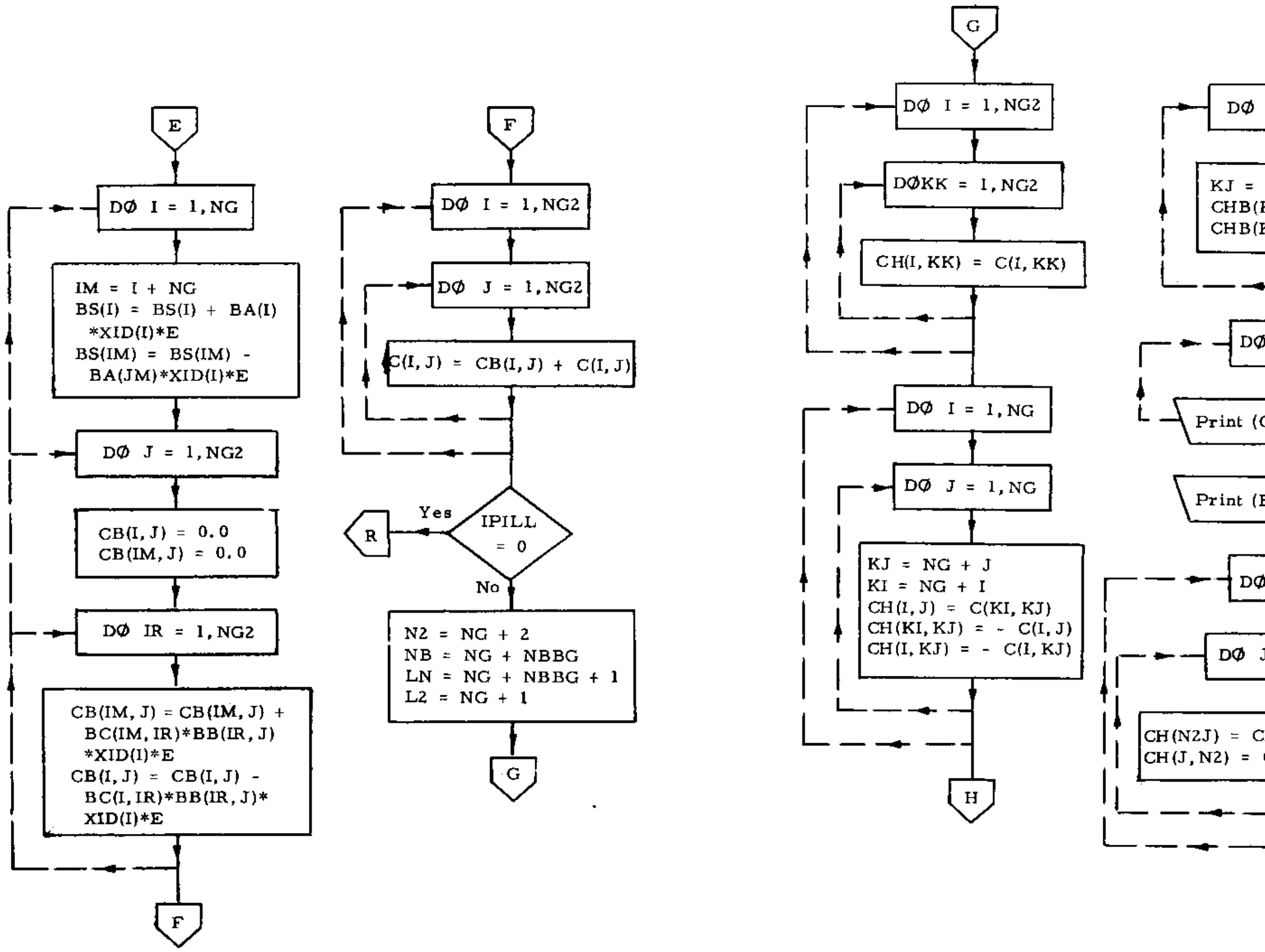
Subroutine SØLVE(A1, A2, A3, A4, RLØAD1, RLØAD2, NØØS, NSS, NBBG, IPILL)



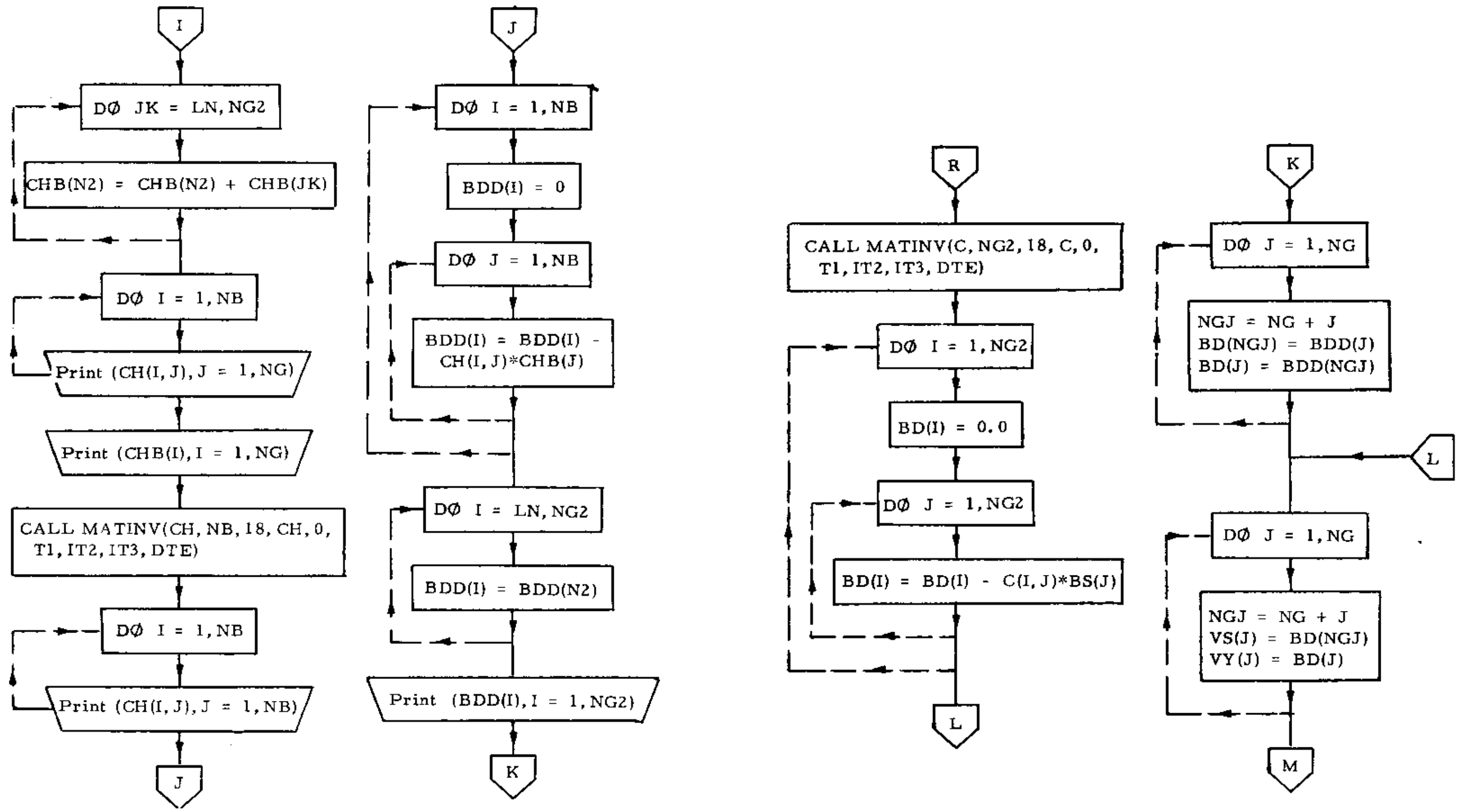

D
DØ I = 1, NG2
DØ J = 1, NG
JR = NG + J
BB(I, JR) = - BB(I, JR)
E

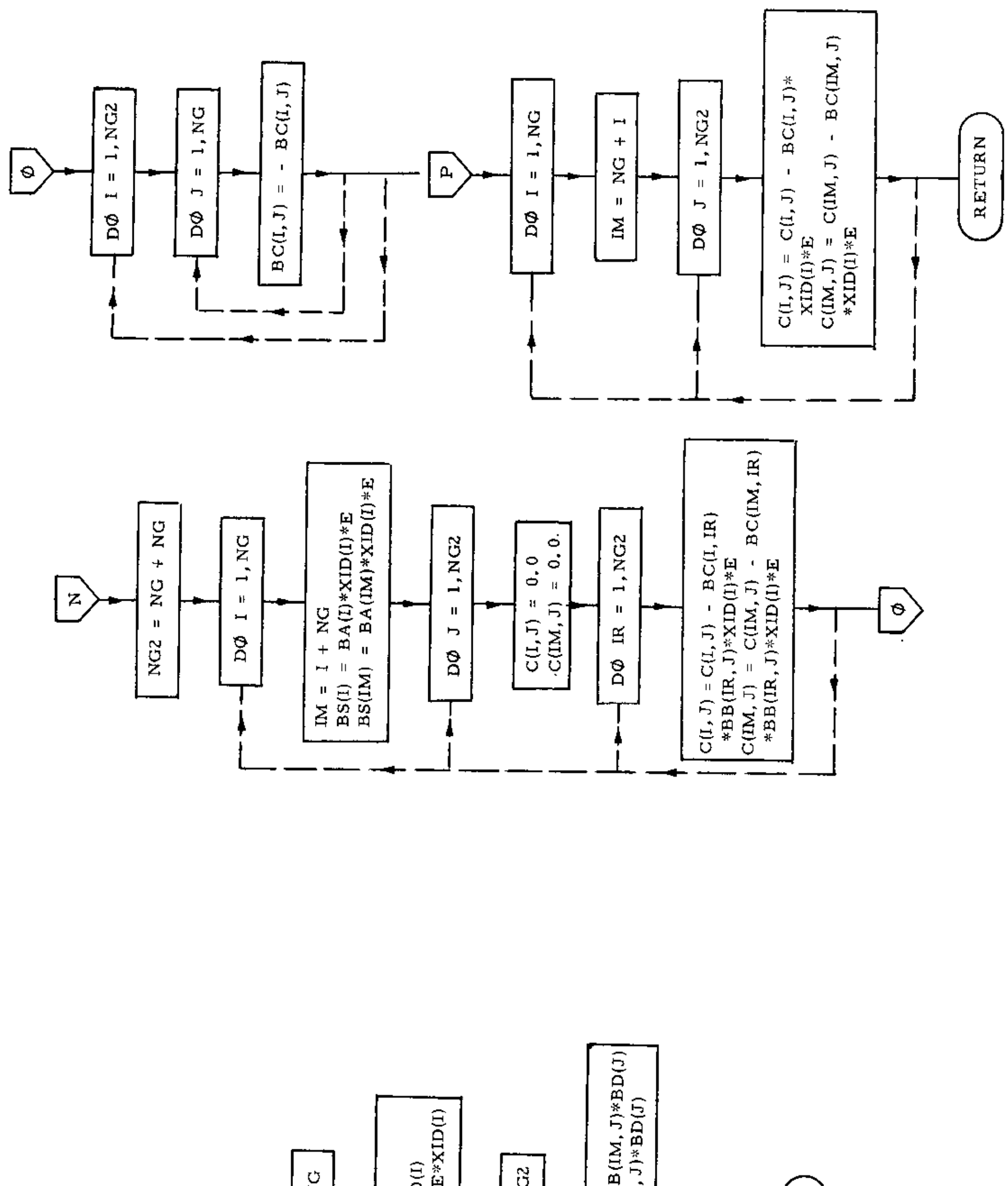

C
DØ I = 1, NG2
BA(I) = BA(I)/AI(1, 1)
DØ J = 1, NG2
BC(I, J) = BC(I, J)/AI(1, 1)
NØTS
< 1
RETURN
> 1
N
= 1
NG2 = NG + NG
D


