# เอกสารอ้างอิง


1. ปรีชา แสงอาสภวิริยะ ภาวดี วรปัญญาวัฒนา วาสุเทพ ธะประสม และวีรยุทธ วงษ์ศิริ ภาษาคอมพิวเตอร์ฟอร์แทน 77 ภาควิชาคณิตศาสตร์ สถิติ คณะศิลปศาสตร์ มหาวิทยาลัยธรรมศาสตร์ พิมพ์ครั้งที่ 1, 2527

2. ชยาทิตย์ วัฒนาวิกย์กิจ มาตรฐานการออกแบบสะพาน, หน้า 1-117. ภาควิชาวิศวกรรมศาสตร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยขอนแก่น พิมพ์ ครั้งที่ 1 2525

3. ดำรงค์ หมดตี การวิเคราะห์โครงสร้าง, หน้า 4.1-5.23 ภาควิชาวิศวกรรมศาสตร์ คณะวิศวกรรมศาสตร์มหาวิทยาลัยแก่นพิมพ์ครั้งที่ 2 2526

4. Chu-kia Wang, Ph.d., Statically Indeterminate Structure, P 137 - 280 ,Tosho Printing Co Ltd., Tokyo, 1953

5. C.P.Heins and D.A Firmage, Design of Modern Steel highway Bridge, pp 1-154, John Wiley Sons, Inc,. New York, 1979

6. สนั่น เจริญเผ่า และวินิต ช่อวิเชียร คอนกรีตเสริมเหล็ก, หน้า 39-193 ภาควิชาวิศวกรรมศาสตร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย พิมพ์ครั้งที่ 6, 2527

7. George Winter And Arther H. Nilson, Design of Conrete Structures, pp 560 - 594. Tosho Printing Co.Ltd., tokyo, 1981

8. Conrad P. Heins and Richard A Lawrie, Design of Modern Concrete Highway Gridge, pp 132-203, A Wiley - Intersection Publication, New Youk, 1984.

9. James A. Moe, Ronald Reagan, J.E. Mc Mahon, R.J. Datel, Manaul of Bridge Design Practice, pp 1-100, 3 rd ed., 1971

10. Thomas Richard Mc. Calla, Introduction to Numerical Methods and Fortran Programming, pp 263 - 299. John Wiley Sons, Incl, New York, 1967.

11. Frederick W. Taylor, Sanford E. Thompson and Edward Smulski, Reiforced - Concrete Bridges, pp 194 - 216, John Wiley Sons, Inc., New York, 1955

12. เพิ่มศรี ลัดดาวัลย์ สูตรโครงสร้าง, หน้า 1-15, Cosmo Technological Consultants Co.Ltd., กรุงเทพฯ, พิมพ์ครั้งที่ 1, 2528

