PROGRAM BEAM

!

! PROGRAM FOR STRUCTURAL ANALYSIS OF A STRAIGHT BEAM.

!

REAL :: X(51),E(50),SMA(50),ESTIFF(4,4),OSTIFF(102,103),

& DELTA(102),VTH(2,51),F(102),L(50),LOAD(2,51)

INTEGER :: NREST(51),NPI(50),NPJ(50)

EQUIVALENCE (F(1),LOAD(1,1)),(DELTA(1),VTH(1,1))

OPEN(5,FILE="DATA")

OPEN(6,FILE="RESULTS")

!

! INPUT AND TEST THE NUMBER OF ELEMENTS.

READ(5,\*) NEL

IF(NEL < 1 .OR. NEL > 50) THEN

WRITE(6,61)

61 FORMAT("NUMBER OF ELEMENTS OUTSIDE THE RANGE 1 TO 50 - STOP")

STOP

END IF

WRITE(6,62)

62 FORMAT("STRUCTURAL ANALYSIS OF A STRAIGHT BEAM")

!

! INPUT THE NODAL POINT DATA.

NNP=NEL+1

READ(5,\*) (I,NREST(I),X(I),LOAD(1,I),LOAD(2,I),N=1,NNP)

!

! INPUT THE ELEMENT DATA.

READ(5,\*) (M,NPI(M),NPJ(M),E(M),SMA(M),N=1,NEL)

!

! PREPARE TO SUM THE OVERALL STIFFNESS COEFFICIENTS.

NEQN=2\*NNP

Each row in turn: DO IROW=1,NEQN

Each column in turn: DO ICOL=1,NEQN

OSTIFF(IROW,ICOL)=0.

END DO Each column in turn

END DO Each row in turn

!

! FORM THE STIFFNESS MATRIX FOR EACH ELEMENT.

Each element in turn: DO M=1,NEL

I=NPI(M)

J=NPJ(M)

L(M)=X(J)-X(I)

FACT=E(M)\*SMA(M)/L(M)\*\*3

ESTIFF(1,1)=FACT\*12.

ESTIFF(1,2)=FACT\*6.\*L(M)

ESTIFF(1,3)=-ESTIFF(1,1)

ESTIFF(1,4)=ESTIFF(1,2)

ESTIFF(2,1)=ESTIFF(1,2)

ESTIFF(2,2)=FACT\*4.\*L(M)\*\*2

ESTIFF(2,3)=-ESTIFF(2,1)

ESTIFF(2,4)=FACT\*2.\*L(M)\*\*2

!

Columns of third row: DO ICE=1,4

ESTIFF(3,ICE)=-ESTIFF(1,ICE)

END DO Columns of third row

!

Columns of fourth row: DO ICE=1,3

ESTIFF(4,ICE)=ESTIFF(ICE,4)

END DO Columns of fourth row

ESTIFF(4,4)=ESTIFF(2,2)

!

! ADD ELEMENT STIFFNESSES TO OVERALL STIFFNESSES.

Each row in turn: DO IRE=1,4

Each column in turn: DO ICE=1,4

IF(IRE < 3)IROW=2\*(I-1)+IRE

IF(IRE >= 3)IROW=2\*(J-1)+IRE-2

IF(ICE < 3)ICOL=2\*(I-1)+ICE

IF(ICE >= 3) ICOL=2\*(J-1)+ICE-2

OSTIFF(IROW,ICOL)=OSTIFF(IROW,ICOL)+ESTIFF(IRE,ICE)

END DO Each column in turn

END DO Each row in turn

!

END DO Each element In turn

!

! APPLY THE RESTRAINTS.

Each node in turn: DO I=1,NNP

!

! ZERO DEFLECTION.

IF(NREST(I) == 1 .OR. NREST(I) == 3) THEN

IROW=2\*(I-1)+1

Each column In turn: DO ICOL=1,NEQN

IF(ICOL /= IROW) OSTIFF(IROW,ICOL)=0.

END DO Each column in turn

LOAD(1,I)=0.

END IF

!

! ZERO ROTATION.

IF(NREST(I) == 2 .OR. NREST(I) == 3) THEN

IROW=2\*(I-1)+2

Each column In turn: DO ICOL=1,NEQN

IF(ICOL /= IROW) OSTIFF(IROW,ICOL)=0.

END DO Each column in turn

LOAD(2,I)=0.

END IF

!

END DO Each node in turn

!