B
DØ I = 1, NG2
BA(I) = 0.0
DØ J = 1, NG2
BA(I) = BA(I) + BC(I, J)
*XLE Q(J)
DØ LJ = 1, NG
J = NG*NØØS + LJ
NGJ = NG + J
V(LJ) = BA(J)/AI(1, 1)
VV(LJ) = BA(NGJ)/AI(1, 1)
C



E
DØ I = 1, NG
IM = I + NG
BS(I) = BS(I) + BA(I) *XID(I)*E
BS(IM) = BS(IM) - BA(JM)*XID(I)*E
DØ J = 1, NG2
CB(I, J) = 0.0
CB(IM, J) = 0.0
DØ IR = 1, NG2
CB(IM, J) = CB(IM, J) + BC(IM, IR)*BB(IR, J) *XID(I)*E
CB(I, J) = CB(I, J) - BC(I, IR)*BB(IR, J)* XID(I)*E
F
F
DØ I = 1, NG2
DØ J = 1, NG2
C(I, J) = CB(I, J) + C(I, J)
IPILL = 0
Yes
R
No
N2 = NG + 2
NB = NG + NBBG
LN = NG + NBBG + 1
L2 = NG + 1
G
G
DØ I = 1, NG2
DØKK = 1, NG2
CH(I, KK) = C(I, KK)
DØ I = 1, NG
DØ J = 1, NG
KJ = NG + J
KI = NG + I
CH(I, J) = C(KI, KJ)
CH(KI, KJ) = - C(I, J)
CH(I, KJ) = - C(I, KJ)
H


I
DØ JK = LN, NG2
CHB(N2) = CHB(N2) + CHB(JK)
DØ I = 1, NB
Print (CH(I, J), J = 1, NG)
Print (CHB(I), I = 1, NG)
CALL MATINV(CH, NB, 18, CH, 0, T1, IT2, IT3, DTE)
DØ I = 1, NB
Print (CH(I, J), J = 1, NB)
J
J
DØ I = 1, NB
BDD(I) = 0
DØ J = 1, NB
BDD(I) = BDD(I) - CH(I, J)*CHB(J)
DØ I = LN, NG2
BDD(I) = BDD(N2)
Print (BDD(I), I = 1, NG2)
K
R
CALL MATINV(C, NG2, 18, C, 0, T1, IT2, IT3, DTE)
DØ I = 1, NG2
BD(I) = 0.0
DØ J = 1, NG2
BD(I) = BD(I) - C(I, J)*BS(J)
L
K
DØ J = 1, NG
NGJ = NG + J
BD(NGJ) = BDD(J)
BD(J) = BDD(NGJ)
L
DØ J = 1, NG
NGJ = NG + J
VS(J) = BD(NGJ)
VY(J) = BD(J)
M


Ø
DØ I = 1, NG2
DØ J = 1, NG
BC(I, J) = - BC(I, J)
P
DØ I = 1, NG
IM = NG + I
DØ J = 1, NG2
C(I, J) = C(I, J) - BC(I, J)*
XID(I)*E
C(IM, J) = C(IM, J) - BC(IM, J)
*XID(I)*E
RETURN
N
NG2 = NG + NG
DØ I = 1, NG
IM = I + NG
BS(I) = BA(I)*XID(I)*E
BS(IM) = BA(IM)*XID(I)*E
DØ J = 1, NG2
C(I, J) = 0.0
C(IM, J) = 0.0
DØ IR = 1, NG2
C(I, J) = C(I, J) - BC(I, IR)
*BB(IR, J)*XID(I)*E
C(IM, J) = C(IM, J) - BC(IM, IR)
*BB(IR, J)*XID(I)*E
Ø

22. Subroutine STEP

a) Abstract:

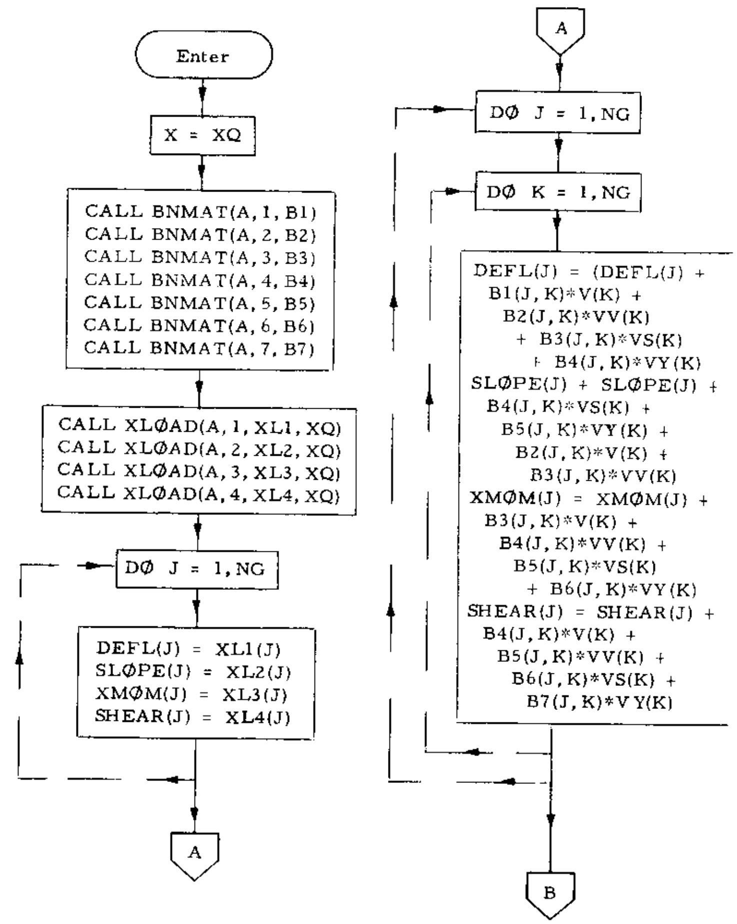
This subroutine is used to calculate the deformations and forces at the frame intersections of the longitudinal girders of the grillage. It solves the grillage deflection equation.

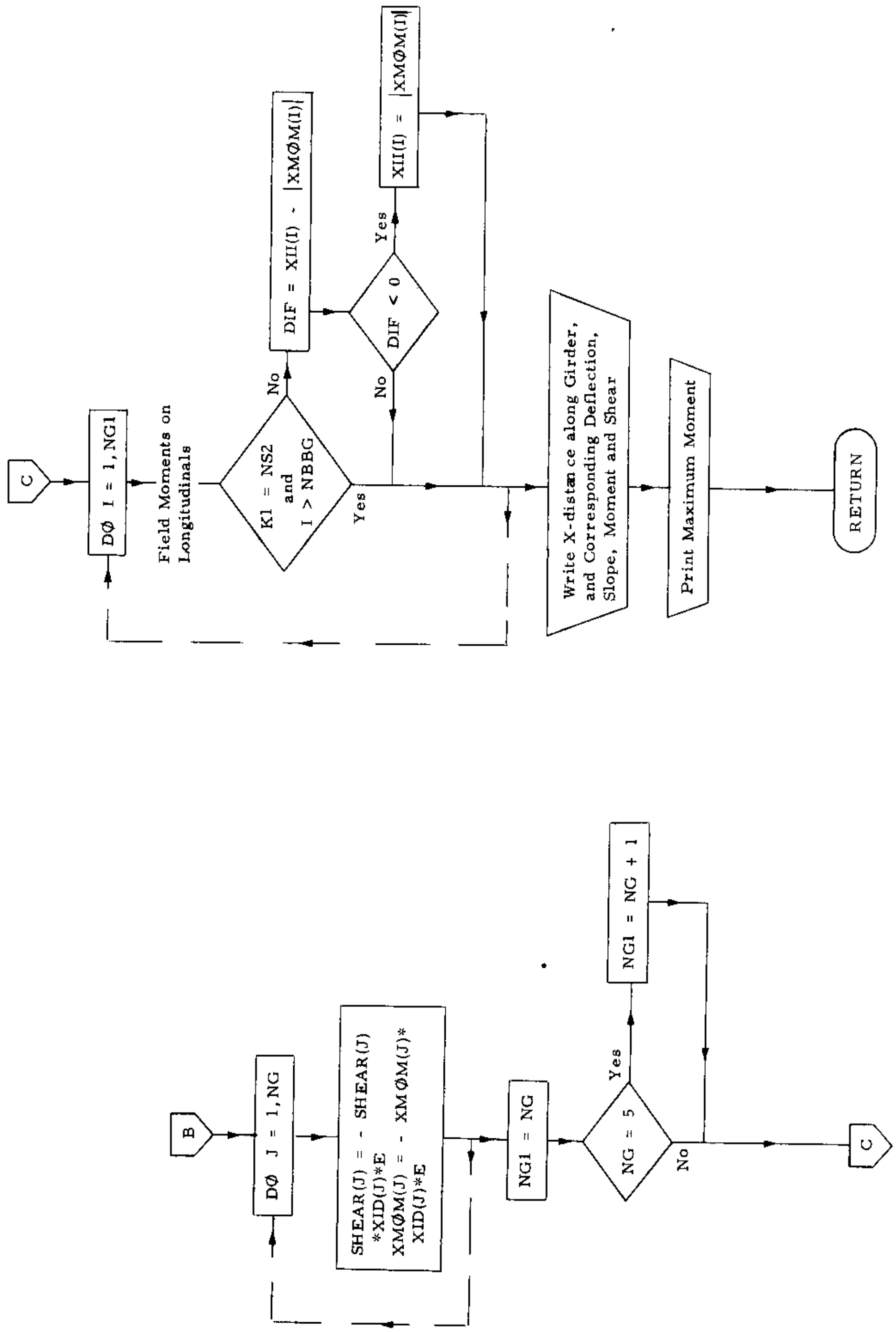
b) Terms specific to this subroutine:

FORTRAN Term	Definition
SHEAR	Shear
SLOPE	Slope
XMOM	Moment

```
      SUBROUTINE STEP(A,K1,NS2,XQ,NBBG)
      COMMON /D/  VS(15),VY(25)
      COMMON  ALFA(15,15),AX( 9,9 ),CC(10),CN,DEFL( 9),GP,H,JJ,K,NG,NS,
     1PD(15,40), ROOT(9),S(15),SS,V(15),VV(15),W(15),X,XID(15),XII(40),
     2XL,XLL,XML(15),XMO(15),XX(40),RR,SCAL,AFAC,NNN
      COMMON /E/ IBT,E
         DIMENSION A( 9,9 ),XL1( 9),XL2( 9),XL3( 9),XL4( 9),SA(40),
     1 SLOPE( 9),XMOM( 9),SHEAR( 9) ,B1(9,9)
     2,B2(9,9),B3(9,9),B4(9,9),B5(9,9),B6(9,9),B7(9,9)
      X=XQ
      CALL BNMAT (A,1,B1       )
      CALL BNMAT (A,2,B2       )
      CALL BNMAT (A,3,B3       )
      CALL BNMAT (A,4,B4       )
      CALL BNMAT (A,5,B5       )
      CALL BNMAT(A,6,B6)
      CALL BNMAT(A,7,B7)
      CALL XLOAD (A,1,XL1,XQ    )
      CALL XLOAD (A,2,XL2,XQ)
      CALL XLOAD (A,3,XL3,XQ)
      CALL XLOAD (A,4,XL4,XQ)
      DO 2000 J=1,NG
      DEFL(J) = XL1(J)
      SLOPE(J) = XL2(J)
      XMOM(J) = XL3(J)
      SHEAR(J) = XL4(J)
 2000 CONTINUE
      DO 1800 J=1,NG
      DO 1800 K=1,NG
      DEFL(J) =(DEFL(J) + B1(J,K) * V(K) + B2(J,K) * VV(K))
      DEFL(J)=DEFL(J)+B3(J,K)*VS(K)+B4(J,K)*VY(K)
      SLOPE(J)=SLOPE(J)+B4(J,K)*VS(K)+B5(J,K)*VY(K)
      SLOPE(J) =(SLOPE(J) + B2(J,K) * V(K) + B3(J,K) * VV(K))
      XMOM(J)=XMOM(J) + B3(J,K) * V(K) + B4(J,K) * VV(K)
     1  +B5(J,K)*VS(K)+B6(J,K)*VY(K)
      SHEAR(J) = SHEAR(J) + B4(J,K) * V(K) + B5(J,K) * VV(K)
     1  +B6(J,K)*VS(K)+B7(J,K)*VY(K)
 1800 CONTINUE
      DO 2001 J=1,NG
      SHEAR(J)=-SHEAR(J) * XID(J) *    E
      XMOM(J)=-XMOM(J)* XID(J) * E
 2001 CONTINUE
      NG1=NG
      IF(NG,EQ,5)  NG1=NG+1
      DO 8300 I=1,NG1
C   FIELD MOMENTS ON LONG,
      IF(K1,EQ,NS2,AND,I,GT,NBBG)  GO TO 8300
       DIF=XII(I)-ABSF(XMOM(I))
       IF (DIF) 8301,8300,8300
 8301 XII(I)=ABSF(XMOM(I))
 8300 CONTINUE
  139 FORMAT (1H0,10X,13H EIGENVALUES //  5X, 7H REAL  9E11,4)
 3000  WRITE OUTPUT TAPE 6,  4000,X,(K,K=1,NG)
       WRITE OUTPUT TAPE 6,  4010,(DEFL(K),K=1,NG)
       WRITE OUTPUT TAPE 6,4020,(SLOPE(K),K=1,NG)
       WRITE OUTPUT TAPE 6, 4030,(XMOM(K),K=1,NG)
       WRITE OUTPUT TAPE 6,  4040,(SHEAR(K),K=1,NG)
 8305 FORMAT(/,*  MAXIMUM MOMENT *,/)
      PRINT 8305
      PRINT 8303,(XII(I)   ,I=1,NG)
      RETURN
 4000 FORMAT (1H0,    28H DISTANCE FROM GIRDER ORIGIN F7,2/9I13)
 4010 FORMAT (1H0,     12H DEFLECTION ,    12H       INCHES /9E13,5)
 4020 FORMAT (1H0,    7H SLOPE ,11H        RAD / 9E13,5)
 4030 FORMAT (1H0,    8H MOMENT , * LB-IN         *,/9E13,5)
 4040 FORMAT (1H0,    7H SHEAR ,11H    LBS    /9E13,5)
 8303 FORMAT(7E13,5)
         END
```

Subroutine STEP (A, K1, NS2, XQ, NBBG)




C
DØ I = 1, NG1
Field Moments on Longitudinals
K1 = NS2 and I > NBBG
Yes
No
DIF = XII(I) - |XMØM(I)|
DIF < 0
Yes
No
XII(I) = |XMØM(I)|
Write X-distance along Girder, and Corresponding Deflection, Slope, Moment and Shear
Print Maximum Moment
RETURN
B
DØ J = 1, NG
SHEAR(J) = - SHEAR(J) *XID(J)*E
XMØM(J) = - XMØM(J)* XID(J)*E
NG1 = NG
NG = 5
Yes
No
NG1 = NG + 1
C

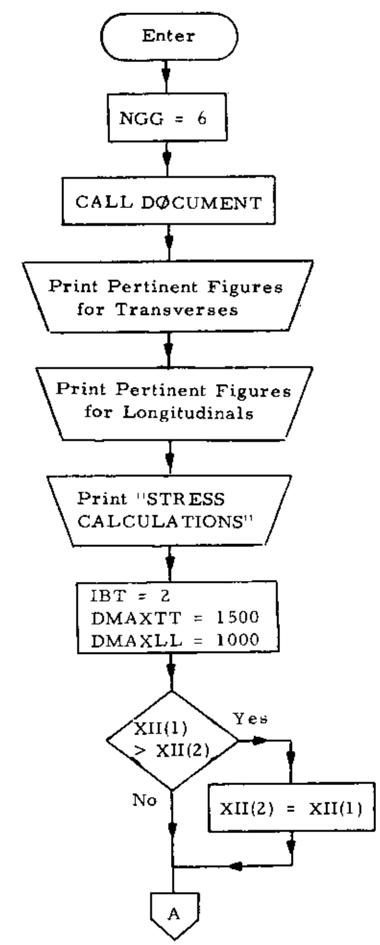
23. Subroutine STRESS

a) Abstract:

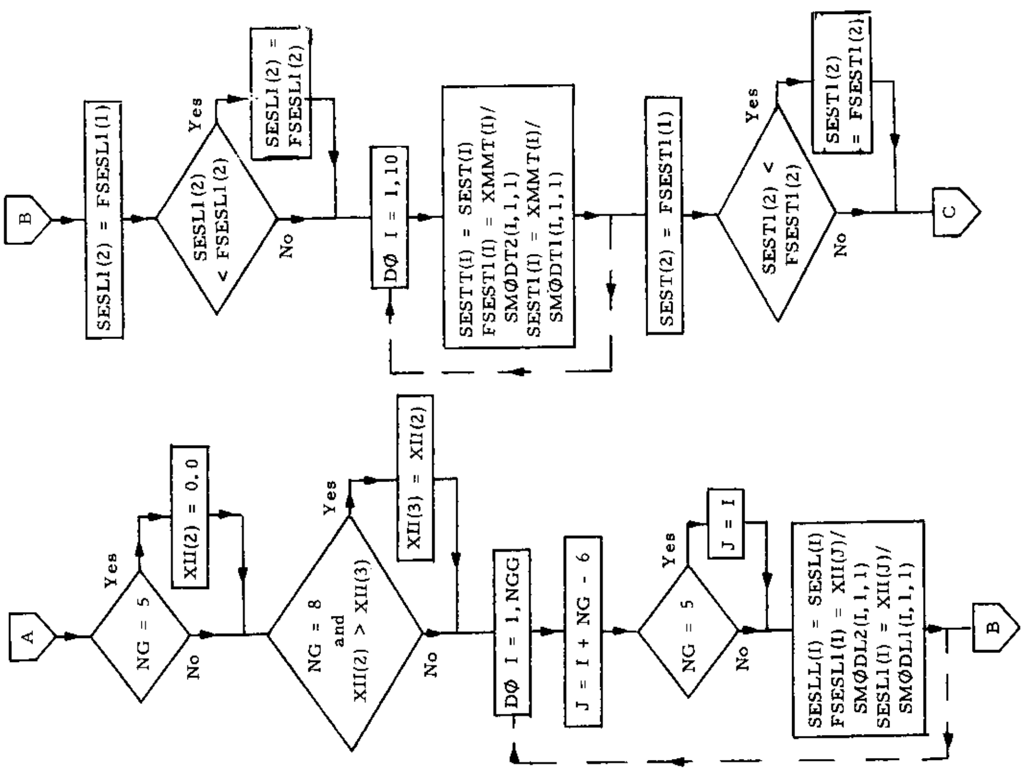
Subroutine STRESS is called from program TRANSHIP and it administers the convergence of the solution. It calculates stress intensities based on scantlings calculated in subroutine SECTION and grillage moments calculated in subroutine GRILLAGE. New criterion stress intensities are then calculated unless the last calculation meets the criteria for stress intensities.

```
      SUBROUTINE  STRESS(            N,NN)
      COMMON  ALFA(15,15),AX( 9,9 ),CC(10),CN,DEFL( 9),GP,H,JJ,K,NG,NS,
     1PD(15,40), ROOT(9),S(15),SS,V(15),VV(15),W(15),X,XID(15),XII(40),
     2XL,XLL,XML(15),XMO(15),XX(40),RR,SCAL,AFAC,NNN
      COMMON/DE/ DEF(20)
      COMMON/LO/FSECLO(12),FSESL(17),FSESL1(17),SMODL2(17,2,3),PSLL(17)
      COMMON/TR/ FSECTR(17),FSEST(17),FSEST1(17),SMODT2(17,2,3),SMODT3(
     1 17,2,5)
      COMMON/H/ XMMT(17)
      DIMENSION  SEST1(17),SESL1(17)
      COMMON/LABEL/ TRANLAB(10),PLATLAB(15),LONGLAB(6)
      COMMON /B/ A,B,NDECKS,IPLATE,PSL,PRM,HMAIN,HNEUT,TESL,TEST,SESL,
     1     SEST,          KPANELS,BEAM,XNG,HATCH,THIKK,THIKK1,EFFBR,EFFB1,
     2     EFFW,EFFW1,HFLOOR,TWEENH,THIKX,EFBR,WEB,WEB1,XIPFX,CHI,
     3     XIS, AREAT1,SMAL1,SMODL1,WEBL1,SUMAREAP,SUMAREAL,SUMMP,SUMML,
     4     AREA,SUMAREA,SUMMOM,WPL,XPANEL,WBAYLP,XNST,NST,ITR,
     5     WTR,BAYTR,SUMWT,SUMWTR,LLGX,   YIELD,WBAYTR,XLABOR,PLCOST,
     6     COST,COSTMIN,GNOT,GNNT,KNEUT,SUMSMAP,SUMSMAL,SUMSMA,CODE,
     7WAVEH,PRHEAD,DRAFT,ALOAD,XLHOLD,XLPANEL,AAA,BBB,AA,BB,DELTESL,
     8      SUMSMA1,ZP,ZL,ILS,AREAL1,ASX,   SMAT1,SMODT1
      DIMENSION ZP(17),SESL(17),SEST(17),TESL(17),TEST(17),XNG(7),THIKK(
     117),       THIKK1(17,2,5),EFFBR(17),EFFW(17),EFFB1(17,2,5),EFFW1(17
     2,2,5),AREA(17,5),TWEENH(5),SMAT1(12,2,5),SMODT1(12,2,5),THIKX(12,2
     3),EFBR(12,2),AREAT1(12,2,5),WEB1(12,2,5),SMAL1(17,2,3),SMODL1(17,2
     4,3),WEBL1(7,2,3),AREAL1(7,2,3),SUMAREAP(5),SUMAREAL(5),SUMMP(5),SU
     5MML(5),SUMAREA(5),SUMMOM(5),WPL(5),XLPANEL(5),SUMWTR(5),WTR(12),ZL
     6(7),TIKK1(17,5),CODE(17),ALOAD(17),SUMSMAP(5),SUMSMAL(5),SUMSMA(5)
      DIMENSION RATIO1(12),RATIO 2(12),RATIO3(12)
      DIMENSION DIFFSSL(12),DIFFSST(12),DMAXL(12),DMAXT(12),SESLL(12)
     1 ,SESTT(12)
      NGG=6
      CALL DOCUMENT
      PRINT 1212
 1212 FORMAT( * TRANSVERSES*//
     1       *             SECOND      SECTION    SECTION    SHAPE      EFFECTIVE
     2  LEVER FEET  LOAD PSI */
     3       *             MOMENT      MODULUS    MODULUS    AREA       BREADTH*/
     4       *             OF AREA     PLATE      FLANGE   *//)
      DO 702 I=1,10
  702 PRINT 795 ,TRANLAB(I),   SMAT1(I,1,1),SMODT1(I,1,1),SMODT2(I,1,1),
     1 AREAT1(I,1,1),EFFB1(I,1,1),ZP(I) ,ALOAD(I)   ,DEF(I)
      PRINT 1111
 1111 FORMAT( * LONGITUDINALS*//
     1*                        SECOND      SECTION    SECTION    SHAPE      EFFECTIVE
     2  LEVER FEET */
     3*                        MOMENT      MODULUS    MODULUS    AREA       WIDTH  */
     4*                        OF AREA     PLATE      FLANGE   *//)
      DO 704  I=1,6
  704 PRINT 795 ,LONGLAB(I),SMAL1(I,1,1),SMODL1(I,1,1),SMODL2(I,1,1),
     1 AREAL1(I,1,1),EFFW1(I,1,1),ZL(I)
      PRINT 6521
 6521 FORMAT( 1H1, * STRESS CALCULATIONS *,///)
      IBT=2
      DMAXTT=1500,
      DMAXLL=1000,
      IF(XII(1),GT,XII(2) ) XII(2)=XII(1)
      IF(NG,EQ,5)XII(2)=0,0
      IF(NG,EQ,8,AND,XII(2),GT,XII(3) )  XII(3)=XII(2)
      DO 3625 I=1,NGG
       J=I+NG-6
      IF(NG,EQ,5) J=I
       SESLL(I)=SESL(I)
      FSESL1(I)=XII(J)/SMODL2(I,1,1)
      SESL1(I) = XII(J)           /SMODL1(I,1,1)
 3625 CONTINUE
        SESL1(2)=FSESL1(1)
      IF(SESL1(2),LT,FSESL1(2)   ) SESL1(2)=FSESL1(2)
      DO 3626 I=1,10
        SESTT(I)=SEST(I)
      FSEST1(I) = XMMT(I)/SMODT2(I,1,1)
        SEST1(I) = XMMT(I)/SMODT1(I,1,1)
 3626 CONTINUE
         SEST1(2)=FSEST1(1)
      IF(SEST1(2),LT,FSEST1(2)   ) SEST1(2)=FSEST1(2)
 8012 FORMAT(/,7E13,5,/)
      N=0
       DO  315 I=1,NGG
      RATIO1(I)=(PSLL(I)+SESL1(I)+TESL(I))/YIELD
      RATIO2(I)=(PSLL(I)+SESL1(I))/YIELD
      RATIO3(I)=PSLL(I)/YIELD
       J=I+NG-6
      IF(NG,EQ,5)  J=I
       DIFFSSL(I)=ABSF(SESL(I)-SESL1(I))
      DMAXL(I)=ABSF(FSESL(I)-FSESL1(I))
      IF(I,LE,IBT)   DMAXL(I)=ABSF(SESL(2)-FSESL1(I) )
                         FSESL(I)=   ,8*YIELD -PSLL(I)
      SESL(I)=SESLL(I)+( SESL1(I)-SESLL(I) )/2,0
        FSECLO(I)=XII(J)            /FSESL(I)
      IF(DMAXL(I)-DMAXLL   ) 7099,7099,325
  325 N=1+N
 7099 CONTINUE
  315 CONTINUE
      DO 316 I=1,10
        DIFFSST(I)=ABSF(SEST(I)-SEST1(I))
      DMAXT(I)=ABSF(FSEST(I)-FSEST1(I))
      IF(I,LE,IBT)   DMAXT(I)=ABSF(SEST(2)-FSEST1(I) )
  316 CONTINUE
      DO 7998 I=1,10
       SEST(I) = SEST1(I)
                                                FSEST(I)=0,80*YIELD
      FSECTR(I)=XMMT(I)/FSEST(I)
 7998 CONTINUE
      NN=0
      DO 7997  I=1,10
      XII(I)=XII(I)/1000,
       XMMT(I)=XMMT(I)/1000,
      IF(DMAXT(I)-DMAXTT) 7997,7997,338
  338 NN=NN+1
 7997 CONTINUE
  323 JCODE=1
 8400 CONTINUE
      PRINT 771
```

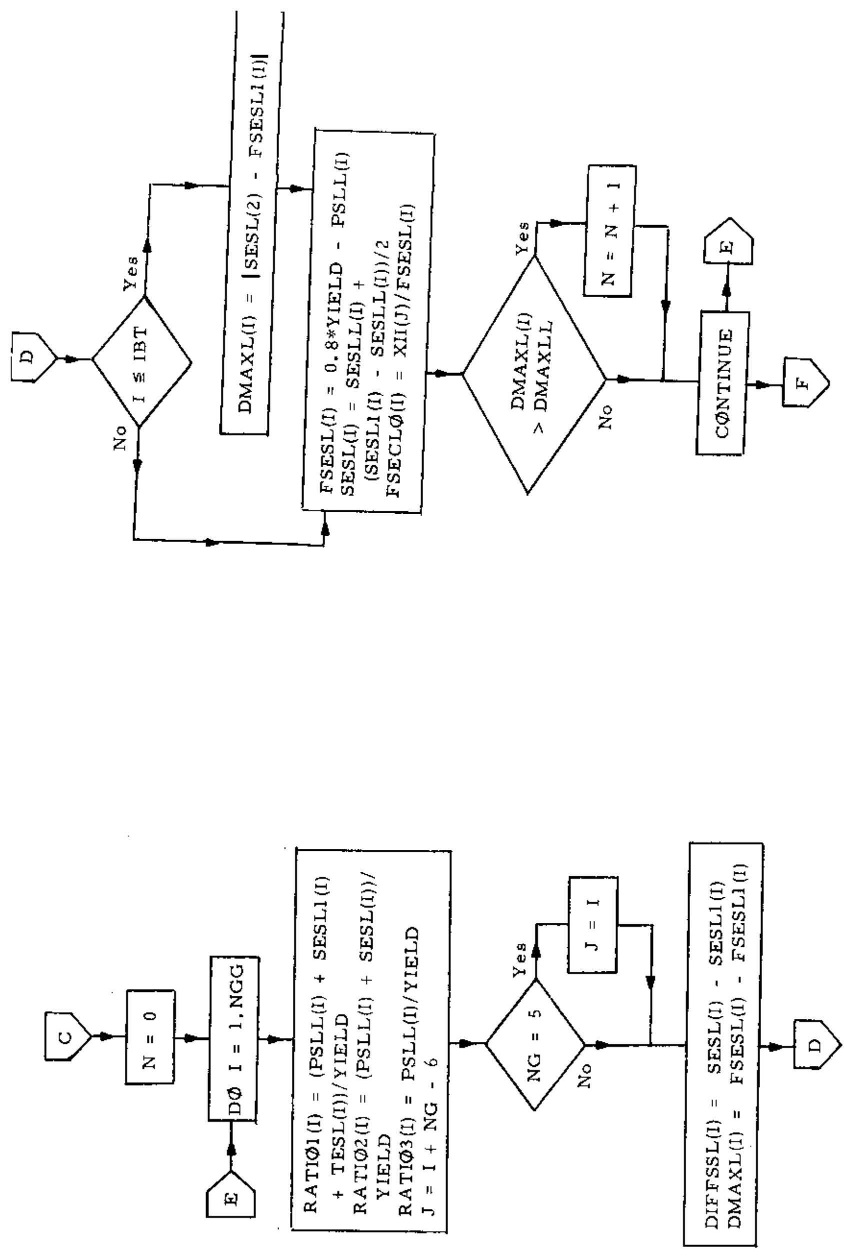
Subroutine STRESS (N, NN)