# ภาคผนวก ก.

## โปรแกรมคอมพิวเตอร์

## โปรแกรมคอมพิวเตอร์

```
C
C  ***************************************************************************
C  *      BEAM   DESIGN  FOR  BRIDGE  STRUCTURE WITH VARY CROSS SECTION       *
C  *                        USE  FORTRAN 77                                   *
C  *                    DATE-WRITTEND  26 FEBUARY 1997                        *
C  ***************************************************************************
C
C
C                    &----------------------&
C                    * INITIALIZE CONSTANT  *
C                    &----------------------&
C                    *----------------------*
C                    *      OPEN FILES      *
C                    *----------------------*
C
C  OPEN INPUT FILE1 Material properties
      OPEN ( 5,FILE='DATA1.TXT',STATUS='OLD',ACCESS='SEQUENTIAL')
C
C  OPEN OUTPUT FILE1 for reinforced concrete design
      OPEN ( 6,FILE='OUT1.TXT')
      CLOSE( 6,STATUS='DELETE')
      OPEN ( 6,FILE='OUT1.TXT', STATUS='NEW',ACCESS='SEQUENTIAL')
C
C  OPEN OUTPUT FILE2 for moment & shear analysis
      OPEN (11,FILE='OUT2.TXT')
      CLOSE(11,STATUS='DELETE')
      OPEN (11,FILE='OUT2.TXT', STATUS='NEW',ACCESS='SEQUENTIAL')
C
C  OPEN OUTPUT FILE3 for number of steel
      OPEN (13,FILE='OUT3.TXT')
      CLOSE(13,STATUS='DELETE')
      OPEN (13,FILE='OUT3.TXT', STATUS='NEW',ACCESS='SEQUENTIAL')
C
C  OPEN FILE4
      OPEN (14,FILE='OUT4.TXT')
      CLOSE(14,STATUS='DELETE')
      OPEN (14,FILE='OUT4.TXT', STATUS='NEW',ACCESS='SEQUENTIAL')
C
C  OPEN BRIDGE DIMENSION INPUT DATA FILE
      OPEN (12,FILE='DATA2.TXT',STATUS='OLD',ACCESS='SEQUENTIAL')
C  OPEN INPUT MOMENT & SHEAR FILE
      OPEN (13,FILE='TRANS.DAT',STATUS='OLD',ACCESS='SEQUENTIAL')
C
C
C
C                    &----------------------&
C                    *      MAIN MENU       *
C                    &----------------------&
C
 100  CALL CLRSCR
      CALL PAGE (2)
      CALL HEAD (' ***********************************************')
      CALL HEAD (' *********          MAIN MENU          *********')
      CALL HEAD (' ***********************************************')
      CALL PAGE (2)
      CALL HEAD ('       1>  EDIT DATA MENU                       ')
      CALL PAGE (1)
      CALL HEAD ('       2>  PROCESS CALCULATION :                ')
      CALL HEAD ('           Then outputs are on file OUT1.TXT    ')
      CALL HEAD ('                              and  OUT2.TXT     ')
```

```
      CALL HEAD ('                                    and  OUT3.TXT          ')
      CALL PAGE (1)
      CALL HEAD ('        3>  QUIT to PRIME OPERATING SYSTEM                 ')
      CALL PAGE (2)
      CALL READI('                SELECT NUMBER  (1,2 or 3)  :   ',IWAY  )
      CALL PAGE (1)
      GOTO (1,2,3),IWAY
        GOTO 100
    1 CALL EDMENU
        GOTO 100
    2 CALL COMPUTE
        GOTO 100
    3 CALL PAGE (1)
      CALL HEAD (' ##################  END OF JOB  ################## ')
      CLOSE (5)
      CLOSE (6)
      CLOSE (11)
      CLOSE (14)
      CLOSE (18)
        STOP
        END
C
C
C
C   ***********************************
C   *         BLOCK  DATA             *
C   ***********************************
C
      BLOCK DATA
C
      COMMON /STESEC/ CR,ST(12,2),FORM(12,2)
      COMMON /MKSBAR/ KDIA(12)
      COMMON /REACTI/ REA
C
      DATA CR /2.0/
      DATA ST /0.00,0.25,0.375,0.50,0.625,0.75,0.875,1.00,
     +         1.123,1.27,1.41,1.693,
     *         0.00,0.05,0.11,0.20,0.31,0.44,0.60,0.79,
     +         1.00,1.27,1.56,2.25/
      DATA FORM /0.00,2.75,3.0,3.25,3.5,3.75,4.0,4.25,
     +           4.5,4.75,5.0,5.5,
     *           0.00,3.5,3.75,4.0,4.25,4.5,4.75,5.0,
     +           5.25,5.5,6.0,7.0/
      DATA KDIA /4,6,9,12,15,19,22,25,28,30,33,36/
      DATA REA  /0.0/
      END
C
C
C
C   ***********************************************************************
C   *                        S U B   P R O G R A M 1                      *
C   ***********************************************************************
C
C                  *-----------------------------------*
C                        SUBROUTINE   EDMENU
C                  *-----------------------------------*
C
C   EDITING DATA OF (STEEL,CONCRETE,BRIDGE SPEC.)
C
      CHARACTER  HEADI*52
```

```
      INTEGER    CHANGE, TYP
      COMMON /TITLE/ HEADI
      COMMON /MATER/ FCP, FSY
      COMMON /LCLASS/ LCLASS
      COMMON /STSIZE/ NOSB, NOST, NOSL
      COMMON /BRGDIA/ NUMTRA, WLANE, EGDAP, NOBEAM, CSL, PAREA, NUMSPA
      COMMON /PBEAM/ BWIDTH, SINTER
      COMMON /BEAM/ TYP(9), SL(9), NSC(9), SL1(9), SL2(9), DO(9), D1(9), D2(9)
      COMMON /SETTLE/ STM(10)
      COMMON /LOAD/DD1, DE1, DI1, DLL2, PLM, PLS
C  START  PROCESSING
C
      REWIND 5
      READ (5,1000) HEADI
      READ (5,1100) FCP, FSY, WLANE, GDSPC
      READ (5,1100) CSL, PAREA, EGDAP
      READ (5,1200) NOSB, NOST, NOSL, LCLASS, NUMTRA,
     +              NUMSPA, NOBEAM
      REWIND 12
      READ (12,1250) BWIDTH
      READ (12,1250) SINTER
      DO 10 I=1,9
   10 READ (12,1300) TYP(I), SL(I), NSC(I), SL1(I), SL2(I), DO(I), D1(I), D2(I)
      READ (12,1400) (STM(I), I=1,10)
C
C
C  EDIT MENU
C
      CHANGE = 0
  100 CALL CLRSCR
      CALL HEAD (' ************************************************ ')
      CALL HEAD (' **************   EDITTING  MENU   ************** ')
      CALL HEAD (' ************************************************ ')
      CALL PAGE (1)
      CALL HEAD ('            1>  Edit  Title     Heading             ')
      CALL PAGE (1)
      CALL HEAD ('            2>  Edit  Material  Properties          ')
      CALL PAGE (1)
      CALL HEAD ('            3>  Edit  Load      Class               ')
      CALL PAGE (1)
      CALL HEAD ('            4>  Edit  Bridge    Dimensions          ')
      CALL PAGE (1)
      CALL HEAD ('            5>  Edit  Beams     Dimensions          ')
      CALL PAGE (1)
      CALL HEAD ('            6>  Edit  Bars      Size                ')
      CALL PAGE (1)
      CALL HEAD ('            7>  Edit  Settlement  Size              ')
      CALL PAGE (1)
      CALL HEAD ('            8>  <<< Return To Main-Menu >>>         ')
      CALL PAGE (1)
      CALL READI('                SELECT NUMBER  (1,2,...,8) :  ', IWAY  )
      CALL PAGE (1) ,
      GOTO (1,2,3,4,5,6,7,8), IWAY
        GOTO 100
    1 CHANGE = 1
      CALL EDTIT
        GOTO 100
    2 CHANGE = 1
      CALL EDMAT
        GOTO 100
```

```
    3  CHANGE = 1
       CALL EDLDC
         GOTO 100
    4  CHANGE = 1
       CALL EDDIM
         GOTO 100
    5  CHANGE = 1
       CALL EDBMD
         GOTO 100
    6  CHANGE = 1
       CALL EDBAR
         GOTO 100
    7  CHANGE = 1
       CALL EDSTM
         GOTO 100
    8  IF (CHANGE .EQ. 1) THEN
       REWIND 5
       WRITE(5,1000) HEADI
       WRITE(5,1100) FCP,FSY,WLANE,GDSPC
       WRITE(5,1100) CSL,PAREA,EGDAP
       WRITE(5,1200) NOSB,NOST,NOSL,LCLASS,NUMTRA,
      +              NUMSPA,NOBEAM
       REWIND 12
       WRITE(12,1250) BWIDTH
       WRITE(12,1250) SINTER
       DO 20 I=1,9
   20  WRITE(12,1300) TYP(I),SL(I),NSC(I),SL1(I),SL2(I),D0(I),D1(I),D2(I)
       WRITE(12,1400) (STM(I),I=1,10)
       ENDIF
C
C
C
 1000 FORMAT (A)
 1100 FORMAT (8F10.3)
 1200 FORMAT (16I5)
 1250 FORMAT (F10.4)
 1300 FORMAT (I5,F10.4,I5,5F10.4)
 1400 FORMAT (10F10.3)
       RETURN
       END
C
C
C                    *-------------------------------------*
                              SUBROUTINE   EDTIT
C                    *-------------------------------------*
C  EDIT TITLE OF FILE # 5
       CHARACTER HEADI*52
       COMMON /TITLE/ HEADI
  100  CALL CLRSCR
       CALL PAGE (1)
       CALL HEAD (' ********************************************** ')
       CALL HEAD (' ************  TITLE HEADING EDIT  ************** ')
       CALL HEAD (' ********************************************** ')
       CALL PAGE (2)
       CALL HEAD ('          An Existing Title is shown below      ')
       CALL PAGE (2)
       CALL HEAD (HEADI)
       CALL PAGE (2)
       CALL HEAD ('          1>  ENTER  THE  NEW  TITLE            ')
       CALL PAGE (1)
```

```
      CALL HEAD ('          2>  EVERYTHING  IS  ALLRIGHT                ')
      CALL PAGE (2)
      CALL READI('                SELECT  NUMBER  ( 1 or 2 ) :  ',IWAY  )
      CALL PAGE (1)
      GOTO (1,2) ,IWAY
        GOTO 100
 1    CALL HEAD (' Enter the new Title you want to refer to later.... ')
      CALL PAGE (1)
      CALL READC(' Enter Char. : ',HEADI)
        GOTO 100
 2    RETURN
      END
C
C
C                  #-------------------------------------#
                         SUBROUTINE   EDMAT
C                  #-------------------------------------#
C  EDIT MATERIAL PROPERTIES
C
      COMMON /MATER/ FCP,FSY
C
 100  CALL CLRSCR
      CALL HEAD (' ********************************************** ')
      CALL HEAD (' ***********    MATERIAL  PROPERTIES    *********** ',
      CALL HEAD (' ********************************************** ')
      CALL PAGE (2)
      CALL FTOM (FCP,ZZ,6)
      CALL LISTR('  1>  Ultimate Strength of concrete = ',zz,  ' ksc. ')
      CALL PAGE (1)
      CALL FTOM (FSY,ZZ,6)
      CALL LISTR('  2>  Yield  Strength  of  Steel    = ',zz,  ' ksc. ')
      CALL PAGE (1)
      CALL HEAD ('  3>  EVERYTHING  IS  ALLRIGHT                      ')
      CALL PAGE (2)
      CALL READI('                SELECT  NUMBER  (1,2 or 3) :  ',IWAY  )
      CALL PAGE (1)
      GOTO (1,2,3) ,IWAY
        GOTO 100
 1    CALL READR('  Ultimate Strength of Concrete   (ksc.)  =  ',zz     )
      CALL MTOF (ZZ,FCP,6)
        GOTO 100
 2    CALL READR('  Yield  Strength of Steel        (ksc.)  =  ',zz     )
      CALL MTOF (ZZ,FSY,6)
        GOTO 100
 3    RETURN
      END
C
C
C                  #-------------------------------------#
                         SUBROUTINE   EDLOC
C                  #-------------------------------------#
C  EDIT  LOAD  CLASS
C
      COMMON /LCLASS/ LCLASS
C
 100  CALL CLRSCR
      CALL PAGE (1)
      CALL HEAD (' ********************************************** ')
      CALL HEAD (' *************   LOAD CLASS EDITING   ************* ')
      CALL HEAD (' ********************************************** ')
```

```
      CALL PAGE (2)
      IF (LCLASS .EQ. 1) GOTO 110
      IF (LCLASS .EQ. 2) GOTO 120
        GOTO 200
  110 CALL HEAD ('      =====> by now Load Class is HS20          ')
      CALL PAGE (2)
        GOTO 200
  120 CALL HEAD ('      =====> by now Load Class is HS15          ')
      CALL PAGE (2)
        GOTO 200
  200 CALL HEAD ('                1> HS20 Load Class              ')
      CALL PAGE (1)
      CALL HEAD ('                2> HS15 Load Class              ')
      CALL PAGE (1)
      CALL HEAD ('                3> EVERYTHING IS ALLRIGHT       ')
      CALL PAGE (2)
      CALL READI('                SELECT NUMBER (1,2 or 3) : ',IWAY )
      CALL PAGE (1)
      IF ((IWAY.NE.1).AND.(IWAY.NE.2)) GOTO 300
      LCLASS = IWAY
        GOTO 100
  300 RETURN
      END
C
C
C                  *-------------------------------------*
                       SUBROUTINE   EDDIM
C                  *-------------------------------------*
C  EDIT BRIDGE DIMENSIONS
C
      COMMON /BRGDIA/ NUMTRA,WLANE,EGDAP,NOBEAM,CSL,PAREA,NUMSPA
C
  100 CALL CLRSCR
      CALL HEAD (' ************************************************ ')
      CALL HEAD (' *********   BRIDGE DIMENSION EDITING   ********* ')
      CALL HEAD (' ************************************************ ')
      CALL PAGE (1)
      CALL LISTI('  1> Number of Traffic Lane       ',numtra,'      ')
      CALL PAGE (1)
      CALL FTOM (WLANE,ZZ,4)
      CALL LISTR('  2> Lane    Width                ',zz,    ' m.   ')
      CALL PAGE (1)
      CALL FTOM (EGDAP,ZZ,4)
      CALL LISTR('  3> Outer Dimension Appart Ex-Gir.',zz,   ' m.   ')
      CALL PAGE (1)
      CALL LISTI('  4> Number of Girder per section  ',nobeam,'     ')
      CALL PAGE (1)
      CALL FTOM (CSL,ZZ,4)
      CALL LISTR('  5> Overhanging Slab from Girder  ',zz,    ' m.   ')
      CALL PAGE (1)
      CALL FTOM (PAREA,zz,9)
      CALL LISTR('  6> Area of Parapet Family        ',zz,    ' sq.m.')
      CALL PAGE (1)
      CALL LISTI('  7> Number of span                ',NUMSPA,'      ')
      CALL PAGE (1)
      CALL FTOM (RR,ZZ,9)
      CALL LISTR('  8> Area of  Curb                 ',ZZ,    ' sq.m.')
      CALL PAGE (1)
      CALL FTOM (CB,ZZ,9)
      CALL LISTR('  9> Area of cross beam            ',ZZ,    ' sq.m.')
      CALL PAGE (1)
      CALL HEAD ('  10> EVERYTHING  IS  ALLRIGHT                    ')
      CALL PAGE (1)
      CALL READI('                SELECT NUMBER (1,2,...,10) : ',IWAY )
      CALL PAGE (1)
      GOTO (1,2,3,4,5,6,7,8,9,10),IWAY
```

```
      GOTO 100
    1 CALL READI('                    Number of Traffic Lane  =  ',NUMTRA)
      GOTO 100
    2 CALL READR('                          Lane Width  (m.)  =  ',ZZ    )
      CALL MTOF(ZZ,WLANE,4)
      GOTO 100
    3 CALL READR('  Outer dimension appart EX-Girder  (m.)  =  ',ZZ    )
      CALL MTOF (ZZ,EGDAP,4)
      GOTO 100
    4 CALL READI('              Number of Girder per section  =  ',NOBEAM)
      GOTO 100
    5 CALL READR(' Overhang Slab from c.g. of Girder  (m.)  =  ',ZZ    )
      CALL MTOF (ZZ,CSL,4)
      GOTO 100
    6 CALL READR('  Cross section Area of Parapet  (sq.m.)  =  ',ZZ    )
      CALL MTOF (ZZ,PAREA,9)
      GOTO 100
    7 CALL READI('                  Total Number of Span  =  ',NUMSPA)
      GOTO 100
    8 CALL READR(' Curb  area                      (sq.m.)  =  ',ZZ    )
      CALL MTOF (ZZ,RR,9)
      GOTO 100
    9 CALL READR(' Cross beam  area                (sq.m.)  =  ',ZZ    )
      CALL MTOF (ZZ,CB,9)
      GOTO 100
   10 RETURN
      END
C
C
C
C                  *---------------------------*
                       SUBROUTINE  EDBMD
C                  *---------------------------*
C
C
C  EDIT  BEAM  DIMENSIONS
C
      COMMON /BRGDIA/ NUMTRA,WLANE,EGDAP,NOBEAM,CSL,PAREA,NUMSPA
      COMMON /PBEAM/ BWIDTH,SINTER
C
  100 CALL CLRSCR
      CALL HEAD (' ************************************************** ')
      CALL HEAD (' *********    BEAM  DIMENSION  EDITING    ********* ')
      CALL HEAD (' ************************************************** ')
      CALL PAGE (2)
      CALL FTOM (BWIDTH,ZZ,4)
      CALL LISTR('        1>  Beam   Width          = ',ZZ,    ' m.   ')
      CALL PAGE (1)
      CALL FTOM (SINTER,ZZ,4)
      CALL LISTR('        2>  Length  of  Interval  = ',ZZ,    ' m.   ')
      CALL PAGE (1)
      CALL HEAD ('        3>  EDIT  EACH  SPAN                       ')
      CALL PAGE (1)
      CALL HEAD ('        4>  EVERY THING IS ALLRIGHT                ')
      CALL PAGE (2)
      CALL READI('                SELECT NUMBER (1,2,...,4)  :  ',IWAY  )
      CALL PAGE (1)
      GOTO (1,2,3,4),IWAY
      GOTO 100
    1 CALL READR('                        Beam  Width  (m.)  =  ',ZZ    )
      CALL MTOF (ZZ,BWIDTH,4)
      GOTO 100
    2 CALL READR('   Length of Segment Divides into  (m.)  =  ',ZZ    )
      CALL MTOF (ZZ,SINTER,4)
      GOTO 100
    3 CALL EDBMDS(NUMSPA)
      GOTO 100
```

```
    4   RETURN
          END
C
C
C*-----------------------------------*
        SUBROUTINE  EDBMDS(NUMSPA)
C*-----------------------------------*
  200   CALL CLRSCR
        CALL HEAD (' ************************************************************ ')
        CALL HEAD (' **********     BEAM  DIMENSION  EDITING     ********** ')
        CALL HEAD (' ************************************************************ ')
        CALL PAGE (2)
        DO  80 I = 1,NUMSPA
       WRITE (*,1000) I,I
 1000 FORMAT (16X,I3,'>  Edit  Span  No. ',I2)
   80   CALL PAGE (1)
        WRITE (*,1100) I
 1100 FORMAT (16X,I3,'> EVERYTHING IS ALLRIGHT                                 ')
        CALL PAGE (1)
        CALL READI('                    SELECT NUMBER  (1,2,3,...) :  ',IWAY  )
        IF (IWAY.GE.I) GOTO 11
        CALL PAGE (1)
        CALL DEPTH(IWAY)
        GOTO 200
    11  RETURN
          END
C
C
C-------------------------------
        SUBROUTINE  DEPTH(I)
C-------------------------------
C
        INTEGER TYP
        COMMON /PBEAM/ BWIDTH,SINTER
        COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D1(9),D2(9)
C
   300  CALL CLRSCR
        CALL HEAD (' ************************************************************ ')
        WRITE (*,333) I
   333  FORMAT(14X,' **************   SPAN  NUMBER  ',I2,'   ***************
       **')
        CALL HEAD (' ************************************************************ ')
        CALL PAGE (2)
        CALL LISTJ('     ====>   by now  Code of Beam Shape  = ',TYP(I)   )
        CALL PAGE (2)
        CALL HEAD ('     Code :        1 =  Straight haunch                 ')
        CALL PAGE (1)
        CALL HEAD ('                   2 =  Parabolar haunch                ')
        CALL PAGE (1)
        CALL HEAD ('                   3 =  Constant Cross Section          ')
        CALL PAGE (1)
        CALL HEAD ('   .  Command :    4 >  CONTINUE                        ')
        CALL PAGE (1)
        CALL HEAD ('                   5 >  RETURN                          ')
        CALL PAGE (2)
        CALL READI('                  SELECT NUMBER (1,2...,5)  :  ',IWAY  )
        CALL PAGE (1)
        GOTO (1,2,3,4,5),IWAY
        GOTO 300
     1  TYP(I) = IWAY
```

```
      GOTO 300
    2 TYP(I) = IWAY
      GOTO 300
    3 TYP(I) = IWAY
      GOTO 300
    4 GOTO (400, 400, 500), TYP(I)
    5 RETURN
C
C
C
  400 CALL CLRSCR
      CALL HEAD (' ************************************************** ')
      WRITE (*, 444) I
  444 FORMAT(14X, ' *************   SPAN  NUMBER  ', I2, '   ***************
     <*')
      CALL HEAD (' ************************************************** ')
      CALL PAGE (1)
      CALL FTOM (SL(I), ZZ, 4)
      CALL LISTR('     1>  Span  Length                = ', ZZ,    ' m    ')
      CALL PAGE (1)
      IZZ =  NSC(I)
      CALL LISTI('         Number of Section Consider = ', IZZ,   ' sect.')
      CALL PAGE (1)
      CALL FTOM (D1(I), ZZ, 4)
      CALL LISTR('     2>  Left side  Depth            = ', ZZ,    ' m.   ')
      CALL PAGE (1)
      CALL FTOM (DO(I), ZZ, 4)
      CALL LISTR('     3>  Centre     Depth            = ', ZZ,    ' m.   ')
      CALL PAGE (1)
      CALL FTOM (D2(I), ZZ, 4)
      CALL LISTR('     4>  Right side Depth            = ', ZZ,    ' m.   ')
      CALL PAGE (1)
      CALL FTOM (SL1(I), ZZ, 4)
      CALL LISTR('     5>  Left  side Length           = ', ZZ,    ' m.   ')
      CALL PAGE (1)
      CALL FTOM (SL2(I), ZZ, 4)
      CALL LISTR('     6>  Right side Length           = ', ZZ,    ' m.   ')
      CALL PAGE (1)
      CALL HEAD ('     7>  EVERY THING IS ALLRIGHT                        ')
      CALL PAGE (1)
      CALL READI('                 SELECT NUMBER (1,2,...,7)  :  ', IWAY  )
      CALL PAGE (1)
      GOTO (11, 12, 13, 14, 15, 16, 17), IWAY
        GOTO 400
   11 CALL READR('                 Span   Length       (m.)  =  ', ZZ     )
      CALL MTOF (ZZ, SL(I), 4)
      NSC(I) = SL(I)/SINTER + 1.5
        GOTO 400
   12 CALL READR('                 Left  side  Depth   (m.)  =  ', ZZ     )
      CALL MTOF (ZZ, D1(I), 4)
        GOTO 400
   13 CALL READR('                 Centre      Depth   (m.)  =  ', ZZ     )
      CALL MTOF (ZZ, DO(I), 4)
        GOTO 400
   14 CALL READR('                 Right  side  Depth  (m.)  =  ', ZZ     )
      CALL MTOF (ZZ, D2(I), 4)
        GOTO 400
   15 CALL READR('                 Left  side  Length  (m.)  =  ', ZZ     )
      CALL MTOF (ZZ, SL1(I), 4)
        GOTO 400
```

```
  16  CALL READR('                Right  side  Length  (m.)  =  ',ZZ    )
      CALL MTOF (ZZ,SL2(I),4)
        GOTO 400
  17  RETURN
C
C
 500  CALL CLRSCR
      CALL HEAD (' ********************************************** ')
      WRITE (*,555) I
 555  FORMAT(14X,' *************   SPAN  NUMBER  ',I2,'  *************
     **')
      CALL HEAD (' ********************************************** ')
      CALL PAGE (2)
      CALL FTOM (SL(I),ZZ,4)
      CALL LISTR('    1>  Span  Length                 = ',ZZ,   ' m.    ')
      CALL PAGE (1)
      IZZ =  NSC(I)
      CALL LISTI('        Number of Section Consider = ',IZZ,  ' sect.')
      CALL PAGE (1)
      CALL FTOM (D1(I),ZZ,4)
      CALL LISTR('    2>  Depth  of  Beam              = ',ZZ,   ' m.    ')
      CALL PAGE (1)
      CALL HEAD ('    3>  EVERY THING IS ALLRIGHT                        ')
      CALL PAGE (2)
      CALL READI('                SELECT NUMBER (1,2,...,3)  :  ',IWAY  )
      CALL PAGE (1)
      GOTO (21,22,23),IWAY
        GOTO 500
  21  CALL READR('                Span   Length        (m.)  =  ',ZZ    )
      CALL MTOF (ZZ,SL(I),4)
      SL1(I) = SL(I)
      SL2(I) = SL(I)
      NSC(I) = SL(I)/SINTER + 1.5
        GOTO 500
  22  CALL READR('                Depth  of  Beam      (m.)  =  ',ZZ    )
      CALL MTOF (ZZ,D1(I),4)
        D0(I)=D1(I)
        D2(I)=D1(I)
       GOTO 500
  23  RETURN
        END
C
C
C
C                  *-----------------------------------*
                     SUBROUTINE   EDBAR
C                  *-----------------------------------*
C  EDIT SIZE OF STEEL BAR
      COMMON /STSIZE/ NOSB,NOST,NOSL
 100  CALL CLRSCR
      CALL HEAD (' ********************************************** ')
      CALL HEAD (' *************   BAR  SIZE  EDITING   ************ ')
      CALL HEAD (' ********************************************** ')
      CALL PAGE (2)
      CALL LSTSTS
      CALL PAGE (2)
      CALL LISTI('   1>  BEAM Bar Size              #  ',NOSB  ,'      ')
      CALL PAGE (1)
      CALL LISTI('   2>  BEAM Stirrup  Bar Size     #  ',NOST  ,'      ')
      CALL PAGE (1)
```

```
      CALL LISTI('   3>  SLAB Bar Size              #  ',NOSL  ,'       ')
      CALL PAGE (1)
      CALL HEAD ('   4>  EVERYTHING IS ALLRIGHT                          ')
      CALL PAGE (2)
      CALL READI('                  SELECT NUMBER  (1,2,...,4) :  ',IWAY  )
      CALL PAGE (1)
      GOTO (1,2,3,4),IWAY
        GOTO 100
    1 CALL READI('                  BEAM  Bar  Size             #  ',NOSB  )
        GOTO 100
    2 CALL READI('                  BEAM  Stirrup  Bar Size     #  ',NOST  )
        GOTO  100
    3 CALL READI('                  SLAB   Bar  Size            #  ',NOSL  )
        GOTO  100
    4 RETURN
      END
C
C---------------------------------------
C         LIST BARS DIAMETER
C---------------------------------------
      SUBROUTINE  LSTSTS
      COMMON /MKSBAR/ KDIA(12)
      WRITE (*,2000) (I,I=1,12)
      WRITE (*,2010) (KDIA(I),I=1,12)
 2000 FORMAT ('          Bar #',12I5)
 2010 FORMAT ('    Diam.(mm.)',12I5)
      RETURN
      END
C
C
C               *----------------------------*
                    SUBROUTINE  EDSTM
C               *----------------------------*
C
      COMMON /BRGDIA/ NUMTRA,WLANE,EGDAP,NOBEAM,CSL,PAREA,NUMSPA
      COMMON /SETTLE/ STM(10)
  100 CALL  CLRSCR
      CALL HEAD (' ************************************************** ')
      CALL HEAD (' ***********    SETTLEMENT   EDITING    *********** ')
      CALL HEAD (' ************************************************** ')
      CALL PAGE (2)
      DO  80 I = 1,NUMSPA+1
      CALL FTOM (STM(I),ZZ,4)
      WRITE (*,1000) I,I,ZZ
 1000 FORMAT (16X,I3,'> Settlement at Joint  # ',I2,'  = ',F8.2,'  m.  ')
   80 CALL PAGE (1)
      WRITE (*,1100) I
 1100 FORMAT (16X,I3,'> EVERYTHING IS ALLRIGHT                          ')
      CALL PAGE (1)
      CALL READI('                  SELECT NUMBER  (1,2,3,...) :  ',IWAY  )
      IF (IWAY.GE.I) GOTO 11
      CALL PAGE (1)
      CALL READR('                  Settlement  at  this joint  =  ',ZZ    )
      CALL MTOF (ZZ,STM(IWAY),4)
      GOTO 100
   11 RETURN
      END
C
C
```

```
C                          *-----------------------*
C                          *      SUBPROGRAM       *
C                          *        SCREEN         *
C                          *-----------------------*
C
C
C                    *-------------------------------------*
C                                SUBROUTINE   CLRSCR
C                    *-------------------------------------*
C
C     CLEAR SCREEN
C
      CHARACTER*2 CLS
      COMMON /CLEAR/ CLS
      CLS = CHAR(155)//CHAR(191)
      WRITE (*,*) CLS
      RETURN
      END
C
C
C                    *-------------------------------------*
C                              SUBROUTINE    PAGE (N)
C                    *-------------------------------------*
C
C     SKIP ON SCREEN  (N) LINES
C
      DO  10  I = 1,N
   10 WRITE (*,*)
      RETURN
      END
C
C
C                    *-------------------------------------*
C                              SUBROUTINE   HEAD (STRING)
C                    *-------------------------------------*
C
C  PRINT MESSAGE ONE LINE
C
      CHARACTER*52 STRING
      WRITE (*,10) STRING
  10  FORMAT (14X,A52)
      RETURN
      END
C
C
C                    *-------------------------------------*
C                            SUBROUTINE   READC (LABEL,CVAR)
C                    *-------------------------------------*
C
C  Read Character in one line
C
      CHARACTER  LABEL*15
      CHARACTER  TVAR*52, CVAR*52
      CALL  TNOUA (LABEL, INTS(15))
      READ  (*,'(A)')  TVAR
      IF (TVAR.EQ.CHAR(32)) GOTO 55
      CVAR = CHAR(32)//TVAR
   55 RETURN
      END
```

```
C
C
C                    *-----------------------------------------*
C                          SUBROUTINE   READI (LABEL, IVAR)
C                    *-----------------------------------------*
C
C  Read Integer in one line
C
      CHARACTER*45 LABEL
 10   CALL  TNOUA (LABEL, INTS(45))
      READ  (*, *, ERR=10)  IVAR
      RETURN
      END
C
C
C                    *-----------------------------------------*
C                          SUBROUTINE   READR (LABEL, RVAR)
C                    *-----------------------------------------*
C
C  Read Real in one line
C
      CHARACTER*45 LABEL
 10   CALL  TNOUA (LABEL, INTS(45))
      READ  (*, *, ERR=10)  RVAR
      RETURN
      END
C
C
C                    *-----------------------------------------*
C                          SUBROUTINE   FTOM (FOOT, XMETRE, NO)
C                    *-----------------------------------------*
C
C  Convert English Unit  to  SI Unit
C       NO.   1)  in.  ->  cm.
C             2)  in.  ->  m.
C             3)  ft.  ->  cm.
C             4)  ft.  ->  m.
C             5)  lb.  ->  kg.
C             6)  psi  ->  ksc.
C             7)  psf  ->  ksm.
C             8)  sq. in.  ->  sq. cm.
C             9)  sq. in.  ->  sq. m.
C            10)  ft-kip  ->  kg-m.
C            11)  kips  ->  kg.
C    ------------------------------------
      DIMENSION C(11)
      DATA C /2.54, .0254, 30.48, .3048, .4536, .0703, 4.8825, 6.4516,
     +       .000645, 138.2573, 453.6/
      XMETRE = FOOT * C(NO)
      RETURN
      END
C
C
C                    *-----------------------------------------*
C                          SUBROUTINE   MTOF (XMETRE, FOOT, NO)
C                    *-----------------------------------------*
C
C CONVERSE METRE TO FOOT
C
      DIMENSION C(11)
      DATA C /2.54, .0254, 30.48, .3048, .4536, .0703, 4.8825, 6.4516,
```

```
     +        .000645,138.2573,453.6/
      FOOT = XMETRE / C(NO)
      RETURN
      END
C
C
C                    *-----------------------------------------*
                        SUBROUTINE   LISTR (FIRST,RVAR,LAST
C                    *-----------------------------------------*
C
C LIST REAL IN ONE LINE
C
      CHARACTER  FIRST*36,LAST*6
      WRITE (*,2000) FIRST,RVAR,LAST
 2000 FORMAT (14X,A36,F10.2,A6)
      RETURN
      END
C
C
C
C                    *-----------------------------------------*
                        SUBROUTINE   LISTI (FIRST,IVAR,LAST)
C                    *-----------------------------------------*
C
C  LIST INTEGER IN 1 LINE
C
      CHARACTER  FIRST*36,LAST*6
      WRITE (*,2000) FIRST,IVAR,LAST
 2000 FORMAT (14X,A36,I10,A6)
      RETURN
      END
C
C
C
C                    *-----------------------------------------*
                        SUBROUTINE   LISTJ (FIRST,JVAR)
C                    *-----------------------------------------*
C
C  LIST INTEGER IN 1 LINE
C
      CHARACTER  FIRST*42
      WRITE (*,2000) FIRST,JVAR
 2000 FORMAT (14X,A42,I2)
      RETURN
      END
C
C
C
C
C
C
C    ***************************************************************************
C    *                                                                         *
C    *                        S U B   P R O G R A M  2                         *
C    *                                                                         *
C    *                   COMPUTE CONSTANTS AND CALCULATION MENU                *
C    *                                                                         *
C    ***************************************************************************
C
```

```
C
C                    *-----------------------------------*
C                              SUBROUTINE      COMPUTE
C                    *-----------------------------------*
C
C    ANALYSIS AND DESIGN BRIDGE VARY CROSS SECTION
C
      INTEGER     CHANGE, TYP
      CHARACTER   HEADI*52
      COMMON /TITLE/ HEADI
      COMMON /MATER/ FCP, FSY
      COMMON /STSIZE/ NOSB, NOST, NOSL
      COMMON /BRGDIA/ NUMTRA, WLANE, EGDAP, NOBEAM, CSL, PAREA, NUMSPA
      COMMON /LCLASS/ LCLASS
      COMMON /SETTLE/ STM(10)
      COMMON /PBEAM/ BWIDTH, SINTER
      COMMON /BEAM/ TYP(9), SL(9), NSC(9), SL1(9), SL2(9), D0(9), D1(9), D2(9)
C
      COMMON /BDEPTH/ BDEPTH
      COMMON /CDATA/ FC, FS, MR, CK, CJ, CRR, VMAX, VCON1, VCON2, GDSPC
      COMMON /LOAD /DD1, DE1, DI1, DLL2, PLM, PLS