! EXTEND THE OVERALL STIFFNESS MATRIX TO INCLUDE THE FORCE VECTOR.

Each row in turn: DO IROW=1,NEQN

OSTIFF(IROW,NEQN+1)=F(IROW)

END DO Each row in turn

!

! SOLVE THE LINEAR EQUATIONS.

CALL ELIMIN(OSTIFF,DELTA,NEQN,102,103,IFLAG)

!

! OUTPUT THE RESULTS.

WRITE(6,63) (M,NPI(M),NPJ(M),E(M),SMA(M),L(M),M=1,NEL)

63 FORMAT(/" M I J MODULUS 2ND MOM AREA LENGTH" /

& (3I5,3E12.4))

WRITE(6,64) (I,NREST(I),X(I),(LOAD(N,I),VTH(N,I),N=1,2),I=1,NNP)

64 FORMAT(/" I REST X LOAD DEFLN MOMENT &

& ROTATION" / (2I5,5E12.4))

!

END PROGRAM BEAM

SUBROUTINE ELIMIN(A,X,NEQN,NROW,NCOL,IFLAG)

!

! SUBROUTINE FOR SOLVING SIMULTANEOUS LINEAR EQUATIONS BY GAUSSIAN

! ELIMINATION WITH PARTIAL PIVOTING.

!

REAL :: A(NROW,NCOL),X(NROW)

!

! INITIALIZE ILL-CONDITIONING FLAG.

IFLAG=0

!

! SCALE EACH EQUATION TO HAVE A MAXIMUM COEFFICIENT MAGNITUDE OF UNITY.

JMAX=NEQN+1

Each equation in turn: DO I=1,NEQN

AMAX=0.

Search for maximum: DO J=1,NEQN

ABSA=ABS(A(I,J))

IF(ABSA > AMAX) AMAX=ABSA

END DO Search for maximum

!

Scale coefficients: DO J=1,JMAX

A(I,J)=A(I,J)/AMAX

END DO Scale coefficients

!

END DO Each equation in turn

!

! COMMENCE ELIMINATION PROCESS.

Eliminate each variable in turn: DO K=1,NEQN-1

!

! SEARCH LEADING COLUMN OF THE COEFFICIENT MATRIX FROM THE DIAGONAL

! DOWNWARDS FOR THE LARGEST VALUE.

IMAX=K

Search for largest value: DO I=K+1,NEQN

IF(ABS(A(I,K)) > ABS(A(IMAX,K))) IMAX=I

END DO Search for largest value

!

! IF NECESSARY, INTERCHANGE EQUATIONS TO MAKE THE LARGEST COEFFICIENT

! BECOME THE PIVOTAL COEFFICIENT.

IF(IMAX /= K) THEN

Interchange coefficients: DO J=K,JMAX

ATEMP=A(K,J)

A(K,J)=A(IMAX,J)

A(IMAX,J)=ATEMP

END DO Interchange coefficients

END IF

!

! ELIMINATE X(K) FROM EQUATIONS (K+1) TO NEQN, FIRST TESTING FOR

! EXCESSIVELY SMALL PIVOTAL COEFFICIENT (ASSOCIATED WITH A SINGULAR

! OR VERY ILL-CONDITIONED MATRIX).

IF(ABS(A(K,K)) < 1.E-5) THEN

IFLAG=1

RETURN

END IF

Each of remaining equations: DO I=K+1,NEQN

FACT=A(I,K)/A(K,K)

Modify coefficients: DO J=K,JMAX

A(I,J)=A(I,J)-FACT\*A(K,J)

END DO Modify coefficients

END DO Each of remaining equations

!

END DO Eliminate each variable in turn

!

! SOLVE THE EQUATIONS BY BACK SUBSTITUTION, FIRST TESTING

! FOR AN EXCESSIVELY SMALL LAST DIAGONAL COEFFICIENT.

IF(ABS(A(NEQN,NEQN)) < 1.E-5) THEN

IFLAG=1

RETURN

END IF

X(NEQN)=A(NEQN,JMAX)/A(NEQN,NEQN)

Then each unknown in turn backwards: DO I=NEQN-1,1,-1

SUM=A(I,JMAX)

Sum products: DO J=I+1,NEQN

SUM=SUM-A(I,J)\*X(J)

END DO Sum products

X(I)=SUM/A(I,I)

END DO Then each unknown in turn backwards

RETURN

END SUBROUTINE ELIMIN