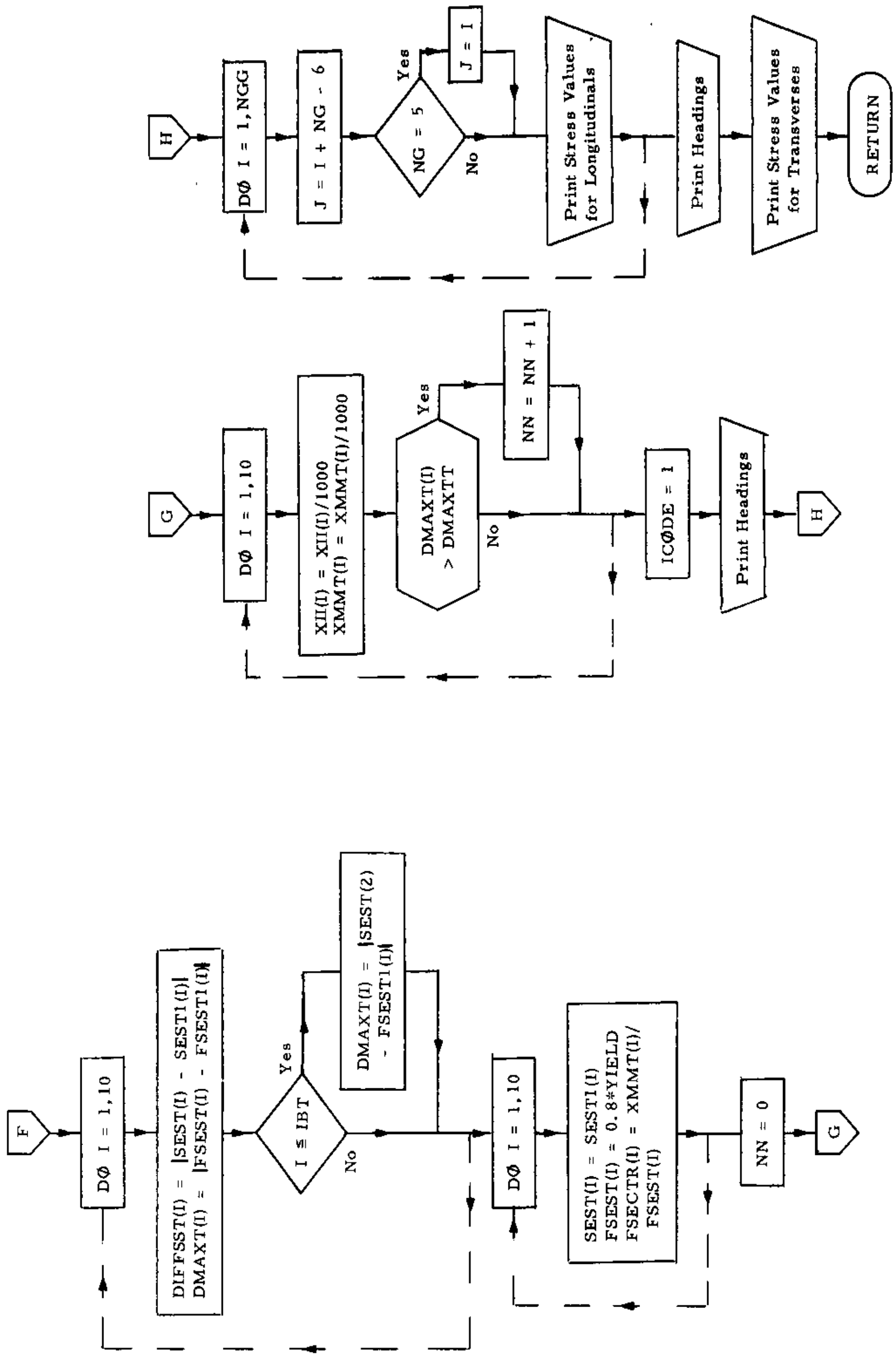



```
 771 FORMAT(1H0,*          MOM/1000    F, STRESS      P, STRESS     TESL
    1     SESL   *,/)
     DO 9702 I=1,NGG
     J=I+NG-6
     IF(NG.EQ,5)  J=I
9702 PRINT 705 ,LONGLAB(I), XII(J),FSESL1(I),SESL1(I ),TESL(I),SESLL(I)
     PRINT 711
 711 FORMAT (/,*             DMAX       SECMOD          RATIO1       RATIO2
    1       RATIO3   *,/)
     DO 706 I=1,6
 706 PRINT 705,LONGLAB(I),DMAXL(I),FSECLO(I),RATIO1(I),RATIO2(I),RATIO3
    1 (I)
     PRINT 791
 791 FORMAT(1H0,*          MOM/1000    F, STRESS      P, STRESS     TEST
    1     SEST    *,/)
        DO 709 I=1,10
 709 PRINT 705 ,TRANLAB(I),XMMT(I),FSEST1(I),SEST1(I),TEST(I),SESTT(I)
     PRINT 799
 799 FORMAT (/,*             DMAX       SECMOD          DIFFT        DIFFL
    1        PSLL    *,/)
     DO 708 I=1,10
 708 PRINT 705,TRANLAB(I),DMAXT(I),FSECTR(I),       DIFFSST(I),DIFFSSL(I)
    1,PSLL(I)
 795  FORMAT(*   *, A8,8F10,2)
 705  FORMAT(*   *, A8,8F12,2)
      RETURN
      END
```



A
NG = 5
Yes
No
XII(2) = 0.0
NG = 8
and
XII(2) > XII(3)
Yes
No
XII(3) = XII(2)
DØ I = 1,NGG
J = I + NG - 6
NG = 5
Yes
No
J = I
SESLL(I) = SESL(I)
FSESL1(I) = XII(J)/
SMØDL2(I,1,1)
SESL1(I) = XII(J)/
SMØDL1(I,1,1)
B
B
SESL1(2) = FSESL1(1)
SESL1(2)
< FSESL1(2)
Yes
No
SESL1(2) =
FSESL1(2)
DØ I = 1,10
SESTT(I) = SEST(I)
FSEST1(I) = XMMT(I)/
SMØDT2(I,1,1)
SEST1(I) = XMMT(I)/
SMØDT1(I,1,1)
SEST(2) = FSEST1(1)
SEST1(2) <
FSEST1(2)
Yes
No
SEST1(2)
= FSEST1(2)
C


C
N = 0
DØ I = 1, NGG
E
RATIØ1(I) = (PSLL(I) + SESL1(I) + TESL(I))/YIELD
RATIØ2(I) = (PSLL(I) + SESL(I))/YIELD
RATIØ3(I) = PSLL(I)/YIELD
J = I + NG - 6
NG = 5
Yes
No
J = I
DIFFSSL(I) = SESL(I) - SESL1(I)
DMAXL(I) = FSESL(I) - FSESL1(I)
D
D
I ≤ IBT
Yes
No
DMAXL(I) = |SESL(2) - FSESL1(I)|
FSESL(I) = 0.8*YIELD - PSLL(I)
SESL(I) = SESLL(I) + (SESL1(I) - SESLL(I))/2
FSECLØ(I) = XII(J)/FSESL(I)
DMAXL(I) > DMAXLL
Yes
No
N = N + 1
CØNTINUE
E
F



H
DØ I = 1,NGG
J = I + NG - 6
NG = 5
Yes
No
J = I
Print Stress Values for Longitudinals
Print Headings
Print Stress Values for Transverses
RETURN
G
DØ I = 1,10
XII(I) = XII(I)/1000
XMMT(I) = XMMT(I)/1000
DMAXT(I) > DMAXTT
Yes
No
NN = NN + 1
ICØDE = 1
Print Headings
H
F
DØ I = 1,10
DIFFSST(I) = |SEST(I) - SEST1(I)|
DMAXT(I) = |FSEST(I) - FSEST1(I)|
I ≤ IBT
Yes
No
DMAXT(I) = |SEST(2) - FSEST1(I)|
DØ I = 1,10
SEST(I) = SEST1(I)
FSEST(I) = 0.8*YIELD
FSECTR(I) = XMMT(I)/FSEST(I)
NN = 0
G

24. Function T

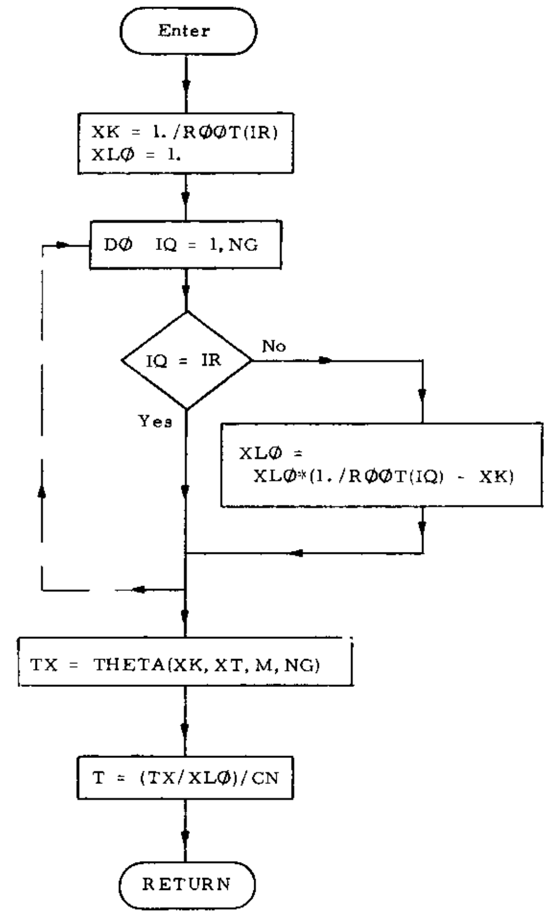
a) Abstract:

This function calculates the values of the Laplace transform used in the grillage calculation. This function is called for by both subroutines BNMAT and XLOAD and in turn calls for function THETA.

b) Term specific to this subroutine:

FORTRAN Term	Definition
AFAC	Scale factor - Set equal to unity in this study
CN	Determinant of A matrix
IR	Number of Nielsen functions
M	Number of Nielsen functions
NG	Number of longitudinal girders
ROOT	Root array of A matrix
XT	Distance of the longitudinal girder from the origin