      REWIND 5
      READ (5, 1000) HEADI
      READ (5, 1100) FCP, FSY, WLANE, GDSPC
      READ (5, 1100) CSL, PAREA, EGDAP
      READ (5, 1200) NOSB, NOST, NOSL, LCLASS, NUMTRA,
     +               NUMSPA, NOBEAM
      REWIND 12
      READ (12, 1250) BWIDTH
      READ (12, 1250) SINTER
      DO 10 I=1, 9
   10 READ (12, 1300) TYP(I), SL(I), NSC(I), SL1(I), SL2(I), D0(I), D1(I), D2(I)
      READ (12, 1400) (STM(I), I=1, 10)
C
 1000 FORMAT (A)
 1100 FORMAT (8F10.3)
 1200 FORMAT (16I5)
 1250 FORMAT (F10.4)
 1300 FORMAT (I5, F10.4, I5, 5F10.4)
 1400 FORMAT (10F10.3)
C
C
      CALL  WRITED
C
      CALL  CONLOAD
C
      CALL  MENUCAL
C
C
      RETURN
        END
C
C
C
C
C                    *-----------------------------------*
C                              SUBROUTINE      WRITED
C                    *-----------------------------------*
C
C    WRITE TO FILE : OUT1.TXT
C
```

```
      INTEGER    CHANGE,TYP
      CHARACTER  HEADI*52
      COMMON /TITLE/ HEADI
      COMMON /MATER/ FCP,FSY
      COMMON /STSIZE/ NOSB,NOST,NOSL
      COMMON /BRGDIA/ NUMTRA,WLANE,EGDAP,NOBEAM,CSL,PAREA,NUMSPA
      COMMON /LCLASS/ LCLASS
      COMMON /SETTLE/ STM(10)
      COMMON /PBEAM/ BWIDTH,SINTER
      COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D1(9),D2(9)
C
      COMMON /BDEPTH/ BDEPTH
      COMMON /CDATA/ FC,FS,MR,CK,CJ,CRR,VMAX,VCON1,VCON2,BDEPC
C
       REWIND 6
C
      CALL WRITEP
      CALL WRTL (4)
      CALL WRTH (' ********************************************** ')
      CALL WRTH (' ************  TITLE HEADING LIST  ************** ')
      CALL WRTH (' ********************************************** ')
      CALL WRTL (2)
      CALL WRTH ('         An Existing Title is shown below          ')
      CALL WRTL (2)
      CALL WRTH (HEADI)
      CALL WRTL (2)
      CALL WRTH ('------------------------------------------------')
      CALL WRTL (4)
C
C
      CALL WRTL (4)
      CALL WRTH (' ********************************************** ')
      CALL WRTH (' ***********    MATERIAL PROPERTIES    ********** ')
      CALL WRTH (' ********************************************** ')
      CALL WRTL (2)
      CALL FTOM (FCP,ZZ,6)
      CALL WRTR (' 1> Ultimate Strength of concrete = ',ZZ,  ' ksc. ')
      CALL WRTL (1)
      CALL FTOM (FSY,ZZ,6)
      CALL WRTR (' 2> Yield Strength of Steel    = ',ZZ,  ' ksc. ')
      CALL WRTL (2)
      CALL WRTH ('------------------------------------------------')
      CALL WRTL (4)
C
C
      CALL WRTL (4)
      CALL WRTH (' ********************************************** ')
      CALL WRTH (' ************  LOAD CLASS LISTING  ************** ')
      CALL WRTH (' ********************************************** ')
      CALL WRTL (2)
      IF (LCLASS .EQ. 1) GOTO 10
      IF (LCLASS .EQ. 2) GOTO 20
        GOTO 30
   10 CALL WRTH ('      =====> by now Load Class is HS20          ')
      CALL WRTL (2)
        GOTO 30
   20 CALL WRTH ('      =====> by now Load Class is HS15          ')
      CALL WRTL (2)
        GOTO 30
   30 CALL WRTH ('------------------------------------------------')
```

```
      CALL WRITEP
      CALL WRTL (4)
      CALL WRTH (' ************************************************** ')
      CALL WRTH (' *********   BRIDGE  DIMENSION  LISTING   ********* ')
      CALL WRTH (' ************************************************** ')
      CALL WRTL (1)
      CALL WRTI (' 1> Number of Traffic Lane          ',NUMTRA,'       ')
      CALL WRTL (1)
      CALL FTOM (WLANE,ZZ,4)
      CALL WRTR (' 2> Lane    Width                   ',ZZ,    ' m.    ')
      CALL WRTL (1)
      CALL FTOM (EGDAP,ZZ,4)
      CALL WRTR (' 3> Outer Dimension Appart Ex-Gir. ',ZZ,    ' m.    ')
      CALL WRTL (1)
      CALL WRTI (' 4> Number of Girder per section   ',NOBEAM,'       ')
      CALL WRTL (1)
      CALL FTOM (CSL,ZZ,4)
      CALL WRTR (' 5> Overhanging Slab from Girder   ',ZZ,    ' m.    ')
      CALL WRTL (1)
      CALL FTOM (PAREA,ZZ,9)
      CALL WRTR (' 6> Area of Parapet Family         ',ZZ,    ' sq.m. ')
      CALL WRTL (1)
      CALL WRTI (' 7> Number of span                 ',NUMSPA,'       ')
      CALL WRTL (1)
      CALL FTOM (RR,ZZ,9)
      CALL WRTR (' 8> Curb  area                     ',ZZ,    ' sq.m. ')
      CALL WRTL (1)
      CALL FTOM (CB,ZZ,9)
      CALL WRTR (' 9> Cross beam  area               ',ZZ,    ' sq.m. ')
      CALL WRTL (2)
      CALL WRTH ('-------------------------------------------------------')
      CALL WRTL (4)
C
C
      CALL WRTL (4)
      CALL WRTH (' ************************************************** ')
      CALL WRTH (' *********    BEAM  DIMENSION  LISTING    ********* ')
      CALL WRTH (' ************************************************** ')
      CALL WRTL (2)
      CALL FTOM (BWIDTH,ZZ,4)
      CALL WRTR ('         1> Beam   Width          = ',ZZ,    ' m.    ')
      CALL WRTL (1)
      CALL FTOM (SINTER,ZZ,4)
      CALL WRTR ('         2> Length  of  Interval  = ',ZZ,    ' m.    ')
      CALL WRTL (2)
      CALL WRTH ('-------------------------------------------------------')
      CALL WRTL (4)
C
      CALL WRITEP
      CALL WRTL (5)
       DO 50 I=1,NUMSPA
       IF(TYP(I).EQ.3) THEN
      CALL WRTH (' **************************************************')
      WRITE (6,333) I
      CALL WRTH (' **************************************************')
      CALL WRTL(2)
      CALL WRTJ('   ======> by now  code  of  beam  shape =',TYP(I)   )
      CALL WRTL(2)
      CALL WRTH('      code :      1 = straight  haunch               ')
      CALL WRTL(1)
      CALL WRTH('                  2 = parabolar haunch               ')
      CALL WRTL(1)
      CALL WRTH('                  3 = constant cross section         ')
```

```
      CALL WRTL (4)
C
C
      CALL WRITEP
      CALL WRTL (4)
      CALL WRTH (' ********************************************** ')
      CALL WRTH (' *********   BRIDGE  DIMENSION  LISTING   ********* ')
      CALL WRTH (' ********************************************** ')
      CALL WRTL (1)
      CALL WRTI (' 1> Number of Traffic Lane        ',numtra,'       ')
      CALL WRTL (1)
      CALL FTOM (WLANE,ZZ,4)
      CALL WRTR (' 2> Lane    Width                 ',zz,    ' m.    ')
      CALL WRTL (1)
      CALL FTOM (EGDAP,ZZ,4)
      CALL WRTR (' 3> Outer Dimension Appart Ex-Gir.',zz,    ' m.    ')
      CALL WRTL (1)
      CALL WRTI (' 4> Number of Girder per section  ',nobeam,'       ')
      CALL WRTL (1)
      CALL FTOM (CSL,ZZ,4)
      CALL WRTR (' 5> Overhanging Slab from Girder  ',zz,    ' m.    ')
      CALL WRTL (1)
      CALL FTOM (PAREA,zz,9)
      CALL WRTR (' 6> Area of Parapet Family        ',zz,    ' sq.m. ')
      CALL WRTL (1)
      CALL WRTI (' 7> Number of span                ',NUMSPA,'       ')
      CALL WRTL (2)
      CALL WRTH ('-----------------------------------------------------')
      CALL WRTL (4)
C
C
      CALL WRTL (4)
      CALL WRTH (' ********************************************** ')
      CALL WRTH (' *********    BEAM  DIMENSION  LISTING    ********* ')
      CALL WRTH (' ********************************************** ')
      CALL WRTL (2)
      CALL FTOM (BWIDTH,ZZ,4)
      CALL WRTR ('        1> Beam   Width          = ',ZZ,    ' m.    ')
      CALL WRTL (1)
      CALL FTOM (SINTER,ZZ,4)
      CALL WRTR ('        2> Length  of  Interval  = ',ZZ,    ' m.    ')
      CALL WRTL (2)
      CALL WRTH ('-----------------------------------------------------')
      CALL WRTL (4)
C
      CALL WRITEP
      CALL WRTL (5)
       DO 50 I=1,NUMSPA
       IF(TYP(I).EQ.3) THEN
      CALL WRTH (' **********************************************')
      WRITE (6,333) I
      CALL WRTH (' **********************************************')
      CALL WRTL(2)
      CALL WRTJ('   ======> by now  code  of  beam  shape =',TYP(I)   )
      CALL WRTL(2)
      CALL WRTH('     code :       1 = straight  haunch             ')
      CALL WRTL(1)
      CALL WRTH('                  2 = parabolar haunch             ')
      CALL WRTL(1)
      CALL WRTH('                  3 = constant cross section       ')
```

```
C
      CALL WRTL(1)
      CALL FTOM(SL(I),ZZ,4)
      CALL WRTR('     1>    span  length                    =',ZZ,  'm.')
      CALL WRTL(1)
       IZZ = NSC(I)
      CALL WRTI('      number of section  consider          =',IZZ,'sect')
      CALL WRTL(1)
      CALL FTOM(D1(I),ZZ,4)
      CALL WRTR('     2>    depth  of  beam                 =',ZZ,  'm.')
      CALL WRTL(2)
      CALL WRTH('-------------------------------------------------------')
      CALL WRTL(4)
       ELSE
C
C
      CALL WRTH (' ************************************************** ')
      WRITE (6,333) I
  333 FORMAT(14X,' **************   SPAN  NUMBER  ',I2,'   ***************
     +**')
      CALL WRTH (' ************************************************** ')
      CALL WRTL (2)
      CALL WRTJ ('    ====>   by now  Code of Beam Shape  = ',TYP(I)   )
      CALL WRTL (2)
      CALL WRTH ('    Code :          1 =  Straight haunch             ')
      CALL WRTL (1)
      CALL WRTH ('                    2 =  Paraboler haunch            ')
      CALL WRTL (1)
      CALL WRTH ('                    3 =  Constant Cross Section      ')
      CALL WRTL (1)
C
      CALL WRTL (1)
      CALL FTOM (SL(I),ZZ,4)
      CALL WRTR ('    1> Span  Length                 = ',ZZ,   ' m.   ')
      CALL WRTL (1)
      IZZ =  NSC(I)
      CALL WRTI ('       Number of Section Consider = ',IZZ,  ' sect.')
      CALL WRTL (1)
      CALL FTOM (D1(I),ZZ,4)
      CALL WRTR ('    2> Left side  Depth             = ',ZZ,   ' m.   ')
      CALL WRTL (1)
      CALL FTOM (DO(I),ZZ,4)
      CALL WRTR ('    3> Centre     Depth             = ',ZZ,   ' m.   ')
      CALL WRTL (1)
      CALL FTOM (D2(I),ZZ,4)
      CALL WRTR ('    4> Right side Depth             = ',ZZ,   ' m.   ')
      CALL WRTL (1)
      CALL FTOM (SL1(I),ZZ,4)
      CALL WRTR ('    5> Left  side Length            = ',ZZ,   ' m.   ')
      CALL WRTL (1)
      CALL FTOM (SL2(I),ZZ,4)
      CALL WRTR ('    6> Right side Length            = ',ZZ,   ' m.   ')
      CALL WRTL (2)
      CALL WRTH ('-------------------------------------------------------')
      CALL WRTL (4)
      ENDIF
   50 CONTINUE
C
```

```
      CALL WRITEP
      CALL WRTH (' ************************************************** ')
      CALL WRTH (' **************   BAR  SIZE  LISTING   ************** ')
      CALL WRTH (' ************************************************** ')
      CALL WRTL (2)
      CALL WRTLST
      CALL WRTL (2)
      CALL WRTI ('   1>  BEAM Bar Size              #  ',NOSB  ,'       ')
      CALL WRTL (1)
      CALL WRTI ('   2>  BEAM Stirrup  Bar Size     #  ',NOST  ,'       ')
      CALL WRTL (1)
      CALL WRTI ('   3>  SLAB Bar Size              #  ',NOSL  ,'       ')
      CALL WRTL (2)
      CALL WRTH ('-------------------------------------------------------')
      CALL WRTL (4)
C
C
      CALL WRTL (4)
      CALL WRTH (' ************************************************** ')
      CALL WRTH (' ***********    SETTLEMENT   LISTING    *********** ')
      CALL WRTH (' ************************************************** ')
      CALL WRTL (2)
      DO  80 I = 1,NUMSPA+1
      CALL RTOM (STM(I),ZZ,4)
      WRITE (6,1000) I,I,ZZ
 1000 FORMAT (16X,I3,'> Settlement at Joint  # ',I2,'  = ',F8.2,'  m. ')
   80 CALL WRTL (1)
      CALL WRTL (1)
      CALL WRTH ('-------------------------------------------------------')
      CALL WRTL (5)
C
C
      RETURN
         END
C
C
C
C
C                  *-------------------------------------*
                         SUBROUTINE      COMLOAD
C                  *-------------------------------------*
C
      INTEGER     CHANGE,TYP
      CHARACTER   HEADI*52
      COMMON /TITLE/ HEADI
      COMMON /MATER/ FCP,FSY
      COMMON /STSIZE/ NOSB,NOST,NOSL
      COMMON /BRGDIA/ NUMTRA,WLANE,EGDAP,NOBEAM,CSL,PAREA,NUMSPA
      COMMON /LCLASS/ LCLASS
      COMMON /SETTLE/ STM(10)
      COMMON /PBEAM/ BWIDTH,SINTER
      COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),DO(9),D1(9),D2(9)
C
      COMMON /BDEPTH/ BDEPTH
      COMMON /CDATA/ FC,FS,MR,CK,CJ,CRR,VMAX,VCON1,VCON2,GDSPC
      COMMON /LOAD/DD1,DE1,DI1,DLL2,PLM,PLS
      TH = 7.0
      GDSPC = (EGDAP-BWIDTH)/(NOBEAM-1)
      FC = 0.4*FCP
      FS = 0.5*FSY
      MR = 29000000/((145**1.5)*33*(FCP**0.5))
      E  = (145**1.5)*33*(FCP**0.5)
      CK = 1.0/(1.0+FS/(MR*FC))
      CJ = 1.0 - CK/3
C
       IF (NUMTRA.EQ.2) FAC = 1.00
       IF (NUMTRA.EQ.3) FAC = 0.90
       IF (NUMTRA.GE.4) FAC = 0.75
C
       SLM   = 0
      BDEPTH = 0
      DO 20 I=1,NUMSPA
      IF (SL(I).GT.SLM) SLM = SL(I)
      IF (DO(I).GT.BDEPTH) BDEPTH=DO(I)
```

```
      IF (D1(I).GT.BDEPTH) BDEPTH=D1(I)
      IF (D2(I).GT.BDEPTH) BDEPTH=D2(I)
   20 CONTINUE
C
      NCB = SLM/60 + 1.00
      CBI = SLM/NCB
      LCRSE=(CB/144)*GDSPC/2*145/CBI*SINTER
      LCRSI=(CB/144)*GDSPC*145/CBI*SINTER
      LCURB=(RR/144)*145*SINTER
      IF (LCLASS.EQ.1) THEN
        ULL = 640*SINTER
        PLM = 18000
        PLS = 26000
      ELSE
        ULL = 480*SINTER
        PLM = 13500
        PLS = 19500
      ENDIF
C
      UDLB = BDEPTH*BWIDTH*SINTER*145
      UDLS = GDSPC*(TH/12)*SINTER*(145/2)
      UDPL =(PAREA/144)*SINTER*145
      FACM = GDSPC/6
      UDLE =(UDLB+UDLS+(UDLS*2*CSL)/GDSPC+UDPL)/1000
      ADDLE=(LCRSE+LCURB)/1000
      UDLE = UDLE+ADDLE
C
      ULL  =(0.5*ULL*FACM)/1000
      PLM  =(0.5*PLM*FACM)/1000
      PLS  =(0.5*PLS*FACM)/1000
      UDLI =(UDLB+2*UDLS)/1000
      ADDLI=(LCRSI)/1000
      UDLI = UDLI+ADDLI
C
      DI1 = UDLI
      DE1 = UDLE
      ULL= ULL*FAC
      DLL2= ULL
      PLM= PLM*FAC
      PLM = PLM
      PLS= PLS*FAC
      PLS = PLS
C
      RETURN
        END
C
C
C
C
C                  *-------------------------------------*
C                        SUBROUTINE   MENUCAL
C                  *-------------------------------------*
C
  100 CALL CLRSCR
      CALL PAGE (2).
      CALL HEAD (' ******************************************************')
      CALL HEAD ('     ************ CALCULATION MENU ************         ')
      CALL HEAD (' ******************************************************')
      CALL PAGE (2)
      CALL HEAD ('
      CALL PAGE (1)                  1> SLAB DESIGN.                      ')
      CALL HEAD ('
      CALL PAGE (1)                  2> BEAM ANALYSIS.                    ')
```

```
      CALL HEAD ('                3> BEAM DESIGN.                          ')
      CALL PAGE (1)
      CALL HEAD ('                4> <<< RETURN TO MAIN MENU >>>           ')
      CALL PAGE (2)
      CALL READI ('                 SELECT NUMBER (1,2,3 or 4) :   ',IWAY  )
      CALL PAGE (2)
      GOTO (1,2,3,4),IWAY
        GOTO 100
C
C   SLAB DESIGN PROCESS
  1   CALL SLAB
        GOTO 100
C
C   BEAM ANALYSIS FOR VARY CROSS-SECTION
  2   CALL ANALYSIS
        GOTO 100
C
C   BEAM DESIGN PROCESS
  3   CALL BDESIGN
        GOTO 100
C
C   RETURN TO MAIN MENU
  4   RETURN
        END .
C
C
C
C   ***************************************************************************
C   *                                                                         *
C   *                      S U B   P R O G R A M  2 A                         *
C   *                                                                         *
C   *                        S L A B    D E S I G N                           *
C   *                                                                         *
C   ***************************************************************************
C
C
C                          *-------------------------*
C                                SUBROUTINE  SLAB
C                          *-------------------------*
C
      CHARACTER  ANYKEY*15
      CHARACTER  HEADI*52
      INTEGER    CHANGE,TYP
C
      COMMON /STESEC/ CR,ST(12,2),FORM(12,2)
C
      COMMON /TITLE/ HEADI
      COMMON /MATER/ FCP,FSY
      COMMON /STSIZE/ NOSB,NOST,NOSL
      COMMON /BRGDIA/ NUMTRA,WLANE,EGDAP,NOBEAM,CSL,PAREA,NUMSPA
      COMMON /LCLASS/ LCLASS
      COMMON /SETTLE/ STM(10)
      COMMON /PBEAM/ BWIDTH,SINTER
      COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D1(9),D2(9)
C
      COMMON /CDATA/ FC,FS,MR,CK,CJ,CRR,VMAX,VCON1,VCON2,GDEPC
C
C
      CALL WRITEP
C
```

```
C
 55   CALL CLRSCR
      CALL PAGE (1)
C
C
      CALL WRITEP
      WRITE (6,*)' ************************************************************ '
      WRITE (6,*)' **************      SLAB    DESIGN     ***************** '
      WRITE (6,*)' ************************************************************ '
C
C
      WRITE (6,*)
      WRITE (6,*)
C  DETECT LIVE LOAD SPEC.
      IF (LCLASS .EQ. 1) PLOAD = 16.0
      IF (LCLASS .EQ. 2) PLOAD = 12.0
      WRITE(6,*) '    PLOAD =', PLOAD
C  ASSUME GIRDER WIDTH
      GW = BWIDTH * 12.0
C  CALCULATE SLAB EFFECTIVE SPAN LENGTH
      SLSP = GDSPC*12.0 - GW*0.5
      WRITE(6,*) '    SPAN eff.  =', SLSP
C  CALCULATE LIVE LOAD BENDING MOMENT
      BMLL = 0.8*(SLSP/12.  + 2.0)*PLOAD/32.
      BMLL = BMLL*1000.*12.
C  CALCULATE IMPACT BENDING MOMENT
      BMIM = 0.3*BMLL
      WRITE(6,*) '    IMPACT MOMENT =', BMIM
C  CALCULATE DEAD WEIGHT OF SLAB
 100  DWSL = (TH/12.)*150.  + (1.5/12.)*125.
      WRITE(6,*) '    DEAD WEIGHT =', DWSL
C  CALCULATE DEAD WEIGHT BENDING MOMENT
      DWBM = DWSL*SLSP*SLSP/(12.*10.)
      WRITE(6,*) '    D.L.  MOMENT =', DWBM
C  SUMMATION ALL BENDING MOMENT
      TOTM = DWBM + BMLL + BMIM
      WRITE(6,*) '    TOTAL MOMENT =', TOTM
C  CALCULATE d FROM ALLOWABLE STRESS DESIGN
      D = SQRT(2.*TOTM/(FC*12.*CJ*CK))
      WRITE(6,*) '    d = ', D
C  FIND REAL TOTAL DEPTH OF SLAB
      RTH = D + ST(NOSL,1)*0.5 +2.
      RTH = ANINT(2.*RTH + 0.5)
      RTH = RTH/2.
      WRITE(6,*) '    REAL TOTAL DEPTH =', RTH
C  CHECK FOR RE-CALCULATION USING TRIAL & ERROR
      IF (RTH .LE. TH) GOTO 200
       TH = RTH
       GOTO 100
C  CALCULATE ACTUAL d OF SLAB
 200  D = RTH - ST(NOSL,1)*0.5 - 2.
      WRITE(6,*) '    Actual  d =', D
C
C
      CALL PAGE (2)
      CALL READC('Press any key : ',ANYKEY)
C
C
C  OUTPUT
C
```

```
 555  CALL CLRSCR
      CALL PAGE (2)
C
      WRITE (6,*)
      WRITE (6,*)
C  CALCULATE  As
      AS = TOTM/(FS*CJ*D)
      CALL PAGE (2)
      CALL PAGE (1)
      CALL WRTL (2)
      WRITE (6,*)'     ========== TRANSVERSE  STEEL ==========='
      CALL WRTL (1)
      CALL SPACIN (AS)
C  CALCALATE LONGITUDINAL STEEL
      AS = 0.67 * AS
      NOSL = NOSL - 1
      CALL PAGE (2)
      CALL PAGE (1)
      CALL WRTL (2)
      WRITE (6,*)'     ======== LONGITUDINAL STEEL ==========='
      CALL WRTL (1)
      CALL SPACIN (AS)
C  CALCULATE TEMPERATURE STEEL
      AS = 1./8.
      NOSL = 4
      CALL PAGE (2)
      CALL PAGE (1)
      CALL WRTL (2)
      WRITE (6,*)'     ========== TEMPERATURE STEEL ==========='
      CALL WRTL (1)
      CALL SPACIN (AS)
C
C
      CALL PAGE (2)
      CALL READC('Press any key :  ',ANYKEY)
      RETURN
      END
C
C
C
C************************************
C   CALCULATE SPACING OF BAR        *
C************************************
C
C                    *-------------------------*
                     SUBROUTINE SPACIN (AS)
C                    *-------------------------*
C
      COMMON  /STSIZE/ NOSB,NOST,NOSL
      COMMON  /STESEC/ CR,ST(12,2),FORM(12,2)
C
C  CALCULATE SPACING
       SP1 = 12./(AS/ST(NOSL,2))
       SP1 = ANINT(SP1 - 0.5)
       IF ( SP1 .GT.  18. ) SP1 = 18.
                      CALL FTOM (SP1 ,ZZ,2)
       WRITE(6,2000) NOSL,ZZ
 2000  FORMAT (20X,'USE #',I2,'     Spacing =',F7.2,' m.')
       RETURN
       END
```

```
C
C
C
C
C
C                    *-----------------------------------------*
                           SUBROUTINE   WRTH (STRING)
C                    *-----------------------------------------*
C
C .PRINT MESSAGE ONE LINE
C
      CHARACTER*52 STRING
      WRITE (6,10) STRING
 10   FORMAT (14X,A52)
      RETURN
      END
C
C
C
C                    *-----------------------------------------*
                        SUBROUTINE   WRTR (FIRST,RVAR,LAST)
C                    *-----------------------------------------*
C
C LIST REAL IN ONE LINE
C
      CHARACTER  FIRST*36,LAST*6
      WRITE (6,2000) FIRST,RVAR,LAST
 2000 FORMAT (14X,A36,F10.2,A6)
      RETURN
      END
C
C
C
C                    *-----------------------------------------*
                        SUBROUTINE   WRTI (FIRST,IVAR,LAST)
C                    *-----------------------------------------*
C
C  LIST INTEGER IN 1 LINE
C
      CHARACTER  FIRST*36,LAST*6
      WRITE (6,2000) FIRST,IVAR,LAST
 2000 FORMAT (14X,A36,I10,A6)
      RETURN
      END
C
C
C
C                    *-----------------------------------------*
                        SUBROUTINE  WRTJ (FIRST,JVAR)
C                    *-----------------------------------------*
C
C  LIST INTEGER IN 1 LINE
C
      CHARACTER  FIRST*42
      WRITE (6,2000) FIRST,JVAR
 2000 FORMAT (14X,A42,I2)
      RETURN
      END
C
```

```
C                    *-----------------------------------------*
C                              SUBROUTINE   WRTL (N)
C                    *-----------------------------------------*
C
C     SKIP ON FILE#6 (N) LINES
C
      DO  10  I = 1,N
   10 WRITE (6,*)
      RETURN
      END
C
C
C
C                    *-----------------------------------------*
C                              SUBROUTINE   WRITEP
C                    *-----------------------------------------*
C
C
      WRITE (6,10)
   10 FORMAT (1H1)
      RETURN
      END
C
C
C---------------------------------------
C         LIST BARS DIAMETER TO FILE#6
C---------------------------------------
      SUBROUTINE  WRTLST
      COMMON /MKSBAR/ KDIA(12)
      WRITE (6,2000) (I, I=1,12)
      WRITE (6,2010) (KDIA(I), I=1,12)
 2000 FORMAT ('          Bar #',12I5)
 2010 FORMAT ('     Diam. (mm.)',12I5)
      RETURN
      END
C
C
C
C
C
C   ************************************************************************
C   *                                                                      *
C   *                    S U B   P R O G R A M  # 3                        *
C   *                                                                      *
C   *                      B E A M   D E S I G N                           *
C   *                                                                      *
C   ************************************************************************
C
C
C   ************************************************************************
C   *                        BEAM   ANALYSIS                               *
C   ************************************************************************
C
C     PRORAM BRIDGE GIRDER ANALYSE
      SUBROUTINE  ANALYSIS
C
      INTEGER  TYP
      CHARACTER  HEADI*52
      COMMON /TITLE/ HEADI
      COMMON /MATER/ FCP,FSY
```

```
      COMMON /STSIZE/ NOSB, NOST, NOSL
      COMMON /BRGDIA/ NUMTRA, WLANE, EGDAP, NOBEAM, CSL, PAREA, NUMSPA
      COMMON /LCLASS/ LCLASS
      COMMON /SETTLE/ STM(10)
      COMMON /PBEAM/ BWIDTH, SINTER
      COMMON /BEAM/ TYP(9), SL(9), NSC(9), SL1(9), SL2(9), D0(9), D1(9), D2(9)
C
      COMMON /BDEPTH/ BDEPTH
      COMMON /CDATA/ FC, FS, MR, CK, CJ, CRR, VMAX, VCON1, VCON2, GDSPC
C
      COMMON /ZZ/ SK1(9), SK2(9), SK3(9), G1(9), G2(9), LL(9)
     * , A(9), B(9), C(9), D(9), Z(9, 250), V(9, 250), F(9, 250), DN(100)
      COMMON /YY/ AA(100, 100), BB(100, 100), A1(100, 20), B1(100, 20), DF(100)
     * , A11(100, 20), B11(100, 20)
      COMMON /YY1/AA1(100, 100), BB1(100, 100)
      COMMON /YY3/Z1(100), A2(100, 12), B2(100, 12), A21(100, 12), B21(100, 12)
      COMMON /OVERFLOW/ CCC(100, 100)
      COMMON /YY4/ AAA(100, 12), BBB(100, 12)
      COMMON /YY5/ AAAA(100, 1), BBBB(100, 1)
      COMMON /YY6/ AAAAA(100, 1), BBBBB(100, 1)
      COMMON /DIST/ X(9, 50), DX(9, 50)
      COMMON /LOAD/DD1, DE1, DI1, DLL2, PLM, PLS
      L = NUMSPA
      DO 5 I=1, 9
      LL(I)=NSC(I)-1
    5 CONTINUE
C
        REWIND 11
        REWIND 14
        REWIND 18
C
C     FOR INITIALIZATION OF ALL DATA TO ZERO
C
      CALL INITZ(A, B, C, D, G1, G2, F, V, Z, L, LL)
      CALL DISTAN(L)
      CALL PRESTM(L)
C
C
      LR = 7
      LC = 8
      LD = 11
      WRITE(LD, 6000)
      WRITE(LD, *)'*************************************************************'
      WRITE(LD, *)' ', HEADI , ' '
      WRITE(LD, *)'*************************************************************'
      WRITE(LD, 901) L
      DO 10 I=1, L
      WRITE(LD, 902) I, LL(I)
      WRITE(LD, 903) SL(I)
   10 CONTINUE
  901 FORMAT(/, ' NUMBER  OF  SPANS  = ', I4)
  902 FORMAT(/, ' SPAN NO.  ', I4, '   INTERVAL = ', I4)
  903 FORMAT(  '                      LENGTH   = ', F9.4)
C
      CALL HADAT(L, LR, LD)
      CALL DISFA(L)
       L = NUMSPA
       LR = 7
       LC = 8
       LD = 11
```

```
      CALL LOAD1 (L,LR,LD)
      K =0
      DO 21 I=1,L
      K = K + LL(I)
   21 CONTINUE
      K = K -1
       I1 = 0
      DO 31 I = 1,L
      DO 41 J = 1,LL(I)
       I1 = I1 +1
   41 CONTINUE
   31 CONTINUE
      CALL ORDIN (AA,A1,K,1,L,LL,Z1)
      CALL ORDIN (BB,B1,K+1,2,L,LL,Z1)
      CALL TRANS
      CALL LOADS(L,LR,LD)
C
      K=0
      DO  20  I=1,L
      K = K + LL(I)
   20 CONTINUE
      K = K - 1
      I1 = 0
      DO  30  I = 1,L
      DO  40  J = 1,LL(I)
      I1 = I1+1
C
C     IF ( I.EQ.L .AND. J.EQ.LL(I) ) GOTO 40
C     READ(LR,99) Z1(I1)
C 99  FORMAT (F10.2)
C
C
   40 CONTINUE
   30 CONTINUE
C
C
      CALL  ORDIN(AA,A1,K,1,L,LL,Z1)
      CALL  ORDIN(BB,B1,K+1,2,L,LL,Z1)
C
C
      DO 50 JP=1,2
      WRITE(LD,6000)
      WRITE(LD,*)'****************'
      IF (JP.EQ.1) WRITE(LD,*)'EXTERIOR-GIRDER'
      IF (JP.EQ.2) WRITE(LD,*)'INTERIOR-GIRDER'
      WRITE(LD,*)'****************'
      IF (JP.EQ.1) DD1 = DE1
      IF (JP.EQ.2) DD1 = DI1
      CALL  IMPAC(DF,DN,1,SL,L,LL)
      CALL  EQUIV(AAA,A1,A11,K,DF,DN,L,LL,LR,LD)
      CALL  FINSTM(L)
      WRITE(LD,1000)
      WRITE(LD,2000)
      WRITE(LD,5000)(I,(AAA(I,J)*0.1383,J=1,12),I=1,K)
      WRITE(LD,6000)
C
      CALL IMPAC(DF,DN,2,SL,L,LL)
      CALL EQUIVS(BBB,B1,B11,K+1,DF,DN,L,LL,LR,LD)
      CALL  FINSTM(L)
      WRITE(LD,1500)
```

```
      WRITE(LD,2000)
      WRITE(LD,5000) (I,(BBB(I,J)*0.4536,J=1,12),I=1,K+1)
      WRITE(LD,6000)
      IF(JP.EQ.1) THEN
       DO 111 I =1,K
       AAAA(I,1)=AAA(I,1)
  111  CONTINUE
       DO 333 I =1,K+1
       BBBB(I,1)=BBB(I,1)
  333  CONTINUE
          ELSE
         DO   555 I =1,K
      AAAAA(I,1)     = AAA(I,1)
  555 CONTINUE
         DO  777 I =1,K+1
      BBBBB(I,1)    = BBB(I,1)
  777 CONTINUE
        ENDIF
   50 CONTINUE
C
C
      CALL TRUCK (SL,LL,K)
      CALL IMPAC (DF,DN,1,SL,L,LL)
      CALL TRUCKT (DF,DN,K,1)
      CALL FINSTM (L)
      CALL TRUCKS (SL,LL,K+1)
      CALL IMPAC (DF,DN,2,SL,L,LL)
      CALL TRUCTS (DF,DN,K+1)
      CALL FINSTM (L)
C
C
       DO 60 JP=1,2
        IF (JP.EQ.1) THEN
       WRITE(LD,6000)
       WRITE(LD,*) '*****************************'
       WRITE(LD,*) '  TRUCK  EXTERIOR GIRDER  '
       WRITE(LD,*) '*****************************'
C
       WRITE(LD,3000)
       WRITE(LD,4000)
       WRITE(LD,5000) (I,(AA(I,J)*0.1383,J=1,12),I=1,K)
       WRITE(LD,6000)
       WRITE(LD,3500)
       WRITE(LD,4000)
       WRITE(LD,5000) (I,(BB(I,J)*0.4536,J=1,12),I=1,K+1)
       WRITE(LD,6000)
            ELSE
       WRITE(LD,6000)
       WRITE(LD,*) '*****************************'
       WRITE(LD,*) '  TRUCK  INTERIOR  GIRDER   '
       WRITE(LD,*) '*****************************'
C
      WRITE(LD,3000)
      WRITE(LD,4000)
      WRITE(LD,5000) (I,(AA1(I,J)*0.1383,J=1,12),I=1,K)
      WRITE(LD,6000)
      WRITE(LD,3500)
      WRITE(LD,4000)
      WRITE(LD,5000) (I,(BB1(I,J)*0.4536,J=1,12),I=1,K+1)
      WRITE(LD,6000)
```

```
          ENDIF
60      CONTINUE
        CALL OUTPUT(AAA,BBB,Z,K-1,L,LL,LR,12)
C
C
C
 1000 FORMAT (//38X,'GIRDER BENDING MOMENT-EQUIVALENT LOADING (TON-M) ')
 1500 FORMAT (//38X,'      GIRDER SHEAR-EQUIVALENT LOADING (TON)   ')
 2000 FORMAT (/'      DEAD-LOAD       LIVE-LOAD            CONCEN.-LOAD
     *     TOTAL LL.            IMPACT LOAD        SETTLE            TOTAL')
C
 3000 FORMAT (//38X,'GIRDER BENDING MOMENT-TRUCK LOADING (TON-M) ')
 3500 FORMAT (//38X,'      GIRDER SHEAR-TRUCK LOADING (TON)   ')
 4000 FORMAT (/'      DEAD-LOAD      HI-AXIAL LOAD       LOW-AXIAL LOAD
     *     TOTAL LL.            IMPACT LOAD        SETTLE            TOTAL')
C
 5000 FORMAT (I3,12F10.4)
 6000 FORMAT (//)
C
        WRITE(LD,*) ' *****   END OF BEAM ANALYSIS   ***** '
        WRITE(LD,6000)
        RETURN
        END
C
C
C
C
C   FOR INITIALIZATION MATRIX
C
        SUBROUTINE INITZ(A,B,C,D,G1,G2,Z,F,V,L,LL)
C
        DIMENSION A(1),B(1),C(1),D(1),G1(1),G2(1),
     * Z(9,1),F(9,1),V(9,1),LL(1),AA(100,1),BB(100,1),A1(100,1),
     * B1(100,1)
C
        L1 = L+1
        L2 = 150
        DO 100 I=1,L1
          A(I) = 0.0
          B(I) = 0.0
          C(I) = 0.0
          D(I) = 0.0
          G1(I) = 0.0
          G2(I) = 0.0
          DO 100 J=1,L2
            F(I,J) =0.0
            V(I,J) =0.0
 100        Z(I,J) =0.0
        RETURN
        END
C
C
C
C
C   INPUT BEAM GEOMETRY FOR HAUNCH
C
        SUBROUTINE HADAT(L,LR,LD)
C
        INTEGER  TYP
        COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D1(9),D2(9)
```

```
      COMMON /ZZ/ SK1(9),SK2(9),SK3(9),Q1(9),Q2(9),LL(9)
     * ,A(9),B(9),C(9),D(9),Z(9,250),V(9,250),F(9,250),DN(100)
C
      DO 200 I=1,L
      CALL HAUNCH(SK1(I),SK2(I),SK3(I),I,FMA,FMB,TYP(I),0.5,SL(I)
     * ,SL1(I),SL2(I),DO(I),D1(I),D2(I))
      CALL SUMST(I,L)
  200 CONTINUE
      RETURN
      END
C
C
C_____________________________________________________________________
C
C   FOR SUM STIFFNESS THIS JOINT
C
      SUBROUTINE SUMST(I,L)
C
      INTEGER  TYP
      COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),DO(9),D1(9),D2(9)
      COMMON /ZZ/ SK1(9),SK2(9),SK3(9),Q1(9),Q2(9),LL(9)
     * ,A(9),B(9),C(9),D(9),Z(9,250),V(9,250),F(9,250),DN(100)
C
      I1 = I+1
      D(I) = D(I) + SK1(I)
      D(I1) = D(I1)+SK2(I)
      C(I) = SK3(I)
      A(I1) = SK3(I)
      RETURN
      END
C
C
C_____________________________________________________________________
C
C   FOR DISTRIBUTED FACTOR
C
      SUBROUTINE DISFA(L)
C
      INTEGER  TYP
      COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),DO(9),D1(9),D2(9)
      COMMON /ZZ/ SK1(9),SK2(9),SK3(9),Q1(9),Q2(9),LL(9)
     * ,A(9),B(9),C(9),D(9),Z(9,250),V(9,250),F(9,250),DN(100)
C
      L1 = L+1
      DO 100 I=2,L1
      I1 = I-1
      A(I) = A(I)/D(I1)
      D(I) = D(I)-A(I)*C(I1)
  100 CONTINUE
      RETURN
      END
C
C
C_____________________________________________________________________
C
C   FOR INPUT LOAD
C
      SUBROUTINE LOADS(L,LR,LD)
C
      INTEGER  TYP
```

```
      COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D1(9),D2(9)
      COMMON /ZZ/ SK1(9),SK2(9),SK3(9),Q1(9),Q2(9),LL(9)
     * ,A(9),B(9),C(9),D(9),Z(9,250),V(9,250),F(9,250),DMMY(100)
      COMMON /YY/ AA(100,100),BB(100,100),DMMY1(100,41)
C
      L1 = L+1
      L2 = 150
      N = 0
      DO 100 I=1,L
  100 CALL SETPT(SL,Z,I,L,LL)
      DO 600 I=1,L
        L3 = LL(I) + 1
        IF (I.EQ.L) L3 = LL(I)
        DO 600 KK=2,L3
          N = N+1
          X = Z(I,KK)
          P = 1.0
          CALL FIXED(SL,X,P,Z,F,V,A1,A2,I,L,LL,TYP,SL1,SL2,D0,D1,D2)
          Q1(I) = Q1(I) + A1
          Q2(I) = Q2(I) + A2
          CALL BALAN(B,A1,A2,I,L,LL)
          CALL CARRY(A,B,C,D,L,LL)
          CALL MEMFC(SL,Z,F,V,B,SK1,SK2,SK3,Q1,Q2,L,LL,LR,LD)
          CALL CHANG(AA,BB,F,V,N,L,LL)
          DO 600 K=1,L1
            B(K) = 0.0
            Q1(K) = 0.0
            Q2(K) = 0.0
            DO 600 J=1,L2
              F(K,J) = 0.0
              V(K,J) = 0.0
  600   CONTINUE
      DO 200 I=1,N
        BB(N+1,I) = BB(N,I)
        IF (I.EQ.N) BB(N+1,I) = BB(N,I) - 1.
  200 CONTINUE
      RETURN
      END
C
C-----------------------------------------------------------------------
C     FOR INPUT VARY LOAD
C
      SUBROUTINE  LOAD1 (L,LR,LD)
C
      INTEGER   TYP
      COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D1(9)
     * ,D2(9)
      COMMON/ZZ/ SK1(9),SK2(9),SK3(9),Q1(9),Q2(9),LL(9),A(9),B(9)
     * ,C(9),D(9),Z(9,250),V(9,250),F(9,250),DMMY(100)
      COMMON/YY/ AA(100,100),BB(100,100),DMMY1(100,41)
      COMMON/BDEPTH/ BDEPTH
      COMMON/DIST/ X(9,50),DX(9,50)
C
      L1 = L+ 1
      L2 =150
      N = 0
      DO 100 I = 1,L
  100 CALL SETPT (SL,Z,I,L,LL)
      DO 600 I =1,L
        L3 = LL(I) + 1
```

```
      IF(I.EQ.L) L3 = LL(I)
      DO 600 KK =2,L3
      N = N+1
      XX = Z(I,KK)
      P = 1.0*DX(I,KK)/BDEPTH
      CALL FIXED (SL,XX,P,Z,F,V,A1,A2,I,L,LL,TYP,SL1,SL2,DO,D1,D2)
      Q1(I) = Q1(I) + A1
      Q2(I) = Q2(I) + A2
      CALL BALAN (B,A1,A2,I,L,LL)
      CALL CARRY (A,B,C,D,L,LL)
      CALL MEMFC (SL,Z,F,V,B,SK1,SK2,SK3,Q1,Q2,L,LL,LR,LD)
      CALL CHANG (AA,BB,F,V,N,L,LL)
      DO 600 K = 1,L1
        B(K) = 0.0
        Q1(K) =0.0
        Q2(K) = 0.0
      DO 600 J= 1,L2
        F(K,J) = 0.0
        V(K,J) = 0.0
600   CONTINUE
      DO 200 I =1,N
       BB(N+1,I) = BB(N,I)
       IF(I.EQ.N) BB(N+1,I) = BB(N,I)-1
200   CONTINUE
      RETURN
      END
C_____________________________________________________________________
C
C   FOR FIXED END POINT LOAD
C
      SUBROUTINE FIXED(SL,X,P,Z,F,V,A1,A2,I,L,LL,TYP,SL1,SL2,DO,D1,D2)
C
      INTEGER  TYP
      DIMENSION SL(1),Z(9,1),F(9,1),V(9,1),LL(1)
     * ,TYP(1),SL1(1),SL2(1),DO(1),D1(1),D2(1)
C
      X2 = X/SL(I)
      CALL HAUNCH(S1,S2,S3,I,FMA,FMB,TYP(I),X2,SL(I),SL1(I),SL2(I),DO(I)
     * ,D1(I),D2(I))
      A1 = FMA*SL(I)
      A2 = -FMB*SL(I)
      CALL POINT(Z,X,SL(I),P,V,F,I,L,LL)
      RETURN
      END
C
C
C_____________________________________________________________________
C
C  FOR BALANCE MOMENT
C
      SUBROUTINE BALAN(B,A1,A2,I,L,LL)
C
      DIMENSION B(1),LL(1)
      I1 = I + 1
      B(I) = B(I) - A1
      B(I1) = B(I1) - A2
      RETURN
      END
C
```

```
C
C
C  FOR CARRY MOMENT
C
      SUBROUTINE CARRY(A,B,C,D,L,LL)
C
      DIMENSION A(1),B(1),C(1),D(1),LL(1)
C
      L1 = L+1
      DO 100 I = 2,L1
        I1 = I - 1
  100 B(I) = B(I) - A(I)*B(I1)
      B(L1) = B(L1)/D(L1)
      DO 200 II=1,L
        I = L1-II
        I1 = I+1
  200 B(I) = (B(I)-C(I)*B(I1))/D(I)
      RETURN
      END
C
C
C
C
C  FOR FIND MEMBER FORCE
C
      SUBROUTINE MEMFC(SL,Z,F,V,B,SK1,SK2,SK3,Q1,Q2,L,LL,LR,LD)
C
      DIMENSION SL(1),Z(9,1),F(9,1),V(9,1),Q1(1),Q2(1)
     *,SK1(1),SK2(1),SK3(1),B(1),LL(1)
C
      DO 100 I= 1,L
        I1 = I+1
        R1 = B(I)
        R2 = B(I1)
        S1 = SK1(I)
        S2 = SK2(I)
        S3 = SK3(I)
        P1 = S1*R1+S3*R2
        P2 = S3*R1+S2*R2
        P1 = P1 + Q1(I)
        P2 = P2 + Q2(I)
        CALL REACT(SL,Z,P1,P2,V,F,I,L,LL)
  100 CONTINUE
      RETURN
      END
C
C
C
C
C  FOR SET DISTANCE THIS POINT
C
      SUBROUTINE SETPT(SL,Z,I,L,LL)
C
      DIMENSION Z(9,1),SL(1),LL(1)
      D = SL(I)/LL(I)
      L1 = LL(I)+1
      DO 100 J = 1 , L1
  100 Z(I,J) = (J-1)*D
      RETURN
      END
```

```
C
C_____________________________________________________________
C
C  FOR SHEAR AND MOMENT EACH POINT FROM POINT LOAD P
C
      SUBROUTINE POINT(Z, X, SO, P, V, F, I, L, LL)
C
      DIMENSION Z(9,1), F(9,1), V(9,1), LL(1)
      L1 = LL(I) + 1
      J = 0
  100 J = J+1
      IF (J.GT.L1) RETURN
      Z1 = Z(I,J)
      IF (Z1.GT.X) GOTO 200
      V2 = P*(SO-X)/SO
      F2 = P*(SO-X)/SO*Z1
  300 V(I,J) = V(I,J) + V2
      F(I,J) = F(I,J) + F2
      GOTO 100
  200 V2 = -(P*X/SO)
      F2 = P*X/SO*(SO-Z1)
      GOTO 300
      END
C
C
C_____________________________________________________________
C
C  FOR REACTION
C
      SUBROUTINE REACT(SL, Z, P1, P2, V, F, I, L, LL)
C
      DIMENSION SL(1), Z(9,1), V(9,1), F(9,1), LL(1)
C
      L1 = LL(I)+1
      SO = SL(I)
      J = 0
  100 J = J+1
      IF (J.GT.L1) RETURN
      V3 = -(P1+P2)/SO
      F3 = P1 + V3*Z(I,J)
      V(I,J) = V(I,J) + V3
      F(I,J) = F(I,J) + F3
      GOTO 100
      END
C
C
C_____________________________________________________________
C
C  FOR CHANGE INFORM INFLUENCE LINE
C
      SUBROUTINE CHANG(AA, BB, F, V, N, L, LL)
C
      DIMENSION AA(100,1), BB(100,1), F(9,1), V(9,1), LL(1)
C
      NN = 0
      DO 100 I = 1,L
      K = LL(I) + 1
C     IF (I.EQ.L) K = LL(I)
      DO 100 J = 2,K
```

```
      NN = NN + 1
      AA(NN,N) = F(I,J)
      BB(NN,N) = V(I,J)
  100 CONTINUE
      RETURN
      END
C
C
C-----------------------------------------------------------------
C
C  FOR FIND SUM MAX MIN IN MATRIX AA
C
      SUBROUTINE ORDIN(AA,A1,N,L1,L,LL,Z1)
C
      DIMENSION AA(100,1),A1(100,1),LL(1),H1(9),Z1(1)
C
      I = 0
      DO 200 I1 = 1,L
      DO 200 I2 = 1,LL(I1)
      IF ((I1.EQ.L.AND.I2.EQ.LL(L)).AND.L1.EQ.1) GOTO 300
      I = I + 1
      G = 0.0
      H = 0.0
      H2 = 0.0
      A1(I,1) = 0.0
      A1(I,2) = 0.0
C     A1(I,3) = 0.0
      J = 0
      DO 500 J1 = 1,L
      DO 400 J2 = 1,LL(J1)
      J = J + 1
      IF (J1.EQ.L.AND.J2.EQ.LL(L)) GOTO 400
C     A1(I,3) = AA(I,J)*Z1(J) + A1(I,3)
      IF (AA(I,J) .GE. 0.0) A1(I,1) = AA(I,J)+A1(I,1)
      IF (AA(I,J) .LT. 0.0) A1(I,2) = AA(I,J)+A1(I,2)
      IF (AA(I,J) .GE. G .AND. AA(I,J) .GE. 0.0) G=AA(I,J)
      IF (AA(I,J) .LE. H .AND. AA(I,J) .LE. 0.0) H=AA(I,J)
  400 CONTINUE
      IF (L1.EQ.2 .AND. H2.GT.H) H2=H
      H1(J1) = H
      H = 0.0
  500 CONTINUE
      A1(I,3) = A1(I,1) + A1(I,2)
      A1(I,4) = G
      IF (I2.LT. 0.5*LL(I1) .AND. I1.NE.1) A1(I,5) = H1(I1)+H1(I1-1)
      IF (I2.GT. 0.5*LL(I1) .AND. I1.NE.L) A1(I,5) = H1(I1)+H1(I1+1)
  200 CONTINUE
  300 RETURN
      END
C
C-----------------------------------------------------------------
C
      SUBROUTINE TRANS
C
      COMMON/YY/AA(100,100),BB(100,100),A1(100,20),B1(100,20),DF(100)
     * ,A11(100,20),B11(100,20)
        DO 20 I =1,100
        DO 30 J =1,20
          A11(I,J) = A1(I,J)
          B11(I,J) = B1(I,J)
```

```
  30    CONTINUE
  20    CONTINUE
        RETURN
        END
C_____________________________________________________________________
C
C   FOR GIRDER  EQUIVALENT - LOADING
C
      SUBROUTINE EQUIV (A1,A2,A21,K,DF,DN,L,LL,LR,LD)
      COMMON /BEAM/TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D01(9),D02(9)
      COMMON /DIST/X(9,50),DX(9,50)
      COMMON /BDEPTH/ BDEPTH
      COMMON/LOAD/DD1,DE1,DI1,DLL2,PLM,PLS
      COMMON/PBEAM/BWIDTH,SINTER
      DIMENSION A1(100,1),A2(100,1),A21(100,1),LL(1),DF(1),DN(1),D(150)
C
      DDD1 =(DD1*453.6)/(0.3048*SINTER)
      DLLL2=(DLL2*453.6)/(0.3048*SINTER)
      PLMM = PLM*453.6
C
      WRITE(LD,*)
      WRITE(LD,*) '  UNIFORM DEAD LOAD INTERVAL  =',DDD1,'KG/M'
      WRITE(LD,*) '  UNIFORM LIVE LOAD INTERVAL  =',DLLL2,'KG/M'
      WRITE(LD,*) '  CONCENTRATED LOAD EQUIVALA  =',PLMM,'KG'
C
C
      DO 100 I = 1,K
      DO 100 J = 1,11
        IF (J.EQ.1) A1(I,J) = A21(I,3)*DD1
        IF (J.EQ.2) A1(I,J) = A2(I,1)*DLL2
        IF (J.EQ.3) A1(I,J) = A2(I,2)*DLL2
        IF (J.EQ.4) A1(I,J) = A2(I,4)*PLM
        IF (J.EQ.5) A1(I,J) = A2(I,5)*PLM
        IF (J.EQ.6) A1(I,J) = A1(I,2)+A1(I,4)
        IF (J.EQ.7) A1(I,J) = A1(I,3)+A1(I,5)
        IF (J.EQ.8) A1(I,J) = A1(I,6)*DF(I)
        IF (J.EQ.9) A1(I,J) = A1(I,7)*DN(I)
        IF (J.EQ.10) A1(I,J) = A1(I,1)+A1(I,6)+A1(I,8)
        IF (J.EQ.11) A1(I,J) = A1(I,1)+A1(I,7)+A1(I,9)
        IF (A1(I,10).LT.0.0) A1(I,10) = 0.0
        IF (A1(I,11) .GT.0.0) A1(I,11) = 0.0
 100    CONTINUE
        RETURN
        END
C-----------------------------------------------------------------
C  FOR GIRDER   EQUIVALENT- LOADING
        SUBROUTINE  EQUIVS (B1,B2,B21,K,DF,DN,L,LL,LR,LD)
        COMMON /BEAM/TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D1(9),D2(9)
        COMMON /DIST/ X(9,50),DX(9,50)
        COMMON /LOAD/ DD1,DE1,DI1,DLL2,PLM,PLS
        COMMON /BDEPTH/BDEPTH
        COMMON /PBEAM/ BWIDTH,SINTER
        DIMENSION  B1(100,1),B2(100,1),B21(100,1),LL(1),DF(1),DN(1)
     *  ,D(150),DD(150)
C
        DDD1 = (DD1*453.6)/(SINTER*0.3048)
        DLLL2= (DLL2*453.6)/(SINTER*0.3048)
        PLSS = PLS*453.6
C
        WRITE(LD,*)
```

```
      WRITE(LD,*)' UNIFORM  DEAD  LOAD INTERVAL=',DDD1,'KG/M'
      WRITE(LD,*)' UNIFORM  LIVE  LOAD INTERVAL=',DLLL2,'KG/M'
      WRITE(LD,*)' CONCENTRATED   LOAD INTERVAL=',PLSS,'KG'
C
       DO 100 I=1,K
       DO 100 J=1,11
        IF(J.EQ.1) B1(I,J) = B21(I,3)*DD1
        IF(J.EQ.2) B1(I,J) = B2(I,1)*DLL2
        IF(J.EQ.3) B1(I,J) = B2(I,2)*DLL2
        IF(J.EQ.4) B1(I,J) = B2(I,4)*PLS
        IF(J.EQ.5) B1(I,J) = B2(I,5)*PLS
        IF(J.EQ.6) B1(I,J) = B1(I,2)+B1(I,4)
        IF(J.EQ.7) B1(I,J) = B1(I,3)+B1(I,5)
        IF(J.EQ.8) B1(I,J) = B1(I,6)*DF(I)
        IF(J.EQ.9) B1(I,J) = B1(I,7)*DN(I)
        IF(J.EQ.10)B1(I,J) = B1(I,1)+B1(I,6)+B1(I,8)
        IF(J.EQ.11)B1(I,J) = B1(I,1)+B1(I,7)+B1(I,9)
        IF(B1(I,10).LT.0.0) B1(I,10) =0.0
        IF(B1(I,11).GT.0.0) B1(I,11) =0.0
  100   CONTINUE
        RETURN
        END
C
C_____________________________________________________________
C
C  FOR COMPUTE IMPACT LOADING
C
      SUBROUTINE IMPAC(DF,DN,N1,SL,L,LL)
C
      DIMENSION LL(1),DF(1),DN(1),SL(1)
C
      N = 0
      DO 100 I=1,L
      DO 100 J=1,LL(I)
      N=N+1
      DF(N) = 50./(125.+SL(I))
      DN(N) =0.0
      IF (J.GT.(LL(I)/2)) DN(N)=50./(125.+(SL(I)+SL(I+1))/2)
      IF (J.LT.(LL(I)/2).AND.I.NE.1) DN(N)=50./(125.+(SL(I-1)+SL(I))/2)
      IF (J.GT.(LL(I)/2).AND.I.EQ.L) DN(N) =0.0
      IF (N1.EQ.2) DN(N) = DF(N)
      IF (DF(N).GT.0.3) DF(N) = 0.3
      IF (DN(N).GT.0.3) DN(N) = 0.3
  100 CONTINUE
      RETURN
      END
C
C
C_____________________________________________________________
C
C  FOR OUTPUT DATA
C
      SUBROUTINE OUTPUT(A,B,C,N,L,LL,LR,LD)
C
      DIMENSION A(100,1),B(100,1),C(7,1),LL(1)
C
      A(0,10) = 0.0
      A(N+1,10) = 0.0
      A(0,11) = 0.0
      A(N+1,11) = 0.0
```

```
      I = -1
      I2 = 0
      DO 200 II = 1,L
        L1 = LL(II)+1
        DO 100 J1 = 1,L1
        I = I+1
        I2 = I2+1
        B1 = B(I2,10)
        B2 = B(I2,11)
        IF (J1.EQ.L1) THEN
          I2 =  I2-1
          IF (I2.EQ.N) THEN
              B((I2+1),10) = 0.0
              B((I2+1),11) = 0.0
          ENDIF
          B1 = B(I2,10) - B((I2+1),11)
          B2 = B((I2+1),10)-B(I2,11)
        ENDIF
       B3 = B1
       IF( ABS(B1).LT.ABS(B2)) B3=ABS(B2)
       WRITE(LD,50) C(II,J1),A(I,10),A(I,11),B3
 50    FORMAT(5F10.2)
       IF (J1.EQ.L1) I=I-1
 100   CONTINUE
 200   CONTINUE
       RETURN
       END
C
C
C_____________________________________________________________________
C
      SUBROUTINE HAUNCH(KA,KB,KAB,K,FMA,FMB,K1,B,L,L1,L2,D,D1,D2)
C
      DIMENSION I2(25),S(25),A(25),M1(25),M2(25),M3(25)
      REAL L,L1,L2,I1,I2,KA,KB,KAB
C
      X=0.0
      F1=0.0
      F2=0.0
      F3=0.0
      C0=0.0
      C1=0.0
      N=20
      M=N+1
      IF (K1.EQ.1) CALL STRAI(X,L,L1,L2,D,D1,D2,I2,N,M)
      IF (K1.EQ.2) CALL PARAB(X,L,L1,L2,D,D1,D2,I2,N,M)
      IF (K1.EQ.3) CALL STRAI(X,L,L1,L2,D,D1,D2,I2,N,M)
      I1 = D*D*D/12.
      DO 100 I=2,N
        MM = I/2
        A1 = I/2
        MM = MM*10
        IA1 = A1*10
        IF(IA1 .EQ.  MM) S(I)=4.0
        IF(IA1 .GT.  MM) S(I)=2.0
 100  CONTINUE
C
      S(1) = 1.0
      S(M) = 1.0
      X =0.0
```

```
      DO 200 I=1,M
        M1(I) = (L-X)*(L-X)
        M2(I) = X*X
        M3(I) = X*(L-X)
         X = X + L/N
  200 CONTINUE
C
      DO 300 I=1,M
        A(I) = S(I)*I1/I2(I)/(3*N)
  300 CONTINUE
C
      X = 0.0
      DO 400 I=1,M
         F1 = F1+M1(I)*A(I)
         F2 = F2+M2(I)*A(I)
         F3 = F3+M3(I)*A(I)
         IF (X.LT.(0.999*B*L)) THEN
            FF1 = F1
            FF2 = F2
            FF3 = F3
         ENDIF
         X = X + 1/N
  400 CONTINUE
      F4 = FF2*(1-B)
      F5 = (F1-FF1)*(B)
      F6 = (F3-FF3)*(B)
      F7 = FF3*(1-B)
      C0 = -(F4+F6)
      C1 = -(F5+F7)
      FMB = (C0*F1-C1*F3)/(F2*F1-F3*F3)
      FMA = (C1*F2-C0*F3)/(F2*F1-F3*F3)
      KA = F2/(F1*F2-F3*F3)*L*L
      KB = F1/(F1*F2-F3*F3)*L*L
      CA = F3/F2
      CB = F3/F1
      KAB = CA*KA
      RETURN
      END
C
C
C_____________________________________________________________
C
      SUBROUTINE STRAI(X,L,L1,L2,D,D1,D2,RL,N,M)
C
      DIMENSION RL(1)
      REAL L,L1,L2
C
      DO 100 I=1,M
         DX = D
         IF (X.LT.L1) DX = D+(L1-X)*(D1-D)/L1
         IF (X.GT.(L-L2)) DX = D+(X+L2-L)*(D2-D)/L2
         RL(I) = DX*DX*DX/12.
         X = X+L/N
  100  CONTINUE
       RETURN
       END
C
C
C_____________________________________________________________
C
```

```
      SUBROUTINE  PARAB(X,L,L1,L2,D,D1,D2,I2,N,M)
C
      DIMENSION I2(1)
      REAL L,L1,L2,I2
C
      DO 100 I=1,M
        DX = D
        IF (X.LT.L1) DX = D + (L1-X)**2*(D1-D)/L1**2
        IF (X.GT.(L-L2)) DX = D+(X+L2-L)**2*(D2-D)/L2**2
        I2(I) = DX*DX*DX/12
        X = X+L/N
  100 CONTINUE
      RETURN
      END
C
C
C-----------------------------------------------------------------------
C
C     SUBROUTINE FIND TRUCK LOAD
C
      SUBROUTINE TRUCK(SL,LL,K)
C
      DIMENSION SL(1),LL(1)
      COMMON /PBEAM/ BWIDTH,SINTER
      COMMON /YY/ B(100,100),XX1(100,100),A(100,20),A1(100,20),XX2(100)
      COMMON /OVERFLOW/ CCC(100,100)
C
      R = 4.32/SINTER
      IR = R
      F = R-FLOAT(IR)
      FACT = F/SINTER
      DO 500 I2 = 1,K
         DO 500 J2 = 1,K
           CCC(I2,J2) = -B(I2,J2)
  500 CONTINUE
      DO 100 I=1,K
        G = 0.0
        H = 0.0
        DO 300 J1 = 1,K
           IF (B(I,J1).GE.G.AND.B(I,J1).GT.0.0) G = B(I,J1)
           IF (B(I,J1).LE.H.AND.B(I,J1).LT.0.0) H = B(I,J1)
  300 CONTINUE
      DO 200 J=1,K
         B(I,0) = 0.0
         B(I,K+1)=0.0
         II = B(I,J)*1000
         JJ = G*1000
         KK = H*1000
         IF (II.EQ.JJ) CALL FINDS(A,B,I,J,IR,1,-FACT,K)
         IF (II.EQ.KK) THEN
           CCC(I,0) = 0.0
           CCC(I,K+1) = 0.0
           CALL FINDS(A1,CCC,I,J,IR,1,-FACT,K)
         ENDIF
  200 CONTINUE
  100 CONTINUE
      RETURN
      END
C
```

```
C
      SUBROUTINE  FINDS(A,B,I,J,IR,F,K)
C
      DIMENSION A(100,1),B(100,1)
C
      J1 = J+IR
      J2 = J1+1
      J3 = J2+1
      J4 = J-IR
      J5 = J4-1
      J6 = J5-1
      IF ((J5.GT.0).AND.(J2.LT.K+1)) THEN
         A(I,1) = B(I,J)
         A(I,2) = B(I,J5)+F*(B(I,J4)-B(I,J5))
         A(I,3) = B(I,J2)+F*(B(I,J1)-B(I,J2))
      ENDIF
      IF ((J5.LE.0).OR.(A(I,2).LT.0.0)) THEN
         A(I,1) = B(I,J2)+F*(B(I,J1)-B(I,J2))
         A(I,2) = B(I,J)
         A(I,3) = B(I,J3)+F*(B(I,J2)-B(I,J3))
      ENDIF
      IF ((J2.GE.K+1) .OR.  (A(I,3).LT.0.0)) THEN
         A(I,1) = B(I,J5)+F*(B(I,J4)-B(I,J5))
         A(I,2) = B(I,J6)+F*(B(I,J5)-B(I,J6))
         A(I,3) = B(I,J)
      ENDIF
      IF (A(I,1).LE.0.0) A(I,1) = 0.0
      IF (A(I,2).LE.0.0) A(I,2) = 0.0
      IF (A(I,3).LE.0.0) A(I,3) = 0.0
      IF (A(I,2).LT.A(I,3)) THEN
         TT = A(I,2)
         A(I,2) = A(I,3)
         A(I,3) = TT
      ENDIF
      RETURN
      END
C
C
C_____________________________________________________________________
C
C  FOR SET TABLE OF TRUCK LOADING
C
      SUBROUTINE TRUCKT(DF,DN,K)
C
      DIMENSION DF(1),DN(1)
      COMMON /YY/ A(100,100),B(100,100),A1(100,20),B1(100,20),X1(100)
      COMMON /YY1/C(100,100),D(100,100)
      COMMON /YY5/AAAA(100,1),BBBB(100,1)
      COMMON /YY6/AAAAA(100,1),BBBBB(100,1)
C
      DO 100 I=1,K
        A(I,1) = AAAA(I,1)
        A(I,2) = A1(I,1)+A1(I,2)
        A(I,3) = -B1(I,1)-B1(I,2)
        A(I,4) = A1(I,3)
        A(I,5) = -B1(I,3)
        A(I,6) = A(I,2)*16.+A(I,4)*4.
        A(I,7) = A(I,3)*16.+A(I,5)*4.
        A(I,8) = A(I,6)*DF(I)
        A(I,9) = A(I,7)*DN(I)
```

```
      A(I,10) = A(I,1)+A(I,6)+A(I,8)
      A(I,11) = A(I,1)+A(I,7)+A(I,9)
      IF (A(I,10).LT.0.0) A(I,10) = 0.0
      IF (A(I,11).GT.0.0) A(I,11) = 0.0
100   CONTINUE
C
       DO 200 I=1,K
      C(I,1) = AAAAA(I,1)
      C(I,2) = A1(I,1)+A1(I,2)
      C(I,3) = -B1(I,1)-B1(I,2)
      C(I,4) = A1(I,3)
      C(I,5) = -B1(I,3)
      C(I,6) = C(I,2)*16.+C(I,4)*4.
      C(I,7) = C(I,3)*16.+C(I,5)*4.
      C(I,8) = C(I,6)*DF(I)
      C(I,9) = C(I,7)*DN(I)
      C(I,10)= C(I,1)+C(I,6)+C(I,8)
      C(I,11)= C(I,1)+C(I,7)+C(I,9)
       IF(C(I,10).LT.0.0) C(I,10) = 0.0
       IF(C(I,11).GT.0.0) C(I,11) = 0.0
200   CONTINUE
      RETURN
       END
C
C
C
C     SUBROUTINE FIND TRUCK LOAD
C
      SUBROUTINE TRUCKS(SL,LL,K)
      COMMON /PBEAM/ BWIDTH,SINTER
      COMMON /YY/ XX1(100,100),B(100,100),A(100,20),A1(100,20),XX2(100)
      COMMON /OVERFLOW/ CCC(100,100)
C
      R = 4.32/SINTER
      IR = R
      F = R - FLOAT(IR)
      FACT = F/SINTER
      DO 500 I2 = 1,K
        DO 500 J2 = 1,K
          CCC(I2,J2) =  -1. * B(I2,J2)
500   CONTINUE
      DO 100 I=1,K
        G = 0.0
        H = 0.0
        DO 300 J1=1,K
          IF (B(I,J1).GE.G .AND. B(I,J1).GT.0.0) G = B(I,J1)
          IF (B(I,J1).LE.H .AND. B(I,J1).LT.0.0) H = B(I,J1)
300   CONTINUE
      DO 200 J=1,K
         B(I,0) = 0.0
         B(I,K+1) =0.0
         II = B(I,J)*1000
         JJ = G*1000
         KK = H*1000
         IF (II.EQ.JJ) CALL FINDS(A,B,I,J,IR,1.-FACT,K)
         IF (II.EQ.KK) THEN
           CCC(I,0) = 0.0
           CCC(I,K+1) = 0.0
           CALL FINDS(A1,CCC,I,J,IR,1.-FACT,K)
         ENDIF
```

```
 200  CONTINUE
 100  CONTINUE
      RETURN
      END
C
C
C_____________________________________________________________________
C
C  FOR SET TABLE OF TRUCK LOADING
C
      SUBROUTINE TRUCTS(DF,DN,K)
C
      DIMENSION DF(1),DN(1)
      COMMON /YY/ B(100,100),A(100,100),A1(100,20),B1(100,20),X1(100)
      COMMON /YY1/C(100,100),D(100,100)
      COMMON /YY5/ AAAA(100,1),BBBB(100,1)
      COMMON /YY6/ AAAAA(100,1),BBBBB(100,1)
C
      DO 100 I=1,K
        A(I,1) = BBBB(I,1)
        A(I,2) = A1(I,1)+A1(I,2)
        A(I,3) = -B1(I,1)-B1(I,2)
        A(I,4) = A1(I,3)
        A(I,5) = -B1(I,3)
        A(I,6) = A(I,2)*16.+A(I,4)*4.
        A(I,7) = A(I,3)*16.+A(I,5)*4.
        A(I,8) = A(I,6)*DF(I)
        A(I,9) = A(I,7)*DN(I)
        A(I,10) = A(I,1)+A(I,6)+A(I,8)
        A(I,11) = A(I,1)+A(I,7)+A(I,9)
        IF (A(I,10).LT.0.0) A(I,10) = 0.0
        IF (A(I,11).GT.0.0) A(I,11) = 0.0
 100    CONTINUE
C
        DO 200 I=1,K
          D(I,1) = BBBBB(I,1)
          D(I,2) = A1(I,1)+A1(I,2)
          D(I,3) = -B1(I,1)-B1(I,2)
          D(I,4) = A1(I,3)
          D(I,5) = -B1(I,3)
          D(I,6) = D(I,2)*16.+D(I,4)*4.
          D(I,7) = D(I,3)*16.+D(I,5)*4.
          D(I,8) = D(I,6)*DF(I)
          D(I,9) = D(I,7)*DN(I)
          D(I,10)= D(I,1)+D(I,6)+D(I,8)
          D(I,11)= D(I,1)+D(I,7)+D(I,9)
           IF(D(I,10).LT.0.0) D(I,10) = 0.0
           IF(D(I,11).GT.0.0) D(I,11) = 0.0
 200      CONTINUE
          RETURN
           END
C_____________________________________________________________________
C
C   INPUT BEAM GEOMETRY FOR HAUNCH
C
      SUBROUTINE HADAT2(L)
C
      INTEGER  TYP
      COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D1(9),D2(9)
C
```

```
      DO 200 I=1,
      CALL HAUNC(
     200 CONTINUE
      RETURN
      END
C
C
C
C
C
      SUBROUTINE
C
      REAL K,KK,
      REAL L,L1,
      COMMON /ST
      DIMENSION
C
      X=0.0
      F1=0.0
      F2=0.0
      F3=0.0
      C0=0.0
      C1=0.0
      N=20
      M=N+1
      IF (K1.EQ.
      IF (K1.EQ.
      IF (K1.EQ.
      I1 = D*D*C
      DO 100 I=2,
        NM = I/2
        A1 = I/2
        NM = NM*
        IA1 = A1
        IF(IA1 .E
        IF(IA1 .E
  100 CONTINUE
C
      S(1) = 1.0
      S(M) = 1.0
      X =0.0
      DO 200 I=1,
        M1(I) =
        M2(I) =
        M3(I) =
          X = X +
  200 CONTINUE
C
      DO 300 I=1,
        A(I) = S(
  300 CONTINUE
C
      X = 0.0
      DO 400 I=1,
         F1 = F1+
         F2 = F2+
         F3 = F3+
         IF (X.LT.
            FF1 =
            FF2 =
```

```
            FF3 = F3
          ENDIF
          X = X + L/N
 400   CONTINUE
       F4 = FF2*(1-B)
       F5 = (F1-FF1)*(B)
       F6 = (F3-FF3)*(B)
       F7 = FF3*(1-B)
       C0 = -(F4+F6)
       C1 = -(F5+F7)
       FMB = (C0*F1-C1*F3)/(F2*F1-F3*F3)
       FMA = (C1*F2-C0*F3)/(F2*F1-F3*F3)
       KA = F2/(F1*F2-F3*F3)*L*L
       KB = F1/(F1*F2-F3*F3)*L*L
       CA = F3/F2
       CB = F3/F1
       KAB = CA*KA
C
       COF(J,1)=CA
       COF(J,2)=CB
       KK(J,1)=KA
       KK(J,2)=KB
       KR(J)=KAB
C        .   .
       RETURN
          END
C
C
C
C-------------------------------------------------------------------------------
C
       SUBROUTINE DISTAN(L)
C
       INTEGER TYP
       COMMON /PBEAM/ BWIDTH,SINTER
       COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D1(9),D2(9)
       COMMON /DIST/ X(9,50),DX(9,50)
C
C
C  Find Lenght along the Span
       WRITE(*,*)
       DO 200 I=1,L
       X(I,1)=0.0
       DO 200 J=2,NSC(I)
       X(I,J)=X(I,J-1)+SINTER
  200 CONTINUE
C
C
C  Find Depth along the Span
       WRITE(*,*)
       DO 250 I=1,L
       IF (TYP(I).EQ.1) GOTO 300
       IF (TYP(I).EQ.2) GOTO 400
       DO 270 J=1,NSC(I)
       DX(I,J) = D0(I)
  270  CONTINUE
       GOTO 250
C
  300  WRITE(*,*)
       DO 350 J=1,NSC(I)
```

```
      DX(I,J) = DO(I)
      IF (X(I,J).LT.SL1(I)) THEN
         DX(I,J) = DO(I)+(SL1(I)-X(I,J))*(D1(I)-DO(I))/SL1(I)
         ENDIF
      IF (X(I,J).GT.(SL(I)-SL2(I))) THEN
         DX(I,J) = DO(I)+(X(I,J)+SL2(I)-SL(I))*(D2(I)-DO(I))/SL2(I)
         ENDIF
 350  CONTINUE
      GOTO 250
C
 400  WRITE(*,*)
      DO 450 J=1,NSC(I)
      DX(I,J) = DO(I)
      IF (X(I,J).LT.SL1(I)) THEN
        DX(I,J) = DO(I)+(SL1(I)-X(I,J))**2*(D1(I)-DO(I))/SL1(I)**2
      ENDIF
      IF (X(I,J).GT.(SL(I)-SL2(I))) THEN
        DX(I,J) = DO(I)+(X(I,J)+SL2(I)-SL(I))**2*(D2(I)-DO(I))/SL2(I)**2
        ENDIF
 450  CONTINUE
 250  CONTINUE
C
      RETURN
        END
C
C
C
C
C_____________________________________________________________________
C
C
C
C
C                 *--------------------------*
C                      SUBROUTINE  PRESTM(L)
C                 *--------------------------*
C
C
C  Find Fixed end Moment
C
      REAL IC
      REAL K,KK,KR
      INTEGER TYP
      COMMON /STM1/ COF(9,2),K(9,2),KK(9,2),KR(9)
      COMMON /STM2/ DF(9,2),FEM(9,2),BAL(9,2),CO(9,2),T(9,2)
      COMMON /SETTLE/ STM(10)
      COMMON /PBEAM/ BWIDTH,SINTER
      COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),DO(9),D1(9),D2(9)
      COMMON /DIST/ X(9,50),DX(9,50)
      COMMON /MATFR/FCP,FSY
C
      CALL HADAT2(L)
C
C
C
C
C  Set Parameter
C
      DO 1 I=1,L
      K(I,1)=KK(I,1)/SL(I)
      K(I,2)=KK(I,2)/SL(I)
    1 CONTINUE
```

```
      DF(1,1)=1.00
      DF(1,2)=K(1,2)/(K(1,2)+K(2,1))
      DO 2 I=2,L
      DF(I,1)=K(I,1)/(K(I-1,2)+K(I,1))
      DF(I,2)=K(I,2)/(K(I,2)+K(I+1,1))
    2 CONTINUE
C
C
      E=(15120*(FCP*0.0703)**0.5*144)/(0.0703*1000)
      DO 3 I=1,L
      IC= BWIDTH*DO(I)*DO(I)*DO(I)/12
      RR= (STM(I+1)-STM(I))/SL(I)
      FEM(I,1)=(KK(I,1)+KR(I))*2*E*IC*RR/SL(I)
      FEM(I,2)=(KK(I,2)+KR(I))*2*E*IC*RR/SL(I)
    3 CONTINUE
      FEM(L+1,1)= 0.0
C
C
C
C
C  Find Fixed End Moment
C
      BAL(1,1)=-1.0*FEM(I,1)*DF(1,1)
      BAL(1,2)=-1.0*(FEM(1,2)+FEM(2,1))*DF(1,2)
      DO 5 I=2,L
        BAL(I,1)=-1.0*(FEM(I-1,2)+FEM(I,1))*DF(I,1)
        BAL(I,2)=-1.0*(FEM(I,2)+FEM(I+1,1))*DF(I,2)
    5 CONTINUE
C
      DO 7 I=1,L
      T(I,1) = FEM(I,1) + BAL(I,1)
      T(I,2) = FEM(I,2) + BAL(I,2)
    7 CONTINUE
C
C
      CO(L+1,1) = 0.0
C
   10 DO 40 I=1,L
        CO(I,1)=BAL(I,2)*COF(I,2)
        CO(I,2)=BAL(I,1)*COF(I,1)
   40 CONTINUE
      BAL(1,1)=-1.0*CO(1,1)*DF(1,1)
      BAL(1,2)=-1.0*(CO(1,2)+CO(2,1))*DF(1,2)
      DO 50 I=2,L
        BAL(I,1)=-1.0*(CO(I-1,2)+CO(I,1))*DF(I,1)
        BAL(I,2)=-1.0*(CO(I,2)+CO(I+1,1))*DF(I,2)
   50 CONTINUE
C
      DO 60 I=1,L
      IF(ABS(CO(I,1)).GT.0.001) GOTO 20
      IF(ABS(CO(I,2)).GT.0.001) GOTO 20
   60 CONTINUE
      GOTO 30
C
   20 DO 70 I=1,L
      T(I,1) = T(I,1)+BAL(I,1)+CO(I,1)
      T(I,2) = T(I,2)+BAL(I,2)+CO(I,2)
   70 CONTINUE
```

```
      GOTO 10
C
 30   WRITE(*,*)
 100  FORMAT(1X,10F9.3)
C
      CALL MIDSTM(L)
C
      RETURN
      END
C
C
C
C
C              *----------------------------*
                   SUBROUTINE  MIDSTM(L)
C              *----------------------------*
C
C
C  find Moment of each secment
C
      REAL IC
      REAL K,KK,KR
      INTEGER TYP
      COMMON /STM1/ COF(9,2),K(9,2),KK(9,2),KR(9)
      COMMON /STM2/ DF(9,2),FEM(9,2),BAL(9,2),CO(9,2),T(9,2)
C
      COMMON /SETTLE/ STM(10)
      COMMON /PBEAM/ BWIDTH,SINTER
      COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D1(9),D2(9)
C
      COMMON /DIST/ X(9,50),DX(9,50)
C
      COMMON /STM3/ R(9,2),BM(9,50),V(9,50)
C
C
      DO 10 I=1,L
      R(I,1)=-(T(I,1)+T(I,2))/SL(I)
      R(I,2)=-R(I,1)
  10  CONTINUE
C
      DO 20 I=1,L
      DO 20 J=1,NSC(I)
         BM(I,J)= T(I,1)+R(I,1)*X(I,J)
        IF(X(I,J).GT.0.0) THEN
        V(I,J)=(T(I,1)-BM(I,J))/X(I,J)
          ELSE
          V(I,J)=R(I,2)
        ENDIF
  20  CONTINUE
C
      RETURN
        END
C
C
C
C              *----------------------------*
                   SUBROUTINE  FINSTM(L)
C              *----------------------------*
C
```

```
C   find Moment of each secment
C
      INTEGER TYP
      COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D1(9),D2(9)
C
      COMMON /YY/ AA(100,100),BB(100,100),A1(100,20),B1(100,20),DF(100)
      COMMON /YY1/AA1(100,100),BB1(100,100)
      COMMON /YY4/ AAA(100,12),BBB(100,12)
      COMMON /STM3/ R(9,2),BM(9,50),V(9,50)
C
C
C
      K = 0
      DO 10 I=1,L
      DO 20 J=1,NSC(I)-1
        JK=J+K
        AA(JK,10)=BM(I,J+1)*0.1383
        AA1(JK,10)=BM(I,J+1)*0.1383
        AAA(JK,10)=BM(I,J+1)*0.1383
   20 CONTINUE
      K = K + NSC(I)-1
   10 CONTINUE
C
C
      K=0
      DO 15 I=1,L
      DO 25 J=1,NSC(I)-1
        JK=J+K
        AA(JK,11)=AA(JK,1)+AA(JK,6)+AA(JK,8)+AA(JK,10)
        AA(JK,12)=AA(JK,1)+AA(JK,7)+AA(JK,9)+AA(JK,10)
          IF (AA(JK,11).LT.0.0) AA(JK,11)=0.0
          IF (AA(JK,12).GT.0.0) AA(JK,12)=0.0
        AA1(JK,11)=AA1(JK,1)+AA1(JK,6)+AA1(JK,8)+AA1(JK,10)
        AA1(JK,12)=AA1(JK,1)+AA1(JK,7)+AA1(JK,9)+AA1(JK,10)
          IF (AA1(JK,11).LT.0.0) AA1(JK,11)= 0.0
          IF (AA1(JK,12).GT.0.0) AA1(JK,12)= 0.0
        AAA(JK,11)=AAA(JK,1)+AAA(JK,6)+AAA(JK,8)+AAA(JK,10)
        AAA(JK,12)=AAA(JK,1)+AAA(JK,7)+AAA(JK,9)+AAA(JK,10)
          IF (AAA(JK,11).LT.0.0) AAA(JK,11)=0.0
          IF (AAA(JK,12).GT.0.0) AAA(JK,12)=0.0
   25 CONTINUE
        K=K+NSC(I)-1
   15 CONTINUE
C
C
C
C
      K = 0
      DO 30 I=1,L
      DO 40 J=1,NSC(I)-1
        JK=J+K
        BB(JK,10)=(V(I,J)+V(I,J+1))*0.4536/2
        BB1(JK,10)=(V(I,J)+V(I,J+1))*0.4536/2
        BBB(JK,10)=(V(I,J)+V(I,J+1))*0.4536/2
   40 CONTINUE
        K=K+NSC(I)-1
   30 CONTINUE
C
C
      K=0
```

```
      DO 35 I=1,L
      DO 45 J=1,NSC(I)-1
        JK=J+K
        BB(JK,11)=BB(JK,1)+BB(JK,6)+BB(JK,8)+BB(JK,10)
        BB(JK,12)=BB(JK,1)+BB(JK,7)+BB(JK,9)+BB(JK,10)
          IF (BB(JK,11).LT.0.0) BB(JK,11)=0.0
          IF (BB(JK,12).GT.0.0) BB(JK,12)=0.0
        BB1(JK,11)=BB1(JK,1)+BB1(JK,6)+BB1(JK,8)+BB1(JK,10)
        BB1(JK,12)=BB1(JK,1)+BB1(JK,7)+BB1(JK,9)+BB1(JK,10)
          IF (BB1(JK,11).LT.0.0) BB1(JK,11)=0.0
          IF (BB1(JK,12).GT.0.0) BB1(JK,12)=0.0
        BBB(JK,11)=BBB(JK,1)+BBB(JK,6)+BBB(JK,8)+BBB(JK,10)
        BBB(JK,12)=BBB(JK,1)+BBB(JK,7)+BBB(JK,9)+BBB(JK,10)
          IF (BBB(JK,11).LT.0.0) BBB(JK,11)=0.0
          IF (BBB(JK,12).GT.0.0) BBB(JK,12)=0.0
   45 CONTINUE
        K=K+NSC(I)-1
   35 CONTINUE
C
C
C
      RETURN
         END
C
C
C
C
C
C  ***********************************************************************