Function T (XT, IR, M, NC, ROOT, CN, AFAC)




```
      FUNCTION T(XT,IR,M,NG,ROOT,CN,AFAC)
      DIMENSION ROOT(8)
   98 FORMAT (I3)
   91 FORMAT (F15.5)
  107 FORMAT(9E12.4)
      XK = 1./ROOT(IR)
      XLO = 1.
      DO 2 IQ = 1,NG
      IF (IQ-IR)1,2,1
    1 XLO = XLO * (1. / ROOT(IQ)-XK)
    2 CONTINUE
      TX=THETA(XK,XT,M,NG)
      T = (TX/XLO)/CN
      RETURN
      END
```

25. Function THETA

a) Abstract:

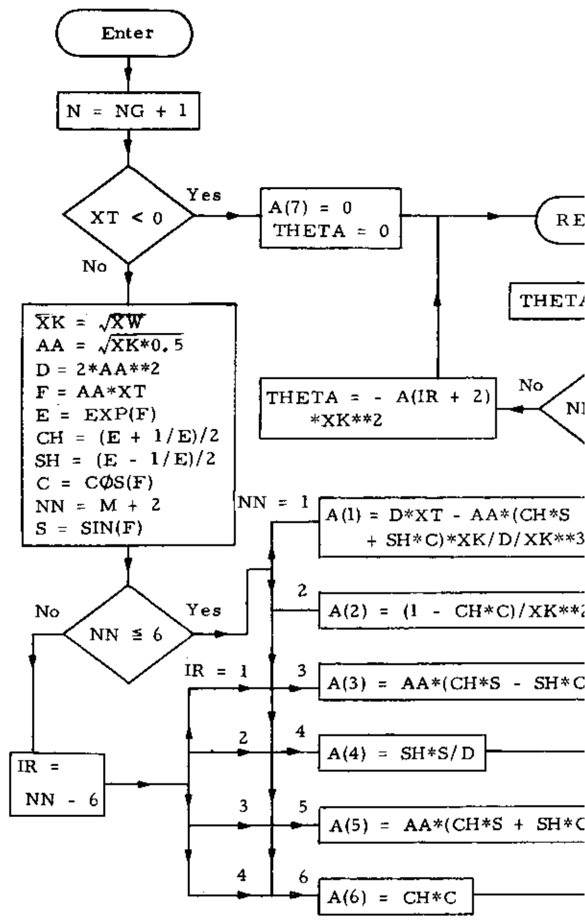
This function is called by function T to calculate the Nielsen functions N_M used in the grillage calculation.

b) Terms specific to this subroutine:

FORTRAN Term	Definition
M	Index of Nielsen function
NG	Number of longitudinal girders in the grillage
XT	Transverse distance of the longitudinal girder from the origin
XW	Root of the grillage matrix

```
      FUNCTION THETA(XW,XT,M,NG)
        DIMENSION  A(15)
   98 FORMAT (I3)
   91 FORMAT (F15,5)
      N = NG+1
      IF (XT)11,1,1
    1 XK = SQRT(XW)
      AA = XK*,5
      AA = SQRT(AA)
       D = 2,*AA**2
       F= AA*XT
       E = EXP(F)
      CH = (E+1,/E)/2,
      SH = (E-1,/E)/2,
       C = COS(F)
      NN = M+2
       S = SIN(F)
      IF (NN-6)2,2,3
    3 IR = NN-6
      GO TO (6,7,8,9),IR
    2 GO TO (4,5,6,7,8,9),NN
    4 A(1)=(D*XT-AA*(CH*S+SH*C))*XK/D/XK**3
      GO TO 10
    5 A(2) = (1,-CH*C)/XK**2
      GO TO 10
    6 A(3) = AA*(CH*S-SH*C)/XK/D
      GO TO 10
    7 A(4) = SH*S/D
      GO TO 10
    8 A(5) = AA*(CH*S+SH*C)/D
      GO TO 10
    9 A(6) = CH*C
      GO TO 10
   11 A(7) =0,
      THETA=0,
      GO TO 15
   10 IF(NN-6)12,12,13
   12 THETA = A(NN)
      GO TO 15
   13 THETA = -A(IR+2)    * XK**2
  107 FORMAT(9E12,4)
   15 RETURN
  101 FORMAT(5E14,8)
      END
```

Function THETA(XW, XT, M, NG)



26. Subroutine TRANSV

a) Abstract:

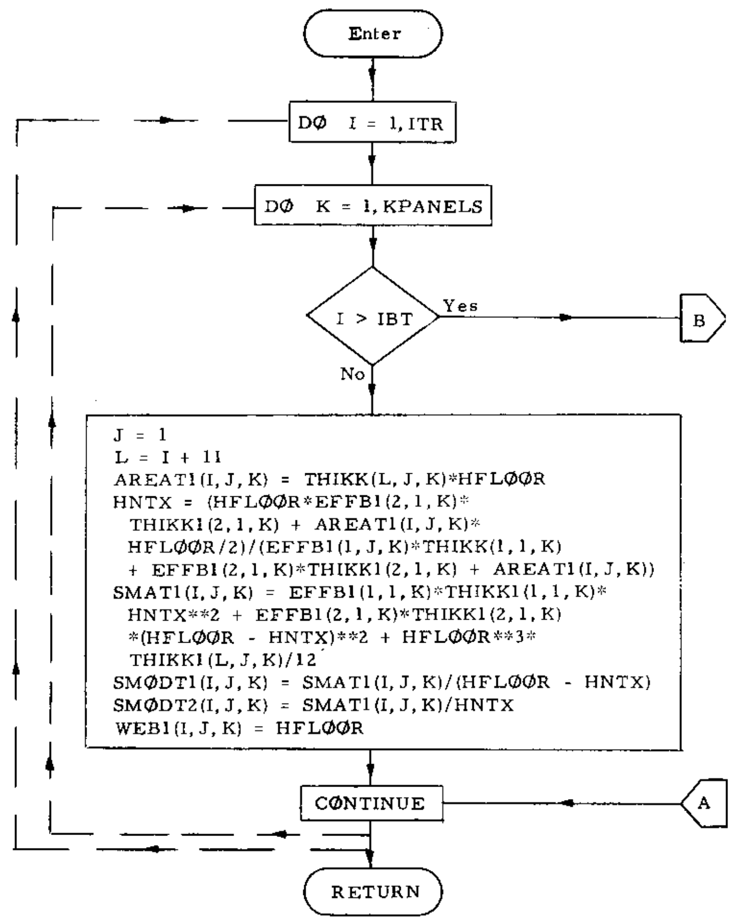
Subroutine TRANSV is called from subroutine SECTION. It calculates the section moduli of all transverses.

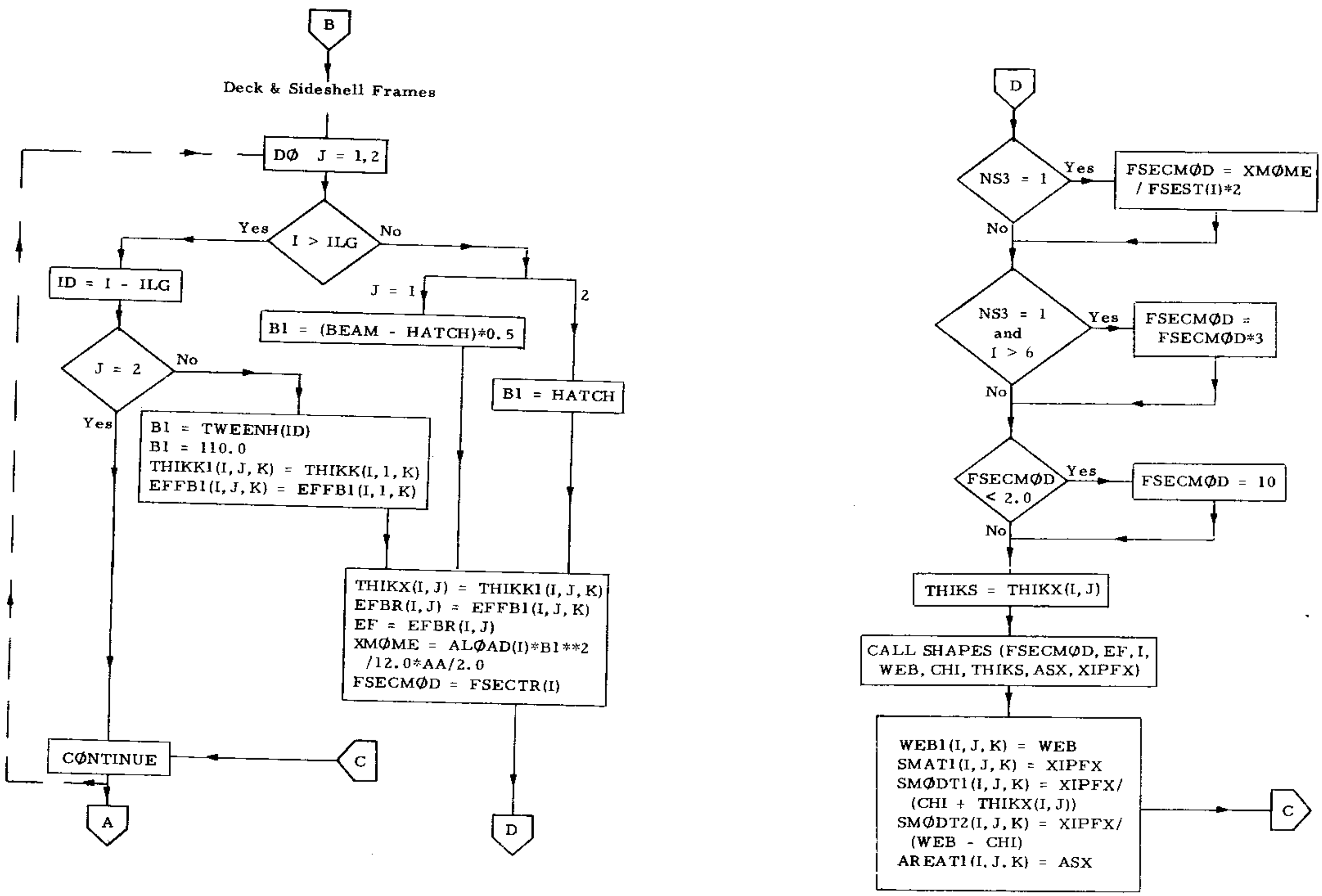
b) Description:

The cross sectional area, the second central moment of area and the section modulus of O. T. floors and N. T. floors are computed directly in subroutine TRANSV. For all other transverses and frames the above quantities are obtained from subroutine SHAPES.

```
      SUBROUTINE TRANSV(NS3)
      COMMON/TR/ FSECTR(17),FSEST(17),FSEST1(17),SMODT2(17,2,3),SMODT3(
     1 17,2,5)
      COMMON /E/ IBT,E
      COMMON /B/ A,B,NDECKS,IPLATE,PSL,PRM,HMAIN,HNEUT,TESL,TEST,SESL,
     1     SEST,        KPANELS,BEAM,XNG,HATCH,THIKK,THIK*1,EFFBR,EFFB1,
     2     EFFW,EFFW1,HFLOOR,TWEENH,THIKX,EFBR,WEB,WEB1,XIPFX,CHI,
     3     XIS, AREAT1,SMAL1,SMODL1,WEBL1,SUMAREAP,SUMAREAL,SUMMP,SUMML,
     4     AREA,SUMAREA,SUMMOM,WPL,XPANEL,WBAYLP,XNST,NST,ITR,
     5     WTR,BAYTR,SUMWT,SUMWTR,LLGX,  YIELD,WBAYTR,XLABOR,PLCOST,
     6     COST,COSTMIN,GNOT,GNNT,KNEUT,SUMSMAP,SUMSMAL,SUMSMA,CODE,
     7WAVEH,PRHEAD,DRAFT,ALOAD,XLHOLD,XLPANEL,AAA,BBB,AA,BB,DELTESL,
     8      SUMSMA1,ZP,ZL,ILG,AREAL1,ASX,  SMAT1,SMODT1
      DIMENSION ZP(17),SESL(17),SEST(17),TESL(17),TEST(17),XNG(7),THIKK(
     117),        THIKK1(17,2,5),EFFBR(17),EFFW(17),EFFB1(17,2,5),EFFW1(17
     2,2,5),AREA(17,5),TWEENH(5),SMAT1(12,2,5),SMODT1(12,2,5),THIKX(12,2
     3),EFBR(12,2),AREAT1(12,2,5),WEB1(12,2,5),SMAL1(17,2,3),SMODL1(17,2
     4,3),WEBL1(7,2,3),AREAL1(7,2,3),SUMAREAP(5),SUMARE L(5),SUMMP(5),SU
     5MML(5),SUMAREA(5),SUMMOM(5),WPL(5),XLPANEL(5),SUMWTR(5),WTR(12),ZL
     6(7),TIKK1(17,5),CODE(17),ALOAD(17),SUMSMAP(5),SUMSMAL(5),SUMSMA(5)
      DO 71 I = 1,ITR
      DO 71 K = 1,KPANELS
 78   IF(I.GT.IBT) GO TO 73
 72   J= 1
      L=I+11
      AREAT1(I,J,K) = THIKK1(L,J,K) * HFLOOR
      HNTX=(HFLOOR*EFFB1(2,1,K)*THIKK1(2,1,K)+AREAT1(I,J,K)*HFLOOR/2.)/(
     1EFFB1(1,J,K)*THIKK1(1,1,K)+EFFB1(2,1,K)*THIKK1(2,1,K)+AREAT1(I,J,K
     2)  )
      SMAT1(I,J,K) = EFFB1(1,1,K)* THIKK1(1,1,K) * HNTX**2 +
     1      EFFB1(2,1,K)* THIKK1(2,1,K)* (HFLOOR -HNTX)** 2
     2  +HFLOOR**3*THIKK1(L,J,K)/12.
      SMODT1(I,J,K) = SMAT1(I,J,K) / (HFLOOR - HNTX)
        SMODT2(I,J,K)=SMAT1(I,J,K)/HNTX
      WEB1 (I,J,K) = HFLOOR
      GO TO 71
      DECK AND SIDESHELL FRAMES
 73   DO 75 J=1,2
      IF(I.GT.ILG) GO TO 83
105   GO TO (91,92) J
 91   B1= (BEAM - HATCH) *0.5
      GO TO 93
 92   B1= HATCH
      GO TO 93
 83   ID = I - ILG
      IF(J.EQ.2)  GO TO 75
      B1= TWEENH(ID)
      B1=110.
      THIKK1(I,J,K) = THIKK1(I,1,K)
      EFFB1(I,J,K) = EFFB1(I,1,K)
 93   THIKX(I,J) = THIKK1(I,J,K)
      EFBR(I,J) = EFFB1(I,J,K)
      EF=EFBR(I,J)
      XMOME =  ALOAD(I)*B1**2 / 12.0 *AA/2.
      FSECMOD=FSECTR(I)
      IF( NS3.EQ.1)   FSECMOD=XMOME/FSEST(I)  *2.0
      IF(NS3.EQ.1.AND.I.GT.6)  FSECMOD=
        IF(FSECMOD.LT. 2.)    FSECMOD
900   FORMAT(/,*  FSECMOD= *,E10.4 ,*
      THIKS=THIKX(I,J)
      CALL       SHAPES  (FSECMOD.EF ,I
      WEB1(I,J,K) = WEB
      SMAT1(I,J,K)= XIPFX
      SMODT1(I,J,K)= XIPFX/ (CHI+THIKX
      SMODT2(I,J,K)= XIPFX/ (WEB-CHI)
      AREAT1(I,J,K)=ASX
 75   CONTINUE
 71   CONTINUE
      RETURN
      END
```

Subroutine TRANSV (NS3)





B
Deck & Sideshell Frames
DØ J = 1,2
I > ILG
Yes
No
ID = I - ILG
J = 2
No
Yes
B1 = TWEENH(ID)
B1 = 110.0
THIKK1(I, J, K) = THIKK(I, 1, K)
EFFB1(I, J, K) = EFFB1(I, 1, K)
J = 1
2
B1 = (BEAM - HATCH)*0.5
B1 = HATCH
THIKX(I, J) = THIKK1(I, J, K)
EFBR(I, J) = EFFB1(I, J, K)
EF = EFBR(I, J)
XMØME = ALØAD(I)*B1**2
/12.0*AA/2.0
FSECMØD = FSECTR(I)
CØNTINUE
C
A
D
D
NS3 = 1
Yes
No
FSECMØD = XMØME
/ FSEST(I)*2
NS3 = 1
and
I > 6
Yes
No
FSECMØD =
FSECMØD*3
FSECMØD
< 2.0
Yes
No
FSECMØD = 10
THIKS = THIKX(I, J)
CALL SHAPES (FSECMØD, EF, I,
WEB, CHI, THIKS, ASX, XIPFX)
WEB1(I, J, K) = WEB
SMAT1(I, J, K) = XIPFX
SMØDT1(I, J, K) = XIPFX/
(CHI + THIKX(I, J))
SMØDT2(I, J, K) = XIPFX/
(WEB - CHI)
AREAT1(I, J, K) = ASX
C

27. Subroutine XLOAD

a) Abstract:

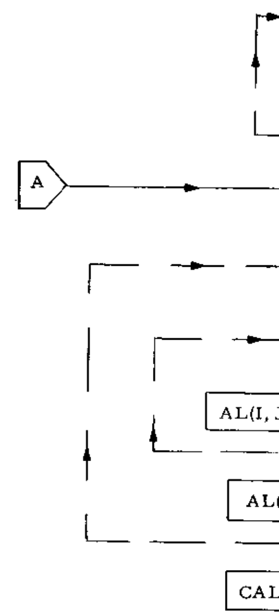
This subroutine is called from program TRANSHIP. It calculates the load transforms for the grillage system.

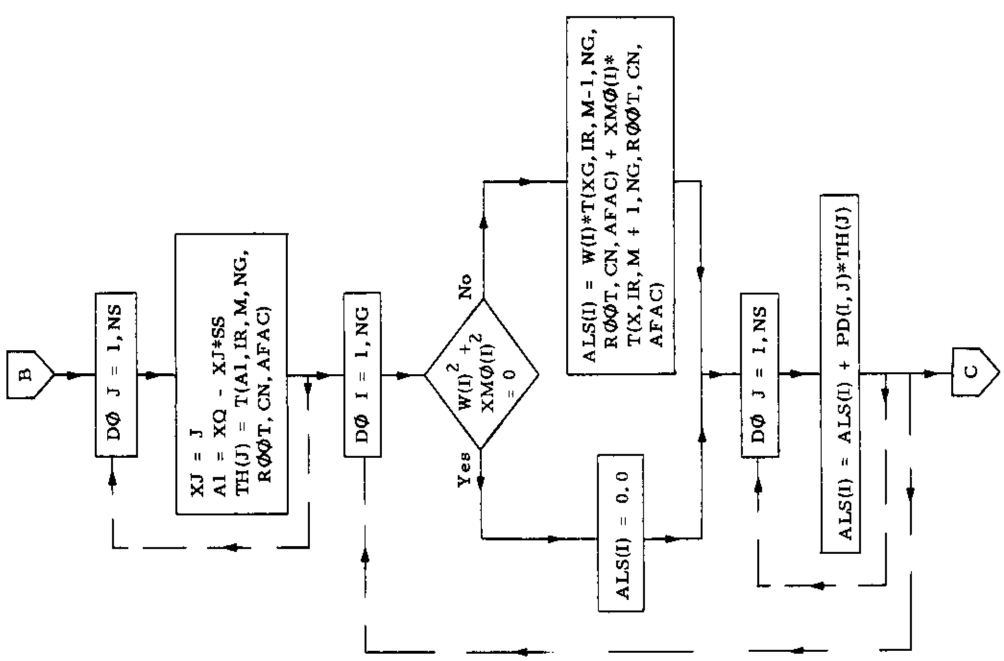
b) Terms specific to this subroutine:

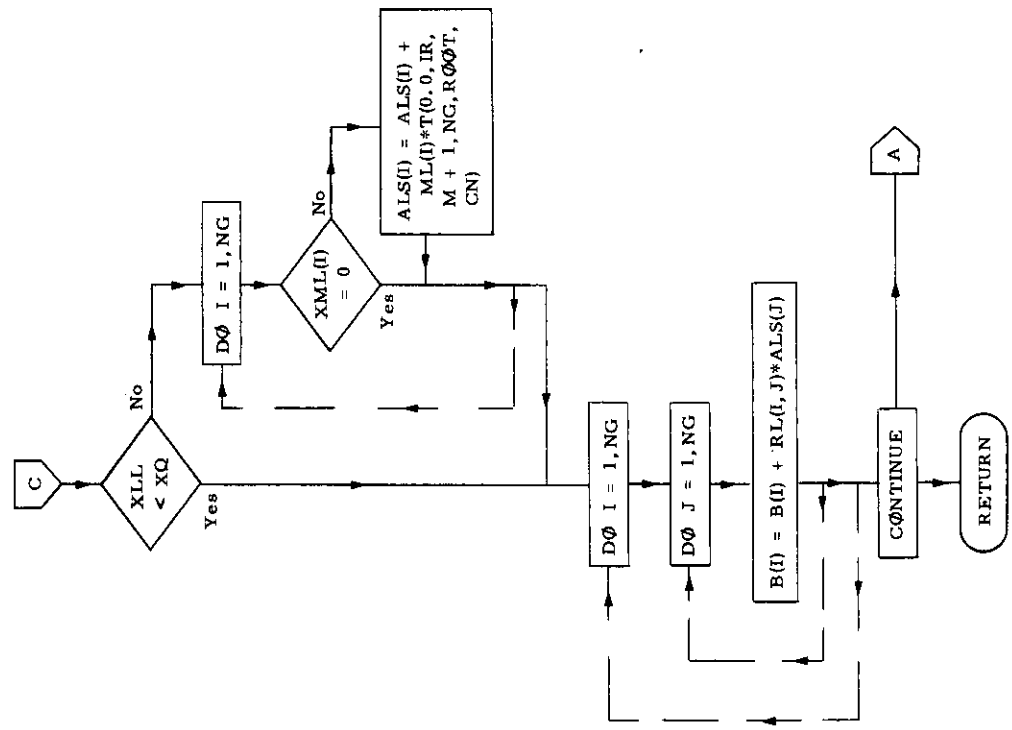
FORTRAN Term	Definition
A	A matrix
AL	Characteristic matrix
B	Load matrix
M	Index of Nielsen's function $N_m(u)$
XQ	x-distance in Nielsen's function

Subroutine XLØAD(A, M,

```
      SUBROUTINE XLOAD(A,M,B,XQ)
      COMMON /E/ IOT,E
      COMMON   ALFA, AX, CC, CN, DEFL,     GP, H, JJ, K, NG, NS, PD,
     1ROOT, S, SS, V, VV, W, X, XID, XII, XL, XLL, XML, XMO, XX,RR,SCAL
     2 ,AFAC,NNN
      DIMENSION  ALFA(15,15),AX( 9,9 ),CC(10),DEFL( 9),PD(15,40),
     1ROOT( 9),S(15),V(15),VV(15),W(15),XID(15),XML(15),XMO(15),XX(40),
     2XII(40),A(9,9),B(9),AL(9,9),RL(9,9),TH(40),ALS(9)
      XQ=X
      DO 500 J = 1,NG
      B(J) = 0.0
  500 CONTINUE
      DO 5000 IR=1,NG
      DO 4000 I=1,NG
      DO 3000 J=1,NG
      AL(I,J)=-A(I,J)/ROOT(IR)
 3000 CONTINUE
      AL(I,I)=AL(I,I)+1.0
 4000 CONTINUE
       CALL EV(AL,NG,RL,CC)
      DO 4500 J=1,NS
      XJ = J
      A1 = XQ- XJ*SS
 4400 TH(J) = T(A1, IR, M, NG, ROOT, CN,AFAC)
 4500 CONTINUE
      DO 4600 I=1,NG
      IF(W(I)**2  +  XMO(I)  **2)   4530,4520,4530
 4520 ALS(I) = 0.0
      GO TO 4540
 4530 ALS(I) = W(I) * T(XQ,IR,M-1,NG, ROOT, CN,AFAC) + XMO(I) *
     1T(X,IR,M+1,NG, ROOT, CN,AFAC)
 4540 DO 4580  J = 1,NS
      ALS(I) = ALS(I) + PD(I,J) * TH(J)
 4580 CONTINUE
 4600 CONTINUE
      IF (XLL -XQ)   4800,4650,4650
 4650 DO 4700 I=1,NG
      IF (XML(I))  4680,4700,4680
 4680 ALS(I) = ALS(I) + XML(I) * T(0.0, IR,M+1,NG, ROOT,  CN)
 4700 CONTINUE
 4800 DO 4850 I=1,NG
      DO 4850 J=1,NG
      B(I)=B(I)+RL(I,J)*ALS(J)
 4850 CONTINUE
 5000 CONTINUE
      RETURN
      END
```




B
DØ J = 1, NS
XJ = J
A1 = XΩ - XJ*SS
TH(J) = T(A1, IR, M, NG, RØØT, CN, AFAC)
DØ I = 1, NG
W(I)^2 + XMØ(I)^2 = 0
Yes
No
ALS(I) = 0.0
ALS(I) = W(I)*T(XG, IR, M-1, NG, RØØT, CN, AFAC) + XMØ(I)* T(X, IR, M + 1, NG, RØØT, CN, AFAC)
DØ J = 1, NS
ALS(I) = ALS(I) + PD(I, J)*TH(J)
C


C
XLL < XΩ
Yes
No
DØ I = 1, NG
XML(I) = 0
Yes
No
ALS(I) = ALS(I) + ML(I)*T(0.0, IR, M + 1, NG, RØØT, CN)
DØ I = 1, NG
DØ J = 1, NG
B(I) = B(I) + RL(I, J)*ALS(J)
CØNTINUE
RETURN
A

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) NATIONAL ENGINEERING SCIENCE COMPANY	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

Program "Tranship" - A Computer Program for the Design of the Midship Section of a Transversely-Framed Dry Cargo Ship, Part II

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Final

5. AUTHOR(S) (First name, middle initial, last name)

Manley St. Denis

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
December, 1971	144	-

8a. CONTRACT OR GRANT NO. N00024-68-C-5403 b. PROJECT NO. c. Ship Structure Committee Research Project SR-175 d.	9a. ORIGINATOR'S REPORT NUMBER(S) 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) SSC-216

10. DISTRIBUTION STATEMENT

Distribution of this document, SSC-216, is unlimited

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Naval Ship Engineering Center

13. ABSTRACT

This report presents the computer program corresponding to the method of design expounded in the Ship Structure Committee Report SSC-215, "A Guide for the Synthesis of Ship Structures - Part One - The Midship Hold of a Transversely-Framed, Dry Cargo Ship". The program consists in an executive routine, called TRANSHIP, and twenty seven subroutines.

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6801

UNCLASSIFIED
Security Classification

UNCLASSIFIED
Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT

DD FORM 1 NOV 65 1473 (BACK)
(PAGE 2)

☆U.S. GOVERNMENT PRINTING OFFICE: 1972 724-534/191 1-3

UNCLASSIFIED
Security Classification

SHIP RESEARCH COMMITTEE
Maritime Transportation Research Board
National Academy of Sciences-National Research Council

The Ship Research Committee has technical cognizance of the inter-agency Ship Structure Committee's research program:

PROF. R. A. YAGLE, Chairman, *Prof. of Naval Architecture, University of Michigan*

DR. H. N. ABRAMSON, *Director, Dept. of Mechanical Sciences, Southwest Research Inst.*

MR. W. H. BUCKLEY, *Coordinator of Hydrofoil Structural Research, Naval Ship Research & Development Center*

MR. E. L. CRISCUOLO, *Senior Nondestructive Testing Specialist, Naval Ordnance Laboratory*

DR. W. D. DOTY, *Research Consultant, U.S. Steel Corporation*

PROF. J. E. GOLDBERG, *School of Civil Engineering, Purdue University*

PROF. W. J. HALL, *Prof. of Civil Engineering, University of Illinois*

MR. J. E. HERZ, *Chief Structural Design Engineer, Sun Shipbuilding & Dry Dock Company*

MR G. E. KAMPSCHAEFER, JR., *Manager, Application Engineering, ARMCO Steel Corporation*

MR. R. C. STRASSER, *Director of Research, Newport News Shipbuilding & Dry Dock Company*

CDR R. M. WHITE, USCG, *Chief, Applied Engineering Section, U.S. Coast Guard Academy*

MR. R. W. RUMKE, *Executive Secretary, Ship Research Committee*

Advisory Group II, "Ship Structural Design" prepared the prospectus, evaluated the proposals, monitored the progress of the work, and reviewed this report.

MR. J. E. HERZ, Chairman, *Chief Structural Design Engineer, Sun Shipbuilding & Dry Dock Company*

MR. C. M. COX, *Asst. Naval Architect, Newport News Shipbuilding & Dry Dock Company*

MR. C. R. CUSHING, *President, Cushing & Nordstrom, Inc.*

PROF. J. E. GOLDBERG, *School of Engineering, Purdue University*

PROF. J. R. PAULLING, JR., *Chairman of Dept. of Naval Architecture, Univ. of California*

MR. D. P. ROSLMAN, *Naval Architect, Hydronautics, Inc.*

CDR R. M. WHITE, USCG, *Chief, Applied Engineering Section, U.S. Coast Guard Academy*

SHIP STRUCTURE COMMITTEE PUBLICATIONS

These documents are distributed by the National Technical Information Service, Springfield, Va. 22151. These documents have been announced in the Clearinghouse Journal U.S. Government Research & Development Reports (USGRDR) under the indicated AD numbers.

SSC-214, *Ship Response Instrumentation Aboard the Container Vessel S. S. BOSTON: Results from Two Operational Seasons in North Atlantic Service* by J. Q. Cragin. 1970. AD 712187.

SSC-215, *A Guide for the Synthesis of Ship Structures Part One - The Midship Hold of a Transversely-Framed Dry Cargo Ship* by Manley St. Denis. 1970.. AD 717357.

SSC-216, *Program TRANSHIP - A Computer Program for the Design of the Midship Section of a Transversely-Framed Dry Cargo Ship, Part II* - by Manley St. Denis. 1971.

SSC-217, *Compressive Strength of Ship Hull Girders - Part I - Unstiffened Plates* by H. Becker, R. Goldman, and J. Pozerycki. 1971. AD 717590.

SSC-218, *Design Considerations for Aluminum Hull Structures: Study of Aluminum Bulk Carriers* by C. J. Altenburg and R. J. Scott. 1971. AD 729021.

SSC-219, *Crack Propagation and Arrest in Ship and Other Steels* by G. T. Hahn, R. G. Hoagland, P. N. Mincer, A. R. Rosenfield, and M. Sarrate. 1971. AD 731674.

SSC-220, *A Limited Survey of Ship Structural Damage* by S. Hawkins, G. H. Levine, and R. Taggart. 1971. AD 733085

SSC-221, *Response of the Delta Test to Specimen Variables* by L. J. McGeady. 1971. AD 733086.

SSC-222, *Catamarans - Technological Limits to Size and Appraisal of Structural Design Information and Procedures* by N. M. Maniar, and W. P. Chiang. 1971. AD 733844

SSC-223, *Compressive Strength of Ship Hull Girders - Part II - Stiffened Plates* by H. Becker, A. Colao, R. Goldman, and J. Pozerycki. 1971. AD 733811

SSC-224, *Feasibility Study of Glass Reinforced Plastic Cargo Ship* by R. J. Scott, and J. H. Sommella. 1971. AD 735113