C  *                                                                     *
C  *                        S U B   P R O G R A M  # 4                   *
C  *                                                                     *
C  *                      BRIDGE DESIGN AND REPORT OUTPUT                *
C  *                                                                     *
C  ***********************************************************************
C
C                    *-----------------------------------*
C                              SUBROUTINE  BDESIGN
C                    *-----------------------------------*
C
      REAL IMP,IMN,ISF
      REAL IEMP,IEMN,IESP,IESN,ITMP,ITMN,ITSP,ITSN
      INTEGER  TYP
      CHARACTER  HEADI*52
      COMMON /TITLE/ HEADI
      COMMON /MATER/ FCP,FSY
      COMMON /STSIZE/ NOSB,NOST,NOSL
      COMMON /BRGDIA/ NUMTRA,WLANE,EGDAP,NOBEAM,CSL,PAREA,NUMSPA
      COMMON /LCLASS/ LCLASS
      COMMON /SETTLE/ STM(10)
      COMMON /PBEAM/ BWIDTH,SINTER
      COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D1(9),D2(9)
C
      COMMON /BDEPTH/ BDEPTH
      COMMON /CDATA/ FC,FS,MR,CK,CJ,CRR,VMAX,VCON1,VCON2,GDSPC
C
      COMMON /EMOMEN/ EEMP(9,50),EEMN(9,50)
      COMMON /TMOMEN/ ETMP(9,50),ETMN(9,50)
      COMMON /ESHEAR/ EESP(9,50),EESN(9,50)
      COMMON /TSHEAR/ ETSP(9,50),ETSN(9,50)
```

```
      COMMON /IMOMEN/ IEMP(9,50),IEMN(9,50)
      COMMON /TMOMEI/ ITMP(9,50),ITMN(9,50)
      COMMON /ISHEAR/ IESP(9,50),IESN(9,50)
      COMMON /TSHEAI/ ITSP(9,50),ITSN(9,50)
      COMMON /MOMENT/ EMP(9,50),EMN(9,50),IMP(9,50),IMN(9,50)
      COMMON /SFORCE/ ESF(9,50),ISF(9,50)
      COMMON /STEELE/ ASTE(9,50),ASBE(9,50),ASDE(9,50)
      COMMON /STEELI/ ASTI(9,50),ASBI(9,50),ASDI(9,50)
      COMMON /DIST/ X(9,50),DX(9,50)
C
      REWIND 11
      REWIND 13
C
      IF (NUMSPA.EQ.1) GOTO 1
      IF (NUMSPA.EQ.2) GOTO 2
      IF (NUMSPA.EQ.3) GOTO 3
      IF (NUMSPA.EQ.4) GOTO 4
      IF (NUMSPA.EQ.5) GOTO 5
      IF (NUMSPA.EQ.6) GOTO 6
      IF (NUMSPA.EQ.7) GOTO 7
      IF (NUMSPA.EQ.8) GOTO 8
      IF (NUMSPA.EQ.9) GOTO 9
    1   READ(11,801)
        GOTO 10
    2   READ(11,802)
        GOTO 10
    3   READ(11,803)
        GOTO 10
    4   READ(11,804)
        GOTO 10
    5   READ(11,805)
        GOTO 10
    6   READ(11,806)
        GOTO 10
    7   READ(11,807)
        GOTO 10
    8   READ(11,808)
        GOTO 10
    9   READ(11,809)
        GOTO 10
  801 FORMAT(/////////////////////////)
  802 FORMAT(//////////////////////////////)
  803 FORMAT(///////////////////////////////////)
  804 FORMAT(////////////////////////////////////////)
  805 FORMAT(/////////////////////////////////////////////)
  806 FORMAT(//////////////////////////////////////////////////)
  807 FORMAT(////////////////////////////////////////////////////////)
  808 FORMAT(//////////////////////////////////////////////////////////////)
  809 FORMAT(////////////////////////////////////////////////////////////////)
C
C
C
   10 WRITE(*,*)
      WRITE(13,*)
      WRITE(13,*)'*********************************************'
      WRITE(13,*)'*************** BEAM DESIGN ***************'
      WRITE(13,*)'*********************************************'
C
C
C  Read Exterior Equivalent Moment
```

```
      EEMP(1,1)=0.0
      EEMN(1,1)=0.0
C
      DO 30 I=1,NUMSPA
      DO 30 J=2,NSC(I)
        READ(11,999) EEMP(I,J),EEMN(I,J)
        EEMP(I,J) = EEMP(I,J)/0.1383
        EEMN(I,J) = EEMN(I,J)/0.1383
   30 CONTINUE
  999 FORMAT(103x,2F10.3)
C
      DO 31 I=2,NUMSPA
      EEMP(I,1)=EEMP(I-1,NSC(I-1))
      EEMN(I,1)=EEMN(I-1,NSC(I-1))
   31 CONTINUE
C
      DO 32 I=1,NUMSPA
      DO 32 J=1,NSC(I)
   32 CONTINUE
C
C
C
C  Read Exterior Equivalent Shear Force
     READ(11,904)
  904 FORMAT(//////////)
      DO 40 I=1,NUMSPA
      DO 40 J=1,NSC(I)-1
        READ(11,999) EESP(I,J),EESN(I,J)
        EESP(I,J) = EESP(I,J)/0.4536
        EESN(I,J) = EESN(I,J)/0.4536
   40   CONTINUE
C
C
C
C  Read Interior Equivalent Moment
     READ(11,905)
  905 FORMAT(//////////////////)
      IEMP(1,1)=0.0
      IEMN(1,1)=0.0
C
      DO 50 I=1,NUMSPA
      DO 50 J=2,NSC(I)
        READ(11,999) IEMP(I,J),IEMN(I,J)
        IEMP(I,J) = IEMP(I,J)/0.1383
        IEMN(I,J) = IEMN(I,J)/0.1383
   50 CONTINUE
C
      DO 51 I=2,NUMSPA
      IEMP(I,1)=IEMP(I-1,NSC(I-1))
      IEMN(I,1)=IEMN(I-1,NSC(I-1))
   51 CONTINUE
C
      DO 52 I=1,NUMSPA
      DO 52 J=1,NSC(I)
   52 CONTINUE
C
C
C
C  Read Interior Equivalent Shear Force
     READ(11,906)
```

```
  906 FORMAT(//////////)
      DO 60 I=1,NUMSPA
      DO 60 J=1,NSC(I)-1
        READ(11,999) IESP(I,J),IESN(I,J)
        IESP(I,J) = IESP(I,J)/0.4536
        IESN(I,J) = IESN(I,J)/0.4536
   60 CONTINUE
C
C
C
C  Read Exterior Truck Moment
      READ(11,907)
  907 FORMAT(//////////////)
      ETMP(1,1)=0.0
      ETMN(1,1)=0.0
C
      DO 70 I=1,NUMSPA
      DO 70 J=2,NSC(I)
        READ(11,999) ETMP(I,J),ETMN(I,J)
        ETMP(I,J) = ETMP(I,J)/0.1383
        ETMN(I,J) = ETMN(I,J)/0.1383
   70 CONTINUE
C
      DO 71 I=2,NUMSPA
      ETMP(I,1)=ETMP(I-1,NSC(I-1))
      ETMN(I,1)=ETMN(I-1,NSC(I-1))
   71 CONTINUE
C
      DO 72 I=1,NUMSPA
      DO 72 J=1,NSC(I)
   72 CONTINUE
C
C
C
C  Read Exterior Truck Shear Force
      READ(11,908)
  908 FORMAT(//////)
C
      DO 80 I=1,NUMSPA
      DO 80 J=1,NSC(I)-1
        READ(11,999) ETSP(I,J),ETSN(I,J)
        ETSP(I,J) = ETSP(I,J)/0.4536
        ETSN(I,J) = ETSN(I,J)/0.4536
   80 CONTINUE
C
C
C  read interior truck moment
      READ (11,910)
  910 FORMAT(//////////////)
      ITMP (1,1) = 0.0
      ITMN (1,1) = 0.0
      DO 90 I=1,NUMSPA
      DO 90 J=2,NSC(I)
        READ (11,999) ITMP(I,J),ITMN(I,J)
        ITMP(I,J) = ITMP(I,J)/0.1383
        ITMN(I,J) = ITMN(I,J)/0.1383
   90 CONTINUE
      DO 91 I=2,NUMSPA
        ITMP(I,1)=ITMP(I-1,NSC(I-1))
        ITMN(I,1)=ITMN(I-1,NSC(I-1))
```

```
   91     CONTINUE
          DO 92 I=1,NUMSPA
          DO 92 J=1,NSC(I)
   92     CONTINUE
C
C
C  read interior truck shear force
          READ (11,911)
  711     FORMAT(//////)
C
          DO 100 I=1,NUMSPA
          DO 100 J=1,NSC(I)-1
           READ(11,999) ITSP(I,J),ITSN(I,J)
             ITSP(I,J) = ITSP(I,J)/0.4536
             ITSN(I,J) = ITSN(I,J)/0.4536
  100     CONTINUE
C
C .
C
C  Find Exterior Bending Moment
        WRITE(13,*)
        WRITE(13,*)
       .WRITE(13,*) 'EXTERIOR   BENDING  MOMENT (TON-M)'
      DO 101 I=1,NUMSPA
        WRITE(13,*) 'NO. SPAN : ',I
       DO 101 J=1,NSC(I)
       EMP(I,J)=EEMP(I,J)
       IF(ETMP(I,J).GT.EEMP(I,J)) EMP(I,J)=ETMP(I,J)
       EMN(I,J)=EEMN(I,J)
       IF(ETMN(I,J).LT.EEMN(I,J)) EMN(I,J)=ETMN(I,J)
       EMP(I,J) = EMP(I,J)*0.1383
       EMN(I,J) = EMN(I,J)*0.1383
       WRITE(13,11)  EMP(I,J),EMN(I,J)
  11    FORMAT(5X,F10.4,3X,F10.4)
  101    CONTINUE
C
C
C
C  Find Exterior Shear Force
        WRITE(13,*)
        WRITE(13,*)
        WRITE(13,*)'EXTERIOR   SHEAR   FORCE (TON)'
       DO 102 I=1,NUMSPA
         WRITE(13,*) 'NO. SPAN : ',I
       DO 102 J=1,NSC(I)-1
       SF1=ABS(EESP(I,J))
       SF2=ABS(EESN(I,J))
       SF3=ABS(ETSP(I,J))
       SF4=ABS(ETSN(I,J))
       ESF(I,J)=SF1
       IF(ESF(I,J).LT.SF2) ESF(I,J)=SF2
       IF(ESF(I,J).LT.SF3) ESF(I,J)=SF3
       IF(ESF(I,J).LT.SF4) ESF(I,J)=SF4
       ESF(I,J) = ESF(I,J)*0.4536
       WRITE(13,12)   ESF(I,J)
  12    FORMAT(5X,F10.4)
  102  CONTINUE
          DO 106     I=1,NUMSPA-1
          SE(I+1)     =ESF(I,NSC(I)-1)+ESF(I+1,1)
  106     CONTINUE
          SE(1)        =ESF(1,1)
          SE(NUMSPA+1)=ESF(NUMSPA,NSC(I)-1)
          DO 1006     I=1,NUMSPA+1
          REACE(I)     =SE(I)*2
  1006    CONTINUE
```

```
C
C
C  Find interior bending moment
       DO 103 I=1,NUMSPA
        DO 103J=1,NSC(I)
        IMP(I,J) = IEMP(I,J)
        IF(ITMP(I,J).GT.IEMP(I,J)) IMP(I,J)=ITMP(I,J)
        IMN(I,J) = IEMN(I,J)
        IF(ITMN(I,J).LT.IEMN(I,J)) IMN(I,J)=ITMN(I,J)
        IMP(I,J) = IMP(I,J)*0.1383
        IMN(I,J) = IMN(I,J)*0.1383
 103     CONTINUE
C
C  Find interoir shear force
       DO 104 I=1,NUMSPA
       DO 104 J=1,NSC(I)-1
       SF1 = ABS(IESP(I,J))
       SF2 = ABS(IESN(I,J))
       SF3 = ABS(ITSP(I,J))
       SF4 = ABS(ITSN(I,J))
        ISF(I,J)=SF1
       IF(ISF(I,J).LT.SF2) ISF(I,J)=SF2
       IF(ISF(I,J).LT.SF3) ISF(I,J)=SF3
       IF(ISF(I,J).LT.SF4) ISF(I,J)=SF4
        ISF(I,J) = ISF(I,J)*0.4536
 104    CONTINUE
C
        DO 105      I=1,NUMSPA-1
        SI(I+1)      =ISF(I,NSC(I)-1)+ISF(I+1,1)
105     CONTINUE
        SI(1)        =ISF(1,1)
        SI(NUMSPA+1) =ISF(NUMSPA,NSC(I)-1)
        DO 1005     I=1,NUMSPA+1
        REAC(I)      =SI(I)*(NOBEAM-2)
1005    CONTINUE
 C
 C
 C
      DO 470 I=1,NUMSPA
        TH =7.0/12
       GDSPC=(EGDAP-BWIDTH)/(NOBEAM-1)
        B1 = SL(I)/4
        B2 = 16*TH + BWIDTH
        B3 =0.5*GDSPC + BWIDTH
        BB = B1
        IF (B2.LT.BB)  BB=B2
        IF (B3.LT.BB)  BB=B3
  470   CONTINUE
        AA =0.3048* BB
        BWIDT=0.3048*BWIDTH
        T=0.3048*TH
        CALL EDESIGN(BB)
        CALL TDESIGN(BB)
        CALL QUANCON1
        CALL QUANCON2
        CALL QUANFORM1
        CALL QUANFORM2
        CALL QUANSTEEL1
        CALL QUANSTEEL2
         RETURN
         END
                    ------------------------------
                      SUBROUTINE  EDESIGN(BB)
```

```
C                *-----------------------------------*
C
C
      REAL IMP,IMN,ISF
      COMMON /MATER/ FCP,FSY
      COMMON /BRGDIA/ NUMTRA,WLANE,EQDAP,NOBEAM,CSL,PAREA,NUMSPA
      COMMON /PBEAM/ BWIDTH,SINTER
      COMMON /BEAM/ TYP(9),SL(9),NSC(9),SL1(9),SL2(9),D0(9),D1(9),D2(9)
      COMMON /STSIZE/ NOSB,NOST,NOSL
      COMMON /BDEPTH/ BDEPTH
      COMMON /CDATA/ FC,FS,MR,CK,CJ,CRR,VMAX,VCON1,VCON2,SDSPC
      COMMON /MOMENT/ EMP(9,50),EMN(9,50),IMP(9,50),IMN(9,50)
      COMMON /SFORCE/ ESF(9,50),ISF(9,50)
      COMMON /DIST/ X(9,50),DX(9,50)
      COMMON /STEELE/ ASTE(9,50),ASBE(9,50),ASDE(9,50)
      COMMON /STESEC/ CR,ST(12,2),FORM(12,2)
C
C
      WRITE (13,*)
      WRITE (13,*)
      WRITE (13,*)' ********************'
      WRITE (13,*)'   EXTERIOR DESIGN   '
      WRITE (13,*)' ********************'
C
C  For Positive Moment
      DD = 5.0/12
      TH = 7.0/12
      DO 500 I=1,NUMSPA
      DO 500 J=1,NSC(I)
         DP = DX(I,J)+TH-DD
         AS = EMP(I,J)*1000*12/(FS*(DP*12-TH*12/2))
         P  = AS/(BB*DP*144)
         AK = (MR*P + 0.5*(TH/DP)**2)/(MR*P + TH/DP)
         Y  = (TH*12*(3*AK*DP*12 - 2*TH*12))/(3*(2*AK*DP*12-TH*12))
       AJD = DP*12 - Y
       AMC = (1-(TH*12/2*AK*DP*12))*FC*BB*12*TH*12*AJD
       ASBE(I,J) = EMP(I,J)*1000*12/(FS*AJD)
  500  CONTINUE
C
C
C  For Negative Moment
      TH =7.0/12
      DD=5.0/12
      DDD=5.0/12
      DO 55 I=1,NUMSPA
      DO 65 J=1,NSC(I)
        DN = DX(I,J)+TH-DDD
        AM1 = 0.5*FC*CK*CJ*BWIDTH*12*DN*DN*144*(-1)/(1000*12)
        IF (AM1.LT.EMN(I,J)) THEN
        ASTE(I,J) = EMN(I,J)*(-1)*1000*12/(FS*CJ*DN*12)
        ASDE(I,J) =0.000
        ELSE
        AS1 =(-1)*AM1*1000*12/(FS*CJ*DN*12)
        AM2 =(EMN(I,J)*(-1)*1000*12)-(AM1*(-1)*1000*12)
        AS2 = AM2/(FS*(DN*12-DD*12))
        ASTE(I,J) = AS1 + AS2
        ASDE(I,J) = 0.5*AS2*(1-CK)/(CK-DD/DN)
        ENDIF
  65  CONTINUE
  55  CONTINUE
```

```
C  Find bottom steel
      DO 700 I=1,NUMSPA
      DO 800 J=1,NSC(I)
C
      IF(ASDE(I,J).GT.ASBE(I,J)) ASBE(I,J)=ASDE(I,J)
  800 CONTINUE
  700 CONTINUE
C
C
      VCC = 0.95*FCP**0.5
      ASE  = ST(NOST,2)
C
      DO 90 I=1,NUMSPA
         WRITE(13,900)
         WRITE(13,1000) I
         WRITE(13,1500)
         WRITE(13,1100)
         WRITE(13,1110) NOSB,NOSB,NOST
         WRITE(13,1500)
         WRITE(13,*)
      DO 95 J=1,NSC(I)
         NAST = ASTE(I,J)/ST(NOSB,2) + 1
         NASB = ASBE(I,J)/ST(NOSB,2) + 1
         IF(NAST.LT.2) NAST = 2
         IF(NASB.LT.2) NASB =2
C
         DV = DX(I,J)+TH-DD
         VC = VCC*BWIDTH*DV*144
         V  = ESF(I,J)*1000-VC
         IF(V.LT.0)  THEN
         S =(DV/2)*0.3048
         ELSE
         S =(2*ASE*FS*DV*12/V)*0.0254
         ENDIF
         IF (S.GT.0.15)  S=0.15
         XX = X(I,J)*0.3048
         DDX = DX(I,J)*0.3048
         WRITE(13,1200) J,XX,DDX,NAST,NASB
         IF (J.EQ.NSC(I)) GOTO 95
         WRITE(13,1300) S
   95 CONTINUE
         WRITE(13,*)
   90 CONTINUE
C
C
  900 FORMAT(//)
 1000 FORMAT(1x,'Span # ',I5/)
 1100 FORMAT(' Section    Distance   Depth     Top Bar     Bottom Bar
     +     Stirrub ')
 1110 FORMAT('    No.          (m.)       (m.)       #',I3,7x,'#',I3,10x,
     +' #',I3,' (m.)'/)
 1200 FORMAT(1x,I5,8x,F5.2,5x,F5.2,5x,I6,5x,I6)
 1300 FORMAT(62x,F7.2)
 1500 FORMAT('-------------------------------------------------------
     +----------------')
C
C
C
```

```
      RETURN
      END
C
C
C
C                    *-------------------------------------*
C                           SUBROUTINE  IDESIGN(BB)
C                    *-------------------------------------*
C
C
      REAL IMP, IMN, ISF
      COMMON /MATER/ FCP, FSY
      COMMON /STSIZE/ NOSB, NOST, NOSL
      COMMON /BRGDIA/ NUMTRA, WLANE, EGDAP, NOBEAM, CSL, PAREA, NUMSPA
      COMMON /PBEAM/ BWIDTH, SINTER
      COMMON /BEAM/ TYP(9), SL(9), NSC(9), SL1(9), SL2(9), D0(9), D1(9), D2(9)
C
      COMMON /BDEPTH/ BDEPTH
      COMMON /CDATA/ FC, FS, MR, CK, CJ, CRR, VMAX, VCON1, VCON2, GDSPC
C
      COMMON /MOMENT/ EMP(9, 50), EMN(9, 50), IMP(9, 50), IMN(9, 50)
      COMMON /SFORCE/ ESF(9, 50), ISF(9, 50)
      COMMON /DIST/ X(9, 50), DX(9, 50)
      COMMON /STEEI/ ASTI(9, 50), ASBI(9, 50), ASDI(9, 50)
C
      COMMON /STESEC/ CR, ST(12, 2), FORM(12, 2)
C
C
C
      WRITE (13, *)
      WRITE (13, *)
      WRITE (13, *)
      WRITE (13, *)' ********************'
      WRITE (13, *)'    INTERIOR DESIGN   '
      WRITE (13, *)' ********************'
C
C  For Positive Moment
      DD = 5.0/12
      TH = 7.0/12
      DO 500 I=1, NUMSPA
      DO 500 J=1, NSC(I)
        DP = DX(I, J)+TH-DD
        AS = IMP(I, J)*1000*12/(FS*(DP*12-TH*12/2))
        P  = AS/(BB*DP*144)
        AK = (MR*P + 0.5*(TH/DP)**2)/(MR*P + TH/DP)
        Y  = (TH*12*(3*AK*DP*12 - 2*TH*12))/(3*(2*AK*DP*12-TH*12))
        AJD = DP*12 - Y
        AMC = (1-(12*TH/2*AK*DP*12))*FC*BB*12*TH*12*AJD
        ASBI(I, J) = IMP(I, J)*1000*12/(FS*AJD)
  500   CONTINUE
C
C
C  For Negative Moment
      TH =7.0/12
       DD=5.0/12
      DDD=5.0/12
      DO 55 I=1, NUMSPA
      DO 65 J=1, NSC(I)
        DN = DX(I, J)+TH-DDD
        AM1 = 0.5*FC*CK*CJ*BWIDTH*12*DN*DN*144*(-1)/(1000*12)
```

```
      IF (AM1.LT.IMN(I,J)) THEN
      ASTI(I,J)=IMN(I,J)*(-1)*1000*12/(FS*CJ*DN*12)
      ASDI(I,J)=0.000
      ELSE
      AS1 = (-1)*AM1*1000*12/(FS*CJ*DN*12)
      AM2 = IMN(I,J)*(-1)*1000*12-(AM1*(-1)*1000*12)
      AS2 = AM2/(FS*(DN*12-DD*12))
      ASTI(I,J) = AS1 + AS2
      ASDI(I,J) = 0.5*AS2*(1-CK)/(CK-DD/DN)
      ENDIF
   65 CONTINUE
   55 CONTINUE
C
C
C  Find bottom steel
      DO 700 I=1,NUMSPA
      DO 900 J=1,NSC(I)
         IF(ASDI(I,J).GT.ASBI(I,J)) ASBI(I,J)=ASDI(I,J)
  900 CONTINUE
  700 CONTINUE
C
C
      VCC = 0.85*FCP**0.5
      ASI  = ST(NOST,2)
C
      DO 90 I=1,NUMSPA
         WRITE(13,700)
         WRITE(13,1000) I
         WRITE(13,1500)
         WRITE(13,1100)
         WRITE(13,1110) NOSB,NOSB,NOST
         WRITE(13,1500)
         WRITE(13,*)
      DO 95 J=1,NSC(I)
         NAST = ASTI(I,J)/ST(NOSB,2) + 1
         NASB = ASBI(I,J)/ST(NOSB,2) + 1
         IF (NAST.LT.2) NAST=2
         IF (NASB.LT.2) NASB=2
         DV = DX(I,J)+TH-DD
         VC = VCC*BWIDTH*DV*144
         V  = ISF(I,J)*1000-VC
         IF (V.LT.0) THEN
         S  = (DV/2)*0.3048
         ELSE
         S  =(2*ASI*FS*D*12/V)*0.0254
         ENDIF
         IF(S.GT.0.15) S=0.15
         XX  = X(I,J)*0.3048
         DDX = DX(I,J)*0.3048
         WRITE(13,1200) J,XX,DDX,NAST,NASB
         IF (J.EQ.NSC(I)) GOTO 95
         WRITE(13,1300) S
   95 CONTINUE
         WRITE(13,*)
   90 CONTINUE
C
C
  900 FORMAT(//)
 1000 FORMAT(1x,'Span # ',I5/)
 1100 FORMAT(' Section    Distance   Depth     Top Bar    Bottom Bar
```

```
     +     Stirrub ')
 1110 FORMAT('   No.          (m.)       (m.)       #', I3, 7x, '#', I3, 10x,
     +'  #', I3, ' (m.)'/)
 1200 FORMAT(1x, I5, 8x, F5.2, 5x, F5.2, 5x, I6, 5x, I6)
 1300 FORMAT(62x, F7.2)
 1500 FORMAT('-------------------------------------------------------------
     +-------------')
C
C
C
      RETURN
      END
C
C
C
C
```

```
C-------------------------------------------------------------------
C  ***************************************************************
C  *                    QUANTITY   OF   CONCRETE                 *
C  ***************************************************************
C
C-----------------------------
      SUBROUTINE  QUANCON1
C-----------------------------
C
C   * Find   Quantity   of   Concrete  Slab *
C
      COMMON /BRGDIA/ NUMTRA, WLANE, EGDAP, NOBEAM, CSL, PAREA, NUMSPA
      COMMON /PBEAM/ BWIDTH, SINTER
      COMMON /BEAM/ TYP(9), SL(9), NSC(9), SL1(9), SL2(9)
     * , D0(9), D1(9), D2(9)
      COMMON /DIST/ X(9,50), DX(9,50)
C
      COMMON /QCON/ TQCS, TQCG, TQC
C
      TH  = 7.00/12
      SLL  = 0.00
      DO 10 I =1, NUMSPA
      SLBL = SL(I) + SLL
 10   CONTINUE
      SLBW = EGDAP + CSL*2.00
      TQCS = (SLBL*SLBW*TH)*0.3048**3
C
C   * Find   Quantity   of   Concrete  Girder *
C
      GBM     = 0.00
      DO 20 I = 1, NUMSPA
      DO 30 J = 1, NSC(I)-1
            N = J+1
      GBM     = 0.5*(DX(I,J)+DX(I,N))*(X(I,N)-X(I,J))*BWIDTH+GBM
 30   CONTINUE
 20   CONTINUE
      GBM     = GBM*NOBEAM
      TQCG    = GBM*0.3048**3
      TQC     = TQCS+TQCG
      RETURN
      END
C-------------------------------------------------------------------
C-----------------------------
      SUBROUTINE QUANCON2
C-----------------------------
C * Quantity of Concrete Slab & Girder
C
      COMMON /QCON/ TQCS, TQCG, TQC
C
      WRITE(13,600)
      WRITE(13,100)
      WRITE(13,500)
      WRITE(13,200)
      WRITE(13,500)
      WRITE(13,100)
      WRITE(13,300)
      WRITE(13,450)
      WRITE(13,100)
      WRITE(13,400) TQCS, TQCG, TQC
      WRITE(13,100)
```

```
600     FORMAT(//////)
500     FORMAT(/)
100     FORMAT(15X,'----------
    *----------')
200     FORMAT(T36,'QUANTITY  OF    CONCRETE (cu.m)')
300     FORMAT(T20,'Slab',T40,'Girder',T60,'Total')
400     FORMAT(T18,F6.2,T38,F8.2,T62,F8.2)
450     FORMAT(T20,'(cu.m)',T40,'(cu.m)',T65,'(cu.m)')
        RETURN
        END
C
C-------------------------------------------------------------------
C
C  ***************************************************************
C  *                    QUANTITY    OF    FORMWORK                 *
C  ***************************************************************
C
C-----------------------------
      SUBROUTINE  QUANFORM1
C-----------------------------
C
C   * Find   Quantity   of    Formwork   Slab *
C
      COMMON /BRGDIA/ NUMTRA,WLANE,EGDAP,NOBEAM,CSL,PAREA,NUMSPA
      COMMON /PBEAM/ BWIDTH,SINTER
      COMMON/BEAM/TYP(9),SL(9),NSC(9),NJ(9),SL2(9),DO(9),D1(9),D2(9)
      COMMON /DIST/ X(9,50),DX(9,50)
C
      COMMON /QFORM/ TQFS,TQFG,TQF
C
       TH   = 7.00/12
       SLBL  = 0.00
       DO 10 I =1,NUMSPA
       SLBL  = SL(I) + SLBL
10     CONTINUE
       SLBW  = EGDAP + CSL*2.00
       SLBW1 = SLBW-NOBEAM*BWIDTH
       TQFS  = (SLBW1*SLBL+TH*SLBL*2)*0.3048**2
C
C   * Find   Quantity   of   Formwork  Girder *
C
       GFMG     = 0.00
          D     = 0.00
       DO 20 I = 1,NUMSPA
       DO 30 J = 1,NSC(I)-1
             N = J+1
       GFMG     = 0.5*(DX(I,J)+DX(I,N))*(X(I,N)-X(I,J))+GFMG
          A     = DX(I,J)-DX(I,N)
          B     = X(I,N)-X(I,J)
          C     = A**2+B**2
          D     = C**0.5+D
30     CONTINUE
20     CONTINUE
       GFMG1    = D*BWIDTH*NOBEAM
       GFMG2    = GFMG*NOBEAM*2
       TQFG     = (GFMG1+GFMG2)*0.3048**2
       TQF      = (TQFG+TQFS)
       RETURN
       END
C-------------------------------------------------------------------
```

```
C--------------------------------
      SUBROUTINE QUANFORM2
C--------------------------------
C * Quantity of Formwork Slab & Girder
C
      COMMON /QFORM/ TQFS,TQFG,TQF
C
      WRITE(13,600)
      WRITE(13,100)
      WRITE(13,500)
      WRITE(13,200)
      WRITE(13,500)
      WRITE(13,100)
      WRITE(13,300)
      WRITE(13,450)
      WRITE(13,100)
      WRITE(13,400) TQFS,TQFG,TQF
      WRITE(13,100)
  600 FORMAT(//////)
  500 FORMAT(/)
  100 FORMAT(15X,'-------------------------------------------------
     *----------')
  200 FORMAT(T36,'QUANTITY OF FORMWORK (sq.m)')
  300 FORMAT(T20,'Slab',T40,'Girder',T65,'Total')
  400 FORMAT(T18,F6.2,T38,F8.2,T62,F8.2)
  450 FORMAT(T20,'(sq.m)',T40,'(sq.m)',T65,'(sq.m)')
      RETURN
      END
C
C--------------------------------------------------------------------
C
C  ********************************************************************
C  *                    QUANTITY OF STEEL                             *
C  ********************************************************************
C
C-----------------------------
      SUBROUTINE QUANSTEEL1
C-----------------------------
C
C   * Find Quantity of Steel Slab *
C
      COMMON /BRGDIA/ NUMTRA,WLANE,BRDIA,NOBEAM,CSL,PAREA,NUMSPA
      COMMON /PBEAM/ BWIDTH,SINTER
      COMMON /BEAM/ TYP(9),SL(9),NOC(9),SL1(9),SL2(9)
     *  ,D0(9),D1(9),D2(9)
      COMMON /DIST/ X(9,50),DX(9,50)
C
      COMMON /NSTEEL/ NASTE(9,50),NASBE(9,50),NASTI(9,50),NASBI(9,50)
      COMMON /STESEC/ CR,ST(11,2),STRN(11,2)
      COMMON /STSIZE/ NOSB,NOST,NOSI
      COMMON /SPACS/ STRE(9,50),STRI(9,50)
      COMMON /SPACSL/ STT,STL,STTT
C
      COMMON /GSTEEL/ TQSS,TQSG,TQS
C
      TH   =27.00/12
      SLL  = 0.00
      DO 10 I =1,NUMSPA
      SLBL = SL(I) + SLL
   10 CONTINUE
```

```
      SLBW = EGDAP + CSL*2.00
C   * Transverse  steel
      NSTT = SLBL/STT+1
      LSTT = NSTT*SLBW
      WT   = ST(NOSL,2)*5.0812
      QST  = (LSTT*0.3048)*WT/1000
C   * Longitudinal Steel
      NOSL = NOSL-1
      NSTL = SLBW/STL+1
      LSTL = NSTL*SLBL
      WT   = ST(NOSL,2)*5.0812
      QSL  = (LSTL*0.3048)*WT/1000
C   *Temperture  Steel
      NSTTT= SLBW/STTT+1
      LSTTT= NSTTT*SLBL
      WT   = ST(NOSL,2)*5.0812
      QSTT = (LSTTT*0.3048)*WT/1000
C
      TQSS = QST+QSL+QSTT
C
C    * Find   Quantity   of   Steel  Girder *
C
C    * Exterior girder & Interior girder
C    *  Top Steel & Bottom Steel
C
      QSTTT2  = 0.00
      QSTTT22 = 0.00
      QSTTT4  = 0.00
      QSTTT44 = 0.00
      QSTTB2  = 0.00
      QSTTB22 = 0.00
      QSTTB4  = 0.00
      QSTTB44 = 0.00
      DO 20 I = 1,NUMSPA
      DO 30 J = 1,NSC(I)
      IF(J.EQ.1.OR.J.EQ.NSC(I)) THEN
      QSTTT1  = NASTE(I,J)*ST(NOSB,?)
      QSTTB1  = NASBE(I,J)*ST(NOSB,?)
      QSTTT11 = NASTI(I,J)*ST(NOSB,?)
      QSTTB11 = NASBI(I,J)*ST(NOSB,?)
      QSTTT2  = (QSTTT1*5.0812)*SIN(?R/?.00*0.3048+QSTTT2
      QSTTB2  = (QSTTB1*5.0812)*SIN(?R/?.00*0.3048+QSTTB2
      QSTTT22 = (QSTTT11*5.0812)*SIN(?R/2.00*0.3048+QSTTT22
      QSTTB22 = (QSTTB11*5.0812)*SIN(?R/2.00*0.3048+QSTTB22
        ELSE
      QSTTT3  = NASTE(I,J)*ST(NOSB,?)
      QSTTB3  = NASBE(I,J)*ST(NOSB,?)
      QSTTT33 = NASTI(I,J)*ST(NOSB,?)
      QSTTB33 = NASBI(I,J)*ST(NOSB,?)
      QSTTT4  = (QSTTT3*5.0812)*SIN(?R*0.3048+QSTTT4
      QSTTB4  = (QSTTB4*5.0812)*SIN(?R*0.3048+QSTTB4
      QSTTT44 = (QSTTT33*5.0812)*SIN(?R*0.3048+QSTTT44
      QSTTB44 = (QSTTB33*5.0812)*SIN(?R*0.3048+QSTTB44
        ENDIF
 30   CONTINUE
 20   CONTINUE
      QSG1    = (QSTTT2+QSTTB2+QSTTT4+QSTTB4)*2
      QSG2    = (QSTTT22+QSTTB22+QSTTT44+QSTTB44)*(NOBEAM-2)
C
      QSG3    = (QSG1+QSG2)/1000
```

```
C
C  * Stirrup   Steel
C
        NOSTRE  = 0
        NOSTRI  = 0
            C   = 0.00
        DO 40 I = 1,NUMSPA
        DO 50 J = 1,NSC(I)
            N   =J+1
        A       = ((DX(I,J)+DX(I,N)/2)+TH-2.00*(4.00/2.54/12))*2
        B       = ((BWIDTH-(4.00/2.54/12.00)*2))*2
        C       = A+B+C
        NOSTRE  = SINTER/(STRE(I,J)*0.280)
        NOSTRI  = SINTER/(STRI(I,J)*0.280)
        NOSTRE  = NOSTRE+NOSTRE
        NOSTRI  = NOSTRI+NOSTRI
50      CONTINUE
40      CONTINUE
        NOSTR   = NOSTRE*2+NOSTRI*(NOBEAM-2)
        QST1    = NOSTR*C*0.3048
C
        QST2    = (ST(NOST,2)*5.0812)*QST1/1000
C
C
        TQSG    = QSG3+QST2
C
C
        TQS     = TQSG+TQSS
C
        RETURN
        END
C-------------------------------------------------------------------------
C--------------------------------
        SUBROUTINE QUANSTEEL2
C--------------------------------
C  * Qauntity   of  Steel
C
        COMMON /QSTEEL/ TQSS,TQSG,TQS
C
        WRITE(13,600)
        WRITE(13,100)
        WRITE(13,500)
        WRITE(13,200)
        WRITE(13,500)
        WRITE(13,100)
        WRITE(13,300)
        WRITE(13,450)
        WRITE(13,100)
        WRITE(13,400) TQSS,TQSG,TQS
        WRITE(13,100)
 600    FORMAT(//////)
 500    FORMAT(/)
 100    FORMAT(15X,'----------
     *----------')
 200    FORMAT(T36,'QUANTITY  OF   STEEL (ton)')
 300    FORMAT(T20,'Slab',T40,'Girder',T60,'Total')
 400    FORMAT(T18,F6.2,T38,F8.2,T63,F8.2)
 450    FORMAT(T20,'(ton)',T41,'(ton)',T60,'(ton)')
        RETURN
        END
```

```
C
C---------------------------
      SUBROUTINE DEFL1
C ---------------------------
C
C
C
C
C
C
C
      INTEGER REACT
      COMMON /ADD/ FAC,NCB,E
      COMMON/ADD1/SE(10),SI(10),REACE(10),REACI(10),REACT(10),DEFL(9)
      COMMON /ADD2/SM(9),S(9)
      COMMON/BEAM/TYP(9),SL(9),NSC(9),SL1(9),SL2(9),DO(9),DI(9),D2(9)
      COMMON/PBEAM/BWIDTH,SINTER
      COMMON/YY4/ AAA(100,12),BBB(100,12)
      COMMON /LOAD/DD1,DE1,DI1,DLL2,PLM,PLS
      COMMON/BRGDIA/NUMTRA,WLANE,EQDAP,NOBEAM,CSL,PAREA,NUMSPA,RR,CB
C
C
        E    =  E*144
        DE1= DE1*1000/SINTER
        K1 = NSC(1)-1
        K2 = K1+NSC(2)-1
        K3 = K2+NSC(3)-1
        K4 = K3+NSC(4)-1
        K5 = K4+NSC(5)-1
        K6 = K5+NSC(6)-1
        K7 = K6+NSC(7)-1
        K8 = K7+NSC(8)-1
        K9 = K8+NSC(9)-1
         K  = 0
        DO 10 I  = 1,NUMSPA
        K  = K+NSC(I)-1
10      CONTINUE
        DO 20 I  = 1,K
        YI1      = 0.00
        IF(I.EQ.K1) YI2      =ABS(AAA(K1,1))*1000
                    YII1     = YI2
        IF(I.EQ.K2) YII2     =ABS(AAA(K2,1))*1000
                    YIII1    = YII2
        IF(I.EQ.K3) YIII2    =ABS(AAA(K3,1))*1000
                    YIY1     = YIII2
        IF(I.EQ.K4) YIY2     =ABS(AAA(K4,1))*1000
                    YY1      = YIY2
        IF(I.EQ.K5) YY2      =ABS(AAA(K5,1))*1000
                    YYI1     = YY2
        IF(I.EQ.K6) YYI2     =ABS(AAA(K6,1))*1000
                    YYIII    = YYI2
        IF(I.EQ.K7) YYII2    =ABS(AAA(K7,1))*1000
                    YYIII1   = YYII2
        IF(I.EQ.K8) YYIII2   =ABS(AAA(K8,1))*1000
                    YIX1     = YYIII2
        IF(I.EQ.K9) YIX2     =ABS(AAA(K9,1))*1000
20      CONTINUE
C
        DO 30 I=1,NUMSPA
        SM(I)   = DE1*SL(I)**2/8
         S1= BWIDTH*DI(I)**3/12
         S2= BWIDTH*DO(I)**3/12
         S3= BWIDTH*D2(I)**3/12
         S(I) =(S1+S2+S3)/3
30      CONTINUE
         DO 40 I=1,NUMSPA
         IF (I.EQ.NUMSPA) GOTO 50
     IF(I.EQ.1)DEFL(I)=SL(I)**2*(5*SM(I)-3*YI2)
    +                  /(48*E*S(I))
```

```
C
      IF(I.EQ.2)DEFL(I)=      **2*(5*SM(I)-3*(YII1+YII2))
     +                        E*S(I))
      IF(I.EQ.3)DEFL(I)=      **2*(5*SM(I)-3*(YIII1+YII2))
     +                        E*S(I))
      IF(I.EQ.4)DEFL(I)=     )**2*(5*SM(I)-3*(YIY1+YY2))
     +                        E*S(I))
      IF(I.EQ.5)DEFL(I)=     )**2*(5*SM(I)-3*(YY1+YY2))
     +                        E*S(I))
      IF(I.EQ.6)DEFL(I)=      **2*(5*SM(I)-3*(YYI1+YYI2))
     +                        E*S(I))
      IF(I.EQ.7)DEFL(I)=     )**2*(5*SM(I)-3*(YYII1+YYII2))
     +                        E*S(I))
      IF(I.EQ.8)DEFL(I)=     )**2*(5*SM(I)-3*(YYIII1+YYIII2))
     +                        E*S(I))
      IF(I.EQ.9)DEFL(I)=     )**2*(5*SM(I)-3*(YIX1+YIX2))
     +                        E*S(I))
40        CONTINUE
50        DEFL(I) = DEFL(
          RETURN
          END
C
C
C-------------------------  ----------------------------------------
C
C-------------------------  --------
          SUBROUTINE DEFL2
C-------------------------  --------
C
          INTEGER REACT
          COMMON /ADD/ FAC,
          COMMON/ADD1/SE(10)      0),REACE(10),REACI(10),REACT(10),DEFL(9)
          COMMON /ADD2/SM(9)
          COMMON/BEAM/TYP(9)      9),NSC(9),SL1(9),SL2(9),DO(9),D1(9),D2(9)
          COMMON/PBEAM/BWIDT      TER
          COMMON/YY4/ AAA(10      ),BBB(100,12)
          COMMON /LOAD/DD1,D      I,DLL2,PLM,PLS
          COMMON/BRGDIA/NUMT      LANE,EGDAP,NOBEAM,CSL,PAREA,NUMSPA,RR,CB
C
C
            WRITE(13,600)
            WRITE(13,100)
            WRITE(13,500)
            WRITE(13,300)
            WRITE(13,500)
            WRITE(13,100)
            WRITE(13,500)
            DO 1 I=1,NUMSPA
            WRITE(13,1000) I      (I)*12*2.54
1           CONTINUE
            WRITE(13,500)
            WRITE(13,100)
600         FORMAT(//////)
100         FORMAT(15X,'-----  ----------------------------------------')
500         FORMAT(/)
300         FORMAT(T20,'DEFL      ON EACH SPAN (MIDDLE SPAN)','(CM.)')
1000        FORMAT(T16,'DEF.        SPAN  ',I1,2X,'= ',F10.3,'  CM.')

            WRITE(13,600)
            WRITE(13,100)
            WRITE(13,500)
```

```
C
          WRITE(13,301)
          WRITE(13,500)
          WRITE(13,100)
          WRITE(13,500)
           DO 20 I=1,NUMSPA+1
          WRITE(13,1001) I,REACT(I)
20         CONTINUE
          WRITE(13,500)
          WRITE(13,100)
301       FORMAT(T20,'REATION AT SUPPORT (TON.)')
1001      FORMAT(T16,'REACTION AT SUPPORT NO. ',I1,'=  ',I8,' TON.')
          RETURN
          END
C
C-------------------------------------------------------------------
C
```

# ภาคผนวก ข.

## การใช้โปรแกรมคอมพิวเตอร์

การใช้โปรแกรมคอมพิวเตอร์

ในการใช้โปรแกรมคอมพิวเตอร์ของงานวิจัยนี้ วิธีการใช้ง่ายและสะดวก การใช้จะเป็นการเลือกหัวข้อซึ่งเป็นหมายเลข ผู้ใช้โปรแกรมจะเป็นผู้เลือกหัวข้อโดยการพิมพ์ตัวเลขหัวข้อ หลังจากนั้นกด RETURN KEY บนแป้นพิมพ์ ผู้ใช้ก็สามารถจะดำเนินการอย่างใดอย่างหนึ่งได้ตามหัวข้อที่ตนเองได้เลือกไว้นั้น

เมื่อเข้าสู่ตัวโปรแกรมเริ่มต้นจะขึ้นด้วยรายการหลัก (Main Menu) ให้ผู้ใช้ทำการเลือกโดยมีหัวข้อใหญ่ 3 หัวข้อคือ

1) EDIT DATA MENU

2) PROCESS CALCULATION :

   Then output are on file OUT 1. TXT

   and OUT 2. TXT

   and OUT 3. TXT

3) QUIT TO PRIME OPERATING SYSTEM

   SELECT NUMBER (1, 2, 3)

เมื่อผู้ใช้โปรแกรมเลือกหัวข้อ 1 ซึ่งเป็นหัวข้อของการใส่ข้อมูล จะขึ้นรายละเอียดในหัวข้อที่ 1 ให้เลือกสำหรับการใส่ข้อมูลอีก 7 หัวข้อคือ

1) Edit Title Heading

2) Edit Material Properties

3) Edit Load Class

4) Edit Bridge Dimensions

5) Edit Beam Dimensions

6) Edit Bar Size

7) Edit Settlement Size

8) <<< Return to main menu>>>

   SELECT NUMBER (1,2,.....,8)

ผู้ใช้โปรแกรมจะเป็นผู้เลือกการใส่ข้อมูลที่ละหัวข้อ หลังจากใส่ข้อมูลครบทุกหัวข้อแล้วก็ให้ทำการเลือกหัวข้อที่ 8 ก็จะกลับคืนสู่รายการหลัก (Main Menu) เริ่มแรกอีกครั้งหนึ่ง หลัง

จากนี้ให้ผู้ใช้เลือกหัวข้อที่ 2 ในรายการหลัก (Main Menu) ซึ่งคือส่วนการวิเคราะห์และการออกแบบ ก็จะมีรายละเอียดให้เลือกอีก 3 หัวข้อคือ

1) SLAB DESIGN

2) BEAM ANALYSIS

3) BEAM DESIGN

4) <<<RETURN TO MAIN MENU>>>

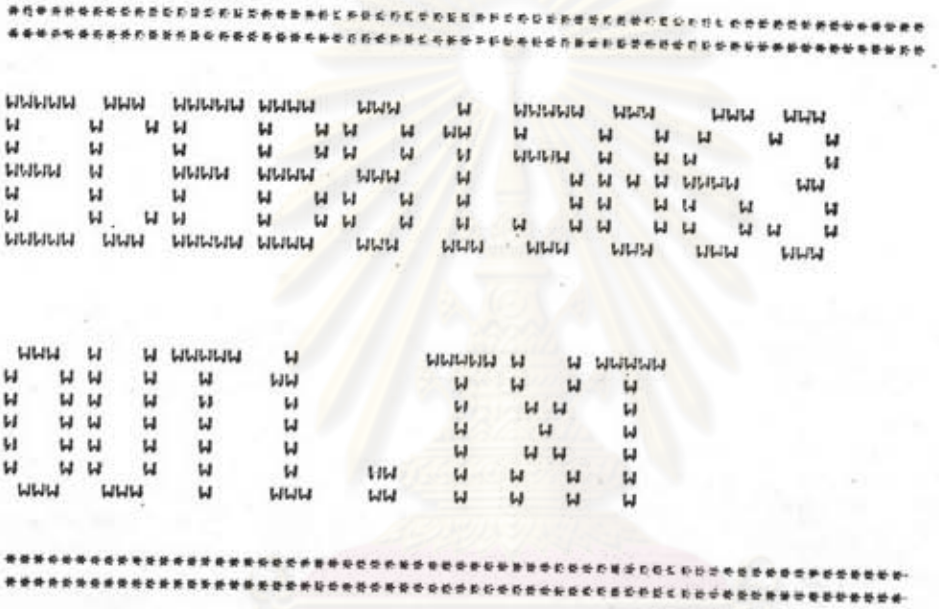
SELECT NUMBER (1, 2, 3, 4)

หลังจากผู้ใช้โปรแกรมได้ทำการเลือกหัวข้อ 1, 2, 3 ซึ่งเป็นการวิเคราะห์และการออกแบบเสร็จเรียบร้อยแล้ว และต้องการกลับสู่ รายการหลัก (Main menu) ก็ให้เลือกหัวข้อที่ 4 สำหรับข้อมูลที่ได้ทำการป้อนในครั้งแรกจะถูกเก็บไว้ในไฟล์ชื่อ OUT 1.TXT ส่วนผลของการวิเคราะห์และการออกแบบจะเก็บไว้ในไฟล์ชื่อ OUT 2. TXT และ OUT 3. TXT และเมื่อผู้ใช้โปรแกรมเลือกหัวข้อ 3 ในรายการหลัก (Main Menu) ก็จะเป็นการสิ้นสุดของการใช้โปรแกรม

ในการออกแบบทุกครั้งผู้ใช้โปรแกรมสามารถจะทำการเปลี่ยนแปลงข้อมูลได้ทุกครั้งโดยการป้อนข้อมูลตัวใหม่ได้

# ภาคผนวก ค.

## การคำนวณจากโปรแกรมคอมพิวเตอร์



```
**********************************************
*********** TITLE HEADING LIST ***************
**********************************************

      An Existing Title is shown below

  CONTINOUS   R.C.   BEAM   FOR   BRIDGE   DESIGN

-----------------------------------------------
```

```
**********************************************
***********    MATERIAL PROPERTIES    ********
**********************************************

 1>  Ultimate Strength of concrete      180.00 ksc.

 2>  Yield  Strength  of  Steel         2500.00 ksc.

-----------------------------------------------
```

```
**********************************************
***********   LOAD CLASS LISTING   ***********
**********************************************

     =====> by  now  Load  Class  is  HS20

-----------------------------------------------
```

**********
********* BRIDGE DIMENSION LISTING *********
**********

| | | |
|---|---|---|
| 1> | Number of Traffic Lane | 2 |
| 2> | Lane Width | 3.00 m. |
| 3> | Outer Dimension Appart Ex-Gir. | 8.00 m. |
| 4> | Number of Girder per section | 4 |
| 5> | Overhanging Slab from Girder | 0.30 m. |
| 6> | Area of Parapet Family | 0.20 sq. m. |
| 7> | Number of span | 3 |
| 8> | Curb area | 0.10 sq. m. |
| 9> | Cross beam area | 0.10 sq. m. |

---

**********
********* BEAM DIMENSION LISTING *********
**********

| | | | |
|---|---|---|---|
| 1> | Beam Width | = | 0.30 m. |
| 2> | Length of Interval | = | 5.03 m. |

---

```
**********************************************
**********   SPAN  NUMBER   1   **********
**********************************************

   ====>   by now  Code of Beam Shape  =  1

   Code :        1 =  Straight haunch

                 2 =  Parabolar haunch

                 3 =  Constant Cross Section

   1>  Span  Length                  =     30.18 m.
       Number of Section Consider =          7 sect.
   2>  Left side  Depth              =      2.00 m.
   3>  Centre     Depth              =      1.80 m.
   4>  Right side Depth              =      2.00 m.
   5>  Left  side Length             =      6.71 m.
   6>  Right side Length             =      6.71 m.

-------------------------------------------------
```

```
**********************************************
**********   SPAN  NUMBER   2   ***************
**********************************************

====>   by now  Code of Beam Shape  =  1

Code :        1 =  Straight haunch
              2 =  Parabolar haunch
              3 =  Constant Cross Section

1>  Span  Length                =     40.23 m.
    Number of Section Consider  =        9 sect.
2>  Left side  Depth            =      2.00 m.
3>  Centre     Depth            =      1.00 m.
4>  Right side Depth            =      2.00 m.
5>  Left  side Length           =      6.71 m.
6>  Right side Length           =      6.71 m.

-----------------------------------------------
```

```
**********************************************
**********   SPAN NUMBER   3   ***************
**********************************************

====>   by now  Code of Beam Shape  =  1

Code :          1 =  Straight haunch
                2 =  Parabolar haunch
                3 =  Constant Cross Section

1>  Span  Length                  =      30.10 m.
    Number of Section Consider    =        7 sect.
2>  Left side  Depth              =       2.00 m.
3>  Centre     Depth              =       1.80 m.
4>  Right side Depth              =       2.00 m.
5>  Left  side Length             =       6.71 m.
6>  Right side Length             =       6.71 m.
```

**********
********** BAR SIZE LISTING **********
**********

| Bar # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Diam. (mm.) | 4 | 6 | 9 | 12 | 15 | 19 | 22 | 25 | 28 | 30 | 33 | 36 |

1> BEAM Bar Size # 8

2> BEAM Stirrup Bar Size # 3

3> SLAB Bar Size # 4

---

**********
********** SETTLEMENT LISTING **********
**********

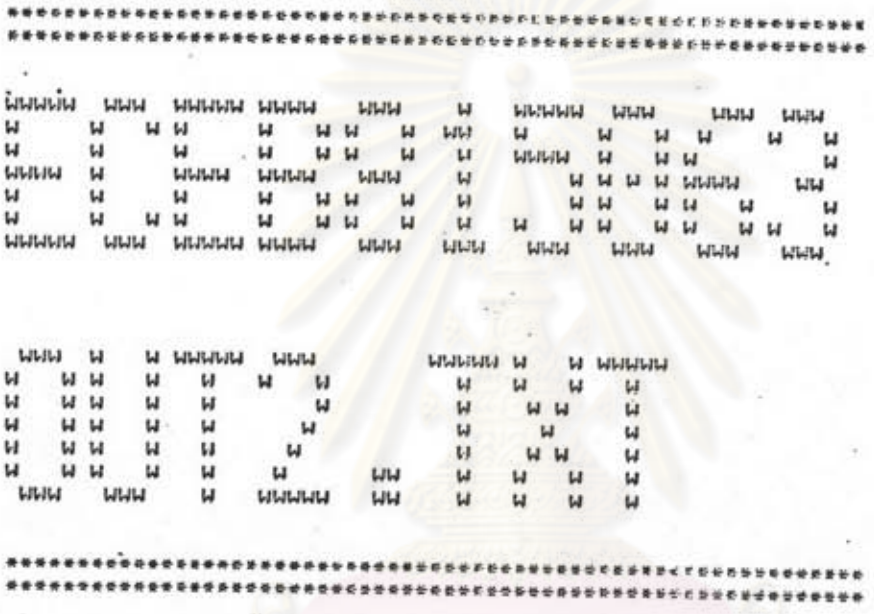
1> Settlement at Joint # 1 = 0.00 m.

2> Settlement at Joint # 2 = 0.01 m.

3> Settlement at Joint # 3 = 0.01 m.

4> Settlement at Joint # 4 = 0.00 m.

---



**********
CONTINOUS R.C. BEAM FOR BRIDGE DESIGN
**********

NUMBER OF SPANS = 3

SPAN NO. 1 INTERVAL = 6
LENGTH = 99.0000

SPAN NO. 2 INTERVAL = 6
LENGTH = 132.0000

SPAN NO. 3 INTERVAL = 6
LENGTH = 99.0000

**********
EXTERIOR-CIRDER
**********

UNIFORM LIVE LOAD INTERVAL = 668.367 KG/M
CONCENTRATED LOAD EQUIVALA = 5729.58 KG

GIRDER BENDING MOMENT-EQUIVALENT LOADING (TON-M)

| | DEAD-LOAD | LIVE-LOAD | | CONCEN.-LOAD | | TOTAL LL. | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 88.8743 | 37.0338 | -9.5416 | 22.6487 | 0.0000 | 59.6825 | -9.5416 | 13.3220 | 0.0000 | -0.1231 | 161.7557 | 0.0000 |
| 2 | 119.0609 | 57.1576 | -19.0832 | 33.3998 | 0.0000 | 90.5574 | -19.0832 | 20.2137 | 0.0000 | -0.2462 | 229.5858 | 0.0000 |
| 3 | 92.1459 | 60.3711 | -28.6248 | 33.6754 | 0.0000 | 94.0465 | -28.6248 | 20.9925 | 0.0000 | -0.3693 | 206.8156 | 0.0000 |
| 4 | 8.1292 | 46.6746 | -38.1664 | 25.9484 | -13.3030 | 72.6230 | -51.4695 | 16.2105 | -10.7005 | -0.4924 | 96.4702 | -54.5333 |
| 5 | -132.9892 | 19.9095 | -51.5495 | 13.5127 | -19.6015 | 33.4221 | -71.1510 | 7.4603 | -14.7923 | -0.6156 | 0.0000 | -219.5481 |
| 6 | -332.7956 | 11.8360 | -100.5345 | 5.2287 | -39.0761 | 17.0647 | -139.6106 | 3.8091 | -29.0251 | -0.7387 | 0.0000 | -502.1700 |
| 7 | -131.3506 | 16.2966 | -45.8097 | 14.6905 | -20.8601 | 30.9872 | -66.6698 | 6.0286 | -13.8607 | -0.7387 | 96.0575 | -43.0661 |
| 8 | 11.4036 | 44.2113 | -31.4490 | 27.2737 | -13.0340 | 71.4850 | -44.4830 | 13.9076 | -9.2480 | -0.7387 | 222.3670 | 0.0000 |
| 9 | 97.0561 | 69.5766 | -31.4491 | 35.9605 | -9.9902 | 105.5371 | -41.4393 | 20.5325 | -8.6152 | -0.7387 | 264.6192 | 0.0000 |
| 10 | 125.6069 | 78.0317 | -31.4491 | 38.9586 | 0.0000 | 116.9902 | -31.4491 | 22.7607 | 0.0000 | -0.7387 | 222.3670 | 0.0000 |
| 11 | 97.0561 | 69.5766 | -31.4492 | 35.9605 | -9.9903 | 105.5371 | -41.4395 | 20.5325 | -8.6153 | -0.7387 | 96.0580 | -43.0666 |
| 12 | 11.4036 | 44.2115 | -31.4492 | 27.2739 | -13.0341 | 71.4854 | -44.4834 | 13.9077 | -9.2481 | -0.7387 | 0.0000 | -212.6206 |
| 13 | -131.3507 | 16.2970 | -45.8101 | 14.6908 | -20.8603 | 30.9878 | -66.6704 | 6.0287 | -13.8608 | -0.7387 | 0.0000 | -502.1712 |
| 14 | -332.7961 | 11.8360 | -100.5346 | 5.2287 | -39.0765 | 17.0647 | -139.6111 | 3.3200 | -29.0252 | -0.7387 | 0.0000 | -219.5492 |
| 15 | -132.9899 | 19.9093 | -51.5495 | 13.5125 | -19.6018 | 33.4218 | -71.1513 | 7.4602 | -14.7924 | -0.6156 | 0.0000 | -219.5492 |
| 16 | 8.1287 | 46.6743 | -38.1662 | 25.9483 | -13.3032 | 72.6225 | -51.4693 | 16.2104 | -10.7005 | -0.4925 | 96.4692 | -54.5335 |
| 17 | 92.1457 | 60.3709 | -28.6246 | 33.6753 | 0.0000 | 94.0462 | -28.6246 | 20.9924 | 0.0000 | -0.3693 | 206.8149 | 0.0000 |
| 18 | 119.0609 | 57.1574 | -19.0831 | 33.3998 | 0.0000 | 90.5572 | -19.0831 | 20.2136 | 0.0000 | -0.2462 | 229.5855 | 0.0000 |
| 19 | 88.8745 | 37.0338 | -9.5415 | 22.6487 | 0.0000 | 59.6825 | -9.5415 | 13.3220 | 0.0000 | -0.1231 | 161.7558 | 0.0000 |

UNIFORM LIVE LOAD INTERVAL= 668.367 KG/M
CONCENTRATED LOAD INTERVAL= 8276.06 KG

GIRDER SHEAR-EQUIVALENT LOADING (TON)

| | DEAD-LOAD | LIVE-LOAD | | CONCEN.-LOAD | | TOTAL LL. | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 17.2342 | 7.3615 | -1.8967 | 6.5030 | 0.0000 | 13.8645 | -1.8967 | 3.0947 | -0.4234 | 0.0803 | 34.2737 | 0.0000 |
| 2 | 5.8537 | 4.7203 | -2.6168 | 4.7949 | 0.0000 | 9.5152 | -2.6168 | 2.1239 | -0.5841 | 0.0803 | 17.5731 | 0.0000 |
| 3 | -5.2193 | 2.7728 | -4.0307 | 3.2230 | 0.0000 | 5.9958 | -4.0307 | 1.3383 | -0.8997 | 0.0803 | 2.1951 | -10.0694 |
| 4 | -16.2922 | 1.4638 | -6.0830 | 1.8626 | -6.0080 | 3.3264 | -12.0910 | 0.7425 | -2.6989 | 0.0803 | 0.0000 | -31.0018 |
| 5 | -27.3652 | 0.7073 | -8.6879 | 0.7760 | -7.3684 | 1.4832 | -16.0562 | 0.3311 | -3.5840 | 0.0803 | 0.0000 | -46.9251 |
| 6 | -38.7457 | 0.3921 | -11.7341 | 0.2502 | -8.4550 | 0.6423 | -20.1891 | 0.1434 | -4.5065 | 0.0803 | 0.0000 | -63.3610 |
| 7 | 39.0635 | 13.1343 | -1.3696 | 7.4818 | 0.0000 | 20.6161 | -1.3696 | 4.0109 | -0.2665 | 0.0000 | 63.6905 | 0.0000 |
| 8 | 27.6823 | 10.0956 | -1.6922 | 6.4937 | -0.7942 | 16.5892 | -2.4864 | 3.2275 | -0.4837 | 0.0000 | 47.4990 | 0.0000 |
| 9 | 16.6094 | 7.4581 | -2.4161 | 5.3515 | -1.7824 | 12.8096 | -4.1985 | 2.4921 | -0.8168 | 0.0000 | 31.9112 | 0.0000 |
| 10 | 5.5365 | 5.2846 | -3.6039 | 4.1380 | 0.0000 | 9.4227 | -3.6039 | 1.8332 | -0.7012 | 0.0000 | 16.7923 | 0.0000 |
| 11 | -5.5365 | 3.6039 | -5.2846 | 2.9245 | -5.0120 | 6.5285 | -10.2966 | 1.2701 | -2.0032 | 0.0000 | 2.2621 | -17.8363 |
| 12 | -16.6094 | 2.4161 | -7.4582 | 1.7824 | -6.2255 | 4.1985 | -13.6836 | 0.8168 | -2.6622 | 0.0000 | 0.0000 | -32.9552 |
| 13 | -27.6824 | 1.6922 | -10.0956 | 0.8739 | -7.3676 | 2.5661 | -17.4632 | 0.4992 | -3.3975 | 0.0000 | 0.0000 | -48.5431 |
| 14 | -39.0636 | 1.3696 | -13.1343 | 0.8739 | -8.3558 | 2.2435 | -21.4901 | 0.4365 | -4.1810 | 0.0000 | 0.0000 | -64.7346 |
| 15 | 38.7457 | 11.7340 | -0.3921 | 7.5001 | 0.0000 | 19.2341 | -0.3921 | 4.2933 | -0.0875 | -0.0803 | 62.1929 | 0.0000 |
| 16 | 27.3652 | 8.6878 | -0.7073 | 6.4135 | -0.7760 | 15.1013 | -1.4832 | 3.3708 | -0.3311 | -0.0803 | 45.7570 | 0.0000 |
| 17 | 16.2922 | 6.0830 | -1.4638 | 5.0531 | 0.0000 | 11.1361 | -1.4638 | 2.4857 | -0.3267 | -0.0803 | 29.8338 | 0.0000 |
| 18 | 5.2193 | 4.0307 | -2.7728 | 3.4811 | 0.0000 | 7.5118 | -2.7728 | 1.6767 | -0.6189 | -0.0803 | 14.3276 | 0.0000 |
| 19 | -5.8537 | 2.6168 | -4.7203 | 1.7731 | 0.0000 | 4.3899 | -4.7203 | 0.9799 | -1.0536 | -0.0803 | 0.0000 | -11.7078 |
| 20 | -18.1570 | 1.8966 | -7.3615 | 0.9549 | 0.0000 | 2.8516 | -7.3615 | 0.6365 | -1.6432 | -0.0803 | 0.0000 | -27.2419 |

**********
INTERIOR-GIRDER
**********

UNIFORM LIVE LOAD INTERVAL = 668.367 KG/M
CONCENTRATED LOAD EQUIVALA = 5729.55 KG

GIRDER BENDING MOMENT-EQUIVALENT LOADING (TON-M)

| | DEAD-LOAD | LIVE-LOAD | | CONCEN. -LOAD | | TOTAL LL | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 86.7009 | 37.0338 | -9.5416 | 22.6487 | 0.0000 | 59.6825 | -9.5416 | 13.3220 | 0.0000 | -0.1231 | 159.5823 | 0.0000 |
| 2 | 116.1493 | 57.1576 | -19.0832 | 33.3996 | 0.0000 | 90.5574 | -19.0832 | 20.2137 | 0.0000 | -0.2462 | 226.6741 | 0.0000 |
| 3 | 89.8925 | 60.3711 | -28.6248 | 33.6754 | 0.0000 | 94.0465 | -28.6248 | 20.9925 | 0.0000 | -0.3693 | 204.5622 | 0.0000 |
| 4 | 7.9304 | 46.6746 | -38.1664 | 25.9484 | -13.3030 | 72.6230 | -51.4695 | 16.2105 | -10.7005 | -0.4924 | 96.2715 | -54.7321 |
| 5 | -129.7369 | 19.9095 | -51.5495 | 13.5127 | -19.6015 | 33.4221 | -71.1510 | 7.4603 | -14.7923 | -0.6156 | 0.0000 | -216.2959 |
| 6 | -324.6571 | 11.83e0 | -100.5345 | 5.2287 | -39.0761 | 17.0647 | -139.6106 | 3.8091 | -29.0251 | -0.7387 | 0.0000 | -494.0315 |
| 7 | -128.1384 | 16.2968 | -45.8097 | 14.6905 | -20.8601 | 30.9872 | -66.6698 | 6.0266 | -13.8607 | -0.7387 | 0.0000 | -209.4075 |
| 8 | 11.1247 | 44.2113 | -31.4490 | 27.2737 | -13.0340 | 71.4850 | -44.4830 | 13.9076 | -9.2490 | -0.7387 | 95.7786 | -43.3445 |
| 9 | 94.e82e | 69.57e6 | -31.4491 | 35.9605 | -9.9902 | 105.5371 | -41.4393 | 20.5325 | -5.6152 | -0.7387 | 220.0135 | 0.0000 |
| 10 | 122.5352 | 78.0317 | -31.4491 | 38.9586 | 0.0000 | 116.9902 | -31.4491 | 22.7607 | 0.0000 | -0.7387 | 261.5478 | 0.0000 |
| 11 | 94.6826 | 69.57ee | -31.4492 | 35.9605 | -9.9903 | 105.5371 | -41.4395 | 20.5325 | -5.6153 | -0.7387 | 220.0135 | 0.0000 |
| 12 | 11.1247 | 44.2115 | -31.4492 | 27.2739 | -13.0341 | 71.4854 | -44.4834 | 13.9077 | -8.2481 | -0.7387 | 95.7791 | -43.3459 |
| 13 | -126.1385 | 16.2970 | -45.8101 | 14.6908 | -20.8603 | 30.9878 | -66.6704 | 6.0287 | -13.8608 | -0.7387 | 0.0000 | -209.4084 |
| 14 | -324.6577 | 11.83e0 | -100.5346 | 5.2287 | -39.0765 | 17.0647 | -139.6111 | 3.3200 | -29.0252 | -0.7387 | 0.0000 | -494.0327 |
| 15 | -129.737e | 19.9093 | -51.5495 | 13.5125 | -19.6016 | 33.4218 | -71.1513 | 7.4602 | -14.7924 | -0.6156 | 0.0000 | -216.2965 |
| 16 | 7.9299 | 46.6743 | -38.1662 | 25.9483 | -13.3032 | 72.6225 | -51.4693 | 16.2104 | -10.7005 | -0.4925 | 96.2704 | -54.7323 |
| 17 | 89.8923 | 60.3709 | -28.6246 | 33.6753 | 0.0000 | 94.0462 | -28.6246 | 20.9924 | 0.0000 | -0.3693 | 204.5615 | 0.0000 |
| 18 | 116.1493 | 57.1574 | -19.0831 | 33.3998 | 0.0000 | 90.5572 | -19.0831 | 20.2136 | 0.0000 | -0.2462 | 226.6739 | 0.0000 |
| 19 | 86.7011 | 37.0338 | -9.5415 | 22.6487 | 0.0000 | 59.6825 | -9.5415 | 13.3220 | 0.0000 | -0.1231 | 159.5824 | 0.0000 |

UNIFORM LIVE LOAD INTERVAL= 665.367 KG/M
CONCENTRATED LOAD INTERVAL= 8276.06 KG

GIRDER SHEAR-EQUIVALENT LOADING (TON)

| | DEAD-LOAD | LIVE-LOAD | | CONCEN.-LOAD | | TOTAL LL. | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 17.6662 | 7.3615 | -1.8967 | 6.5030 | 0.0000 | 13.8645 | -1.8967 | 3.0947 | -0.4234 | 0.0803 | 34.7057 | 0.0000 |
| 2 | 6.0004 | 4.7203 | -2.6168 | 4.7949 | 0.0000 | 9.5152 | -2.6168 | 2.1239 | -0.5841 | 0.0803 | 17.7198 | 0.0000 |
| 3 | -5.3501 | 2.7728 | -4.0307 | 3.2230 | 0.0000 | 5.9958 | -4.0307 | 1.3383 | -0.8997 | 0.0803 | 2.0643 | -10.2007 |
| 4 | -16.7006 | 1.4638 | -6.0830 | 1.8626 | -6.0080 | 3.3264 | -12.0910 | 0.7425 | -2.6989 | 0.0803 | 0.0000 | -31.4102 |
| 5 | -28.0512 | 0.7073 | -8.6879 | 0.7760 | -7.3684 | 1.4832 | -16.0562 | 0.3311 | -3.5840 | 0.0803 | 0.0000 | -47.6111 |
| 6 | -39.7170 | 0.3921 | -11.7341 | 0.2502 | -8.4550 | 0.6423 | -20.1891 | 0.1434 | -4.5065 | 0.0803 | 0.0000 | -64.3323 |
| 7 | 40.0427 | 13.1343 | -1.3696 | 7.4818 | 0.0000 | 20.6161 | -1.3696 | 4.0109 | -0.2665 | 0.0000 | 64.6698 | 0.0000 |
| 8 | 28.3763 | 10.0956 | -1.6922 | 6.4937 | -0.7942 | 16.5892 | -2.4864 | 3.2275 | -0.4837 | 0.0000 | 48.1930 | 0.0000 |
| 9 | 17.0258 | 7.4581 | -2.4161 | 5.3515 | -1.7824 | 12.8096 | -4.1985 | 2.4921 | -0.8168 | 0.0000 | 32.3276 | 0.0000 |
| 10 | 5.6752 | 5.2846 | -3.6039 | 4.1380 | 0.0000 | 9.4227 | -3.6039 | 1.8332 | -0.7012 | 0.0000 | 16.9311 | 0.0000 |
| 11 | -5.6753 | 3.6039 | -5.2846 | 2.9245 | -5.0120 | 6.5285 | -10.2966 | 1.2701 | -2.0032 | 0.0000 | 2.1233 | -17.9751 |
| 12 | -17.0258 | 2.4161 | -7.4582 | 1.7824 | -6.2255 | 4.1985 | -13.6836 | 0.8168 | -2.6622 | 0.0000 | 0.0000 | -33.3716 |
| 13 | -28.3763 | 1.6922 | -10.0956 | 0.8739 | -7.3676 | 2.5661 | -17.4632 | 0.4992 | -3.3975 | 0.0000 | 0.0000 | -49.2370 |
| 14 | -40.0428 | 1.3696 | -13.1343 | 0.8739 | -8.3558 | 2.2435 | -21.4901 | 0.4365 | -4.1810 | 0.0000 | 0.0000 | -65.7139 |
| 15 | 39.7170 | 11.7340 | -0.3921 | 7.5001 | 0.0000 | 19.2341 | -0.3921 | 4.2933 | -0.0875 | -0.0803 | 63.1642 | 0.0000 |
| 16 | 26.0512 | 8.6878 | -0.7073 | 6.4135 | -0.7760 | 15.1013 | -1.4832 | 3.3708 | -0.3311 | -0.0803 | 46.4430 | 0.0000 |
| 17 | 16.7006 | 6.0830 | -1.4638 | 5.0531 | 0.0000 | 11.1361 | -1.4638 | 2.4857 | -0.3267 | -0.0803 | 30.2422 | 0.0000 |
| 18 | 5.3501 | 4.0307 | -2.7728 | 3.4811 | 0.0000 | 7.5118 | -2.7728 | 1.6767 | -0.6189 | -0.0803 | 14.4584 | 0.0000 |
| 19 | -6.0004 | 2.6168 | -4.7203 | 1.7731 | 0.0000 | 4.3899 | -4.7203 | 0.9799 | -1.0536 | -0.0803 | 0.0000 | -11.8546 |
| 20 | -18.6122 | 1.8966 | -7.3615 | 0.9549 | 0.0000 | 2.8516 | -7.3615 | 0.6365 | -1.6432 | -0.0803 | 0.0000 | -27.6971 |

**********
TRUCK EXTERIOR GIRDER
**********

GIRDER BENDING MOMENT-TRUCK LOADING (TON-M)

| | DEAD-LOAD | HI-AXIAL LOAD | | LOW-AXIAL LOAD | | TOTAL LL. | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 88.8743 | 3.5786 | -0.5262 | 1.3152 | -0.2611 | 62.5191 | -9.4630 | 13.9552 | 0.0000 | -0.1231 | 165.2255 | 0.0000 |
| 2 | 119.0609 | 5.2746 | -1.0523 | 2.6229 | -0.5221 | 94.8861 | -18.9260 | 21.1799 | 0.0000 | -0.2462 | 234.8807 | 0.0000 |
| 3 | 92.1459 | 5.3165 | -1.5782 | 2.6482 | -0.7813 | 95.6560 | -28.3759 | 21.3518 | 0.0000 | -0.3693 | 208.7843 | 0.0000 |
| 4 | 8.1292 | 4.0962 | -3.6134 | 2.0353 | -1.7903 | 73.6798 | -64.9765 | 16.4464 | -13.5086 | -0.4924 | 97.7630 | -70.6484 |
| 5 | -132.9892 | 2.1271 | -5.6487 | 1.0528 | -2.7994 | 38.2449 | -101.5771 | 8.5368 | -21.1179 | -0.6156 | 0.0000 | -256.2999 |
| 6 | -332.7956 | 0.8278 | -7.6840 | 0.4126 | -3.8084 | 14.8943 | -136.1778 | 3.3246 | -28.7272 | -0.7387 | 0.0000 | -500.4393 |
| 7 | -131.3506 | 2.3134 | -2.5453 | 1.1446 | -1.2686 | 41.5933 | -45.7987 | 8.0921 | -9.5216 | -0.7387 | 0.0000 | -187.4095 |
| 8 | 11.4036 | 4.3059 | -2.0634 | 2.1399 | -1.0284 | 77.4543 | -37.1283 | 15.0689 | -7.7190 | -0.7387 | 103.1882 | -34.1823 |
| 9 | 97.0561 | 5.6751 | -1.5815 | 2.7900 | -0.7883 | 101.9625 | -28.4578 | 19.8371 | -5.9164 | -0.7387 | 216.1169 | 0.0000 |
| 10 | 125.6069 | 4.0674 | -1.4625 | 1.9964 | -0.6946 | 73.0638 | -26.1776 | 14.2147 | 0.0000 | -0.7387 | 212.1468 | 0.0000 |
| 11 | 97.0561 | 2.4596 | -1.5816 | 1.2029 | -0.7883 | 44.1651 | -28.4563 | 8.5924 | -5.9165 | -0.7387 | 149.0749 | 0.0000 |
| 12 | 11.4036 | 1.8731 | -2.0634 | 0.9247 | -1.0284 | 33.6686 | -37.1288 | 6.5503 | -7.7191 | -0.7387 | 50.8838 | -34.1830 |
| 13 | -131.3507 | 2.3076 | -2.5453 | 1.1256 | -1.2686 | 41.4246 | -45.7993 | 8.0593 | -9.5217 | -0.7387 | 0.0000 | -187.4104 |
| 14 | -332.7961 | 0.8278 | -4.2117 | 0.4126 | -2.0819 | 14.8943 | -75.7156 | 2.8977 | -15.7413 | -0.7387 | 0.0000 | -424.9916 |
| 15 | -132.9899 | 2.1271 | -3.5098 | 1.0528 | -1.7350 | 38.2445 | -63.0964 | 8.5367 | -13.1178 | -0.6156 | 0.0000 | -209.8196 |
| 16 | 8.1287 | 4.0961 | -2.8078 | 2.0353 | -1.3880 | 73.6794 | -50.4771 | 16.4463 | -10.4942 | -0.4925 | 97.7619 | -53.3351 |
| 17 | 92.1457 | 5.3164 | -2.1059 | 2.6482 | -1.0410 | 95.6556 | -37.8578 | 21.3517 | 0.0000 | -0.3693 | 208.7837 | 0.0000 |
| 18 | 119.0609 | 5.2746 | -1.4039 | 2.6229 | -0.6940 | 94.8859 | -25.2386 | 21.1799 | 0.0000 | -0.2462 | 234.8805 | 0.0000 |
| 19 | 88.8745 | 3.5786 | -0.7020 | 1.3152 | -0.3470 | 62.5191 | -12.6193 | 13.9552 | 0.0000 | -0.1231 | 165.2256 | 0.0000 |

GIRDER SHEAR-TRUCK LOADING (TON)

| | DEAD-LOAD | HI-AXIAL LOAD | | LOW-AXIAL LOAD | | TOTAL LL. | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 17.6662 | 0.7743 | -0.1046 | 0.3834 | -0.0502 | 13.9216 | -1.8744 | 3.1075 | -0.4184 | 0.0803 | 34.7756 | 0.0000 |
| 2 | 6.0004 | 0.5242 | -0.1968 | 0.2571 | -0.0975 | 9.4162 | -3.5381 | 2.1018 | -0.7898 | 0.0803 | 17.5987 | 0.0000 |
| 3 | -5.3501 | 0.3521 | -0.3801 | 0.1708 | -0.1850 | 6.3171 | -6.8216 | 1.4101 | -1.5227 | 0.0803 | 2.4573 | -13.6141 |
| 4 | -16.7006 | 0.2032 | -0.5525 | 0.0961 | -0.2709 | 3.6359 | -9.9243 | 0.8116 | -2.2153 | 0.0803 | 0.0000 | -28.7600 |
| 5 | -28.0512 | 0.1593 | -0.7018 | 0.0762 | -0.3453 | 2.8607 | -12.6105 | 0.6385 | -2.8149 | 0.0803 | 0.0000 | -43.3963 |
| 6 | -39.7170 | 0.1593 | -0.8212 | 0.0782 | -0.4045 | 2.8607 | -14.7573 | 0.6385 | -3.2940 | 0.0803 | 0.0000 | -57.6881 |
| 7 | 40.0427 | 0.8193 | -0.0958 | 0.4036 | -0.0477 | 14.7227 | -1.7235 | 2.8643 | -0.3353 | 0.0000 | 57.6297 | 0.0000 |
| 8 | 28.3763 | 0.7471 | -0.0958 | 0.3705 | -0.0477 | 13.4353 | -1.7235 | 2.6139 | -0.3353 | 0.0000 | 44.4254 | 0.0000 |
| 9 | 17.0258 | 0.5820 | -0.1945 | 0.2871 | -0.0915 | 10.4598 | -3.4763 | 2.0350 | -0.6767 | 0.0000 | 29.5206 | 0.0000 |
| 10 | 5.6752 | 0.0958 | -0.3635 | 0.0477 | -0.1796 | 1.7235 | -6.5346 | 0.3353 | -1.2713 | 0.0000 | 7.7340 | -2.1307 |
| 11 | -5.6753 | 0.0958 | -0.3635 | 0.0477 | -0.1796 | 1.7235 | -6.5346 | 0.3353 | -1.2713 | 0.0000 | 0.0000 | -13.4813 |
| 12 | -17.0258 | 0.0958 | -0.3635 | 0.0477 | -0.1796 | 1.7235 | -6.5346 | 0.3353 | -1.2713 | 0.0000 | 0.0000 | -24.8315 |
| 13 | -28.3763 | 0.0958 | -0.3635 | 0.0477 | -0.1796 | 1.7235 | -6.5346 | 0.3353 | -1.2713 | 0.0000 | 0.0000 | -36.1823 |
| 14 | -40.0428 | 0.0958 | -0.8165 | 0.0477 | -0.4036 | 1.7235 | -14.6788 | 0.3353 | -2.8556 | 0.0000 | 0.0000 | -57.5774 |
| 15 | 39.7170 | 0.8212 | -0.1593 | 0.4045 | -0.0784 | 14.7573 | -2.8616 | 3.2940 | -0.6387 | -0.0803 | 57.6881 | 0.0000 |
| 16 | 28.0512 | 0.7018 | -0.1593 | 0.3453 | -0.0784 | 12.6106 | -2.8616 | 2.8149 | -0.6387 | -0.0803 | 43.3963 | 0.0000 |
| 17 | 16.7006 | 0.5525 | -0.2032 | 0.2709 | -0.0961 | 9.9244 | -3.6359 | 2.2153 | -0.8116 | -0.0803 | 28.7600 | 0.0000 |
| 18 | 5.3501 | 0.3801 | -0.3521 | 0.1850 | -0.1708 | 6.8216 | -6.3171 | 1.5227 | -1.4101 | -0.0803 | 13.6141 | -2.4572 |
| 19 | -6.0004 | 0.1928 | -0.5242 | 0.0915 | -0.2571 | 3.4510 | -9.4162 | 0.7703 | -2.1018 | -0.0803 | 0.0000 | -17.5987 |
| 20 | -18.6122 | 0.0783 | -0.7114 | 0.0374 | -0.3508 | 1.4024 | -12.7847 | 0.3130 | -2.8537 | -0.0803 | 0.0000 | -34.3308 |

**********
TRUCK INTERIOR GIRDER
**********

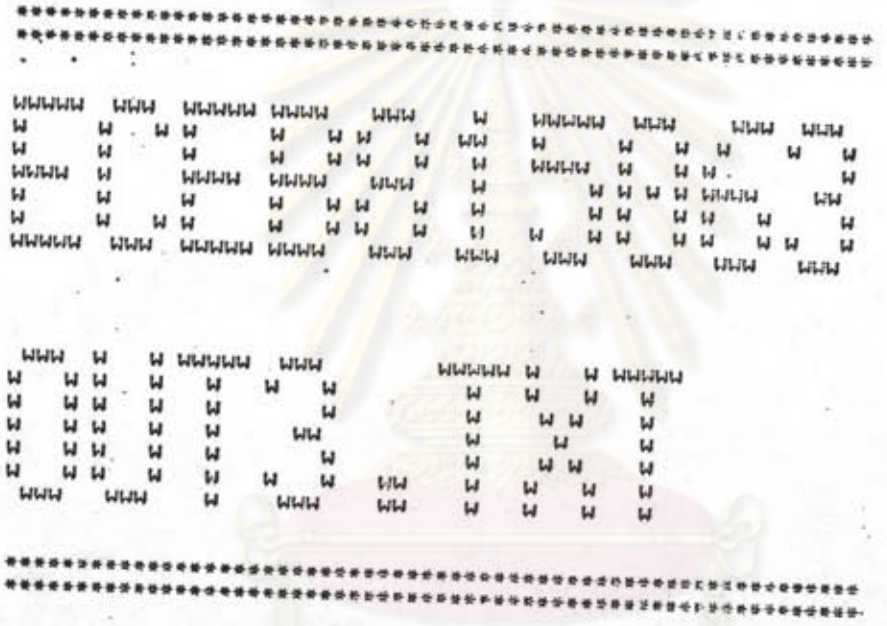
GIRDER BENDING MOMENT-TRUCK LOADING (TON-M)

| | DEAD-LOAD | HI-AXIAL LOAD | | LOW-AXIAL LOAD | | TOTAL LL. | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 86.7009 | 3.5786 | -0.5262 | 1.3152 | -0.2611 | 62.5191 | -9.4630 | 13.9552 | 0.0000 | -0.1231 | 163.0521 | 0.0000 |
| 2 | 116.1493 | 5.2746 | -1.0523 | 2.6229 | -0.5221 | 94.8861 | -18.9260 | 21.1799 | 0.0000 | -0.2462 | 231.9691 | 0.0000 |
| 3 | 89.8925 | 5.3165 | -1.5782 | 2.6482 | -0.7813 | 95.6560 | -28.3759 | 21.3518 | 0.0000 | -0.3693 | 206.5309 | 0.0000 |
| 4 | 7.9304 | 4.0962 | -3.6134 | 2.0353 | -1.7903 | 73.6798 | -64.9765 | 16.4464 | -13.5086 | -0.4924 | 97.5642 | -71.0472 |
| 5 | -129.7369 | 2.1271 | -5.6487 | 1.0528 | -2.7994 | 38.2449 | -101.5771 | 8.5368 | -21.1179 | -0.6156 | 0.0000 | -253.0476 |
| 6 | -324.6571 | 0.8278 | -7.6840 | 0.4126 | -3.8084 | 14.8943 | -138.1778 | 3.3246 | -26.7272 | -0.7387 | 0.0000 | -492.3008 |
| 7 | -128.1384 | 2.3134 | -2.5453 | 1.1446 | -1.2686 | 41.5933 | -45.7987 | 8.0921 | -9.5216 | -0.7387 | 0.0000 | -184.1974 |
| 8 | 11.1247 | 4.3059 | -2.0634 | 2.1399 | -1.0284 | 77.4543 | -37.1283 | 15.0689 | -7.7190 | -0.7387 | 102.9093 | -34.4612 |
| 9 | 94.6826 | 5.6751 | -1.5815 | 2.7900 | -0.7883 | 101.9625 | -28.4578 | 19.8371 | -5.9164 | -0.7387 | 215.7434 | 0.0000 |
| 10 | 122.5352 | 4.0674 | -1.4625 | 1.9964 | -0.6946 | 73.0638 | -26.1776 | 14.2147 | 0.0000 | -0.7387 | 209.0750 | 0.0000 |
| 11 | 94.6826 | 2.4596 | -1.5816 | 1.2029 | -0.7883 | 44.1651 | -28.4583 | 8.5924 | -5.9165 | -0.7387 | 146.7014 | 0.0000 |
| 12 | 11.1247 | 1.8731 | -2.0634 | 0.9247 | -1.0284 | 33.6686 | -37.1283 | 6.5503 | -7.7191 | -0.7387 | 50.6049 | -34.4619 |
| 13 | -128.1385 | 2.3076 | -2.5453 | 1.1256 | -1.2686 | 41.4246 | -45.7993 | 8.0593 | -9.5217 | -0.7387 | 0.0000 | -184.1982 |
| 14 | -324.6577 | 0.8278 | -4.2117 | 0.4126 | -2.0819 | 14.8943 | -75.7156 | 2.8977 | -15.7413 | -0.7387 | 0.0000 | -416.8533 |
| 15 | -129.7376 | 2.1271 | -3.5098 | 1.0528 | -1.7350 | 38.2445 | -63.0964 | 8.5367 | -13.1178 | -0.6156 | 0.0000 | -206.5674 |
| 16 | 7.9299 | 4.0961 | -2.8078 | 2.0353 | -1.3880 | 73.6794 | -50.4771 | 16.4463 | -10.4942 | -0.4925 | 97.5631 | -53.5338 |
| 17 | 89.8923 | 5.3164 | -2.1059 | 2.6482 | -1.0410 | 95.6556 | -37.8578 | 21.3517 | 0.0000 | -0.3693 | 206.5302 | 0.0000 |
| 18 | 116.1493 | 5.2746 | -1.4039 | 2.6229 | -0.6940 | 94.8859 | -25.2386 | 21.1799 | 0.0000 | -0.2462 | 231.9689 | 0.0000 |
| 19 | 86.7011 | 3.5786 | -0.7020 | 1.3152 | -0.3470 | 62.5191 | -12.6193 | 13.9552 | 0.0000 | -0.1231 | 163.0522 | 0.0000 |

GIRDER SHEAR-TRUCK LOADING (TON)

| | DEAD-LOAD | HI-AXIAL LOAD | | LOW-AXIAL LOAD | | TOTAL LL. | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 17.2342 | 0.7743 | -0.1046 | 0.3834 | -0.0502 | 13.9216 | -1.8744 | 3.1075 | -0.4184 | 0.0803 | 34.3436 | 0.0000 |
| 2 | 5.8537 | 0.5242 | -0.1968 | 0.2571 | -0.0975 | 9.4162 | -3.5381 | 2.1018 | -0.7898 | 0.0803 | 17.4520 | 0.0000 |
| 3 | -5.2193 | 0.3521 | -0.3801 | 0.1708 | -0.1850 | 6.3171 | -6.8216 | 1.4101 | -1.5227 | 0.0803 | 2.5881 | -13.4833 |
| 4 | -16.2922 | 0.2032 | -0.5525 | 0.0961 | -0.2709 | 3.6359 | -9.9243 | 0.8116 | -2.2153 | 0.0803 | 0.0000 | -28.3516 |
| 5 | -27.3652 | 0.1593 | -0.7018 | 0.0782 | -0.3453 | 2.8607 | -12.6105 | 0.6385 | -2.8149 | 0.0803 | 0.0000 | -42.7103 |
| 6 | -38.7457 | 0.1593 | -0.8212 | 0.0782 | -0.4045 | 2.8607 | -14.7573 | 0.6385 | -3.2940 | 0.0803 | 0.0000 | -56.7169 |
| 7 | 39.0635 | 0.8193 | -0.0958 | 0.4036 | -0.0477 | 14.7227 | -1.7235 | 2.8643 | -0.3353 | 0.0000 | 56.6505 | 0.0000 |
| 8 | 27.6823 | 0.7471 | -0.0958 | 0.3705 | -0.0477 | 13.4353 | -1.7235 | 2.6139 | -0.3353 | 0.0000 | 43.7315 | 0.0000 |
| 9 | 16.6094 | 0.5820 | -0.1945 | 0.2871 | -0.0915 | 10.4598 | -3.4783 | 2.0350 | -0.6767 | 0.0000 | 29.1042 | 0.0000 |
| 10 | 5.5365 | 0.0958 | -0.3635 | 0.0477 | -0.1796 | 1.7235 | -6.5346 | 0.3353 | -1.2713 | 0.0000 | 7.5952 | -2.2695 |
| 11 | -5.5365 | 0.0958 | -0.3635 | 0.0477 | -0.1796 | 1.7235 | -6.5346 | 0.3353 | -1.2713 | 0.0000 | 0.0000 | -13.3425 |
| 12 | -16.6094 | 0.0958 | -0.3635 | 0.0477 | -0.1796 | 1.7235 | -6.5346 | 0.3353 | -1.2713 | 0.0000 | 0.0000 | -24.4154 |
| 13 | -27.6824 | 0.0958 | -0.3635 | 0.0477 | -0.1796 | 1.7235 | -6.5346 | 0.3353 | -1.2713 | 0.0000 | 0.0000 | -35.4884 |
| 14 | -39.0636 | 0.0958 | -0.8165 | 0.0477 | -0.4036 | 1.7235 | -14.6786 | 0.3353 | -2.8558 | 0.0000 | 0.0000 | -56.5982 |
| 15 | 38.7457 | 0.8212 | -0.1593 | 0.4045 | -0.0784 | 14.7573 | -2.8616 | 3.2940 | -0.6387 | -0.0803 | 56.7168 | 0.0000 |
| 16 | 27.3652 | 0.7018 | -0.1593 | 0.3453 | -0.0784 | 12.6106 | -2.8616 | 2.8149 | -0.6387 | -0.0803 | 42.7103 | 0.0000 |
| 17 | 16.2922 | 0.5525 | -0.2032 | 0.2709 | -0.0961 | 9.9244 | -3.6359 | 2.2153 | -0.8116 | -0.0803 | 28.3516 | 0.0000 |
| 18 | 5.2193 | 0.3801 | -0.3521 | 0.1850 | -0.1708 | 6.8216 | -6.3171 | 1.5227 | -1.4101 | -0.0803 | 13.4833 | -2.5881 |
| 19 | -5.8537 | 0.1928 | -0.5242 | 0.0915 | -0.2571 | 3.4510 | -9.4162 | 0.7703 | -2.1018 | -0.0803 | 0.0000 | -17.4520 |
| 20 | -18.1570 | 0.0783 | -0.7114 | 0.0374 | -0.3508 | 1.4024 | -12.7847 | 0.3130 | -2.8537 | -0.0803 | 0.0000 | -33.6757 |

***** END OF BEAM ANALYSIS *****



```
**************************************************
**************     SLAB   DESIGN     ***************
**************************************************

 PLOAD =     16.0000
 SPAN eff. =     95.1450
 IMPACT MOMENT =     14297.4
 DEAD WEIGHT =     15.6250
 D.L. MOMENT =     1178.72
 TOTAL MOMENT =     63134.1
 d =     6.15304
 REAL TOTAL DEPTH =     8.50000
 DEAD WEIGHT =     121.875
 D.L. MOMENT =     9194.01
 TOTAL MOMENT =     71149.4
 d =     6.53228
 REAL TOTAL DEPTH =     9.00000
 DEAD WEIGHT =     128.125
 D.L. MOMENT =     9665.50
 TOTAL MOMENT =     71620.9
 d =     6.55389
 REAL TOTAL DEPTH =     9.00000
  Actual  d =     6.75000


 ========= TRANSVERSE  STEEL ==========

              USE # 4     Spacing =    0.10 m.

 ======== LONGITUDINAL STEEL ==========

              USE # 3     Spacing =    0.08 m.

 ========= TEMPERATURE STEEL ==========

              USE # 4     Spacing =    0.46 m.
```

**********
********** BEAM DESIGN **********
**********

**********
EXTERIOR DESIGN
**********

Span # 1

| Section No. | Distance (m.) | Depth (m.) | Top Bar # 11 | Bottom Bar # 11 | Stirrub # 4 (m.) |
|---|---|---|---|---|---|
| 1 | 0.00 | 2.00 | 2 | 2 | 0.40 |
| 2 | 5.03 | 1.85 | 2 | 5 | 0.91 |
| 3 | 10.06 | 1.80 | 2 | 7 | 0.89 |
| 4 | 15.09 | 1.80 | 2 | 7 | 0.39 |
| 5 | 20.12 | 1.80 | 2 | 3 | 0.18 |
| 6 | 25.15 | 1.85 | 9 | 7 | 0.12 |
| 7 | 30.18 | 2.00 | 16 | 16 | |

Span # 2

| Section No. | Distance (m.) | Depth (m.) | Top Bar # 11 | Bottom Bar # 11 | Stirrub # 4 (m.) |
|---|---|---|---|---|---|
| 1 | 0.00 | 2.00 | 16 | 16 | 0.13 |
| 2 | 5.03 | 1.85 | 7 | 5 | 0.19 |
| 3 | 10.06 | 1.80 | 2 | 3 | 0.37 |
| 4 | 15.09 | 1.80 | 2 | 7 | 0.89 |
| 5 | 20.12 | 1.80 | 2 | 8 | 15.97 |
| 6 | 25.15 | 1.80 | 2 | 7 | 0.35 |
| 7 | 30.18 | 1.80 | 2 | 3 | 0.17 |
| 8 | 35.21 | 1.85 | 7 | 5 | |

| | | | | | |
|---|---|---|---|---|---|
| 9 | 40.24 | 2.00 | 16 | 16 | 0.12 |

Span # 3

| Section No. | Distance (m.) | Depth (m.) | Top Bar # 11 | Bottom Bar # 11 | Stirrub # 4 (m.) |
|---|---|---|---|---|---|
| 1 | 0.00 | 2.00 | 16 | 16 | 0.14 |
| 2 | 5.03 | 1.85 | 7 | 5 | 0.20 |
| 3 | 10.06 | 1.80 | 2 | 3 | 0.43 |
| 4 | 15.09 | 1.80 | 2 | 7 | 0.89 |
| 5 | 20.12 | 1.80 | 2 | 7 | 847.55 |
| 6 | 25.15 | 1.85 | 2 | 5 | 0.34 |
| 7 | 30.18 | 2.00 | 2 | 2 | |

**********
INTERIOR DESIGN
**********

Span # 1

| Section No. | Distance (m.) | Depth (m.) | Top Bar # 11 | Bottom Bar # 11 | Stirrub # 4 (m.) |
|---|---|---|---|---|---|
| 1 | 0.00 | 2.00 | 2 | 2 | 0.45 |
| 2 | 5.03 | 1.85 | 2 | 5 | 0.91 |
| 3 | 10.06 | 1.80 | 2 | 7 | 0.89 |
| 4 | 15.09 | 1.80 | 2 | 6 | 0.44 |
| 5 | 20.12 | 1.80 | 2 | 3 | 0.20 |
| 6 | 25.15 | 1.85 | 8 | 6 | 0.13 |
| 7 | 30.19 | 2.00 | 15 | 15 | |

Span # 2

| Section No. | Distance (m.) | Depth (m.) | Top Bar # 11 | Bottom Bar # 11 | Stirrub # 4 (m.) |
|---|---|---|---|---|---|
| 1 | 0.00 | 2.00 | 15 | 15 | |
| 2 | 5.03 | 1.85 | 7 | 4 | 0.15 |
| 3 | 10.06 | 1.80 | 2 | 3 | 0.20 |
| 4 | 15.09 | 1.80 | 2 | 7 | 0.41 |
| 5 | 20.12 | 1.80 | 2 | 8 | 0.89 |
| 6 | 25.15 | 1.80 | 2 | 7 | 0.89 |
| 7 | 30.18 | 1.80 | 2 | 3 | 0.38 |
| 8 | 35.21 | 1.85 | 7 | 4 | 0.19 |
| 9 | 40.24 | 2.00 | 15 | 15 | 0.13 |

Span # 3

| Section No. | Distance (m.) | Depth (m.) | Top Bar # 11 | Bottom Bar # 11 | Stirrub # 4 (m.) |
|---|---|---|---|---|---|
| 1 | 0.00 | 2.00 | 15 | 15 | |
| 2 | 5.03 | 1.83 | 7 | 5 | 0.15 |
| 3 | 10.06 | 1.80 | 2 | 3 | 0.22 |
| 4 | 15.09 | 1.80 | 2 | 6 | 0.49 |
| 5 | 20.12 | 1.80 | 2 | 7 | 0.89 |
| 6 | 25.15 | 1.85 | 2 | 5 | 0.89 |
| 7 | 30.18 | 2.00 | 2 | 2 | 0.38 |

GUANTITY OF CONCRETE (cu. m)

| | Girder (cu. m) | Total (cu. m) |
|---|---|---|
| 1 | 222.74 | 376.55 |

GUANTITY OF FORMWORK (sq. m)

| | Girder (sq. m) | Total (sq. m) |
|---|---|---|
| 7 | 1605.63 | 2385.75 |

GUANTITY OF STEEL (ton)

| | Girder (ton) | Total (ton) |
|---|---|---|
| | 43.77 | 45.69 |

DEFLECTION EACH SPAN (MIDDLE SPAN)(CM.)

---

DEF. FOR SPAN 1 = 2.831 CM.
DEF. FOR SPAN 2 = 6.974 CM.
DEF. FOR SPAN 3 = 2.831 CM.

---

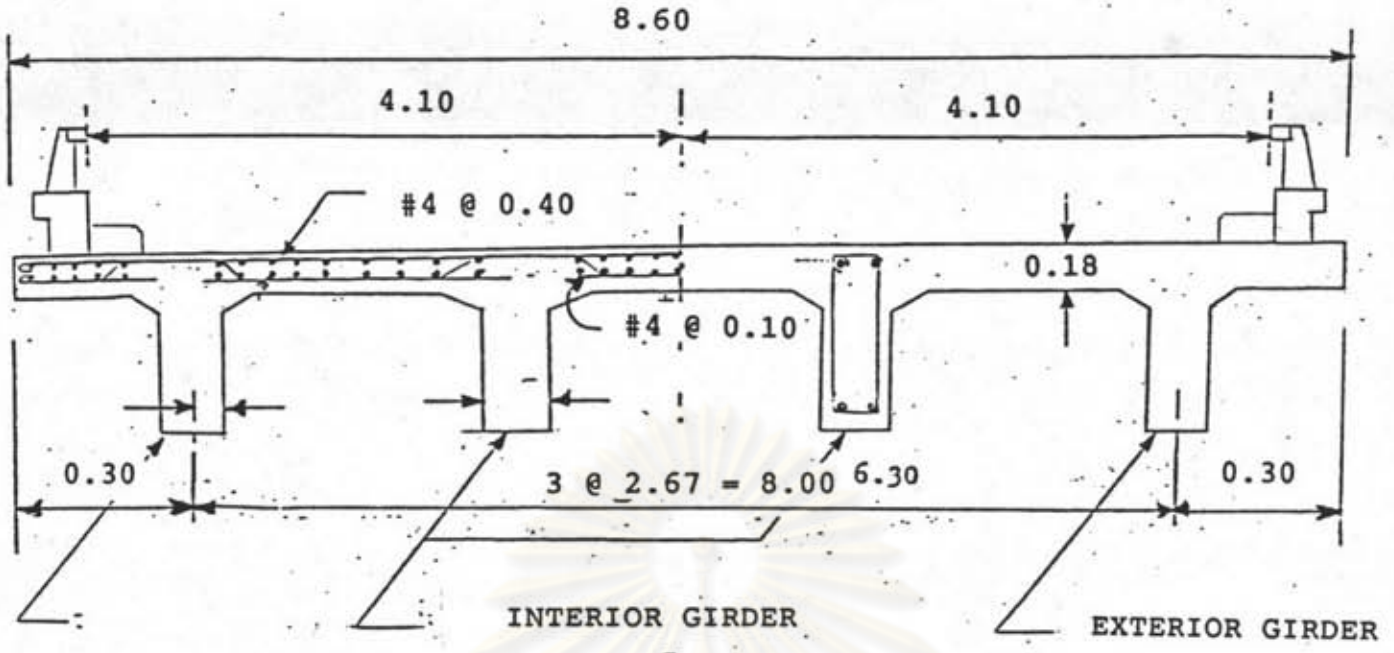
---

REATION AT SUPPORT (TON.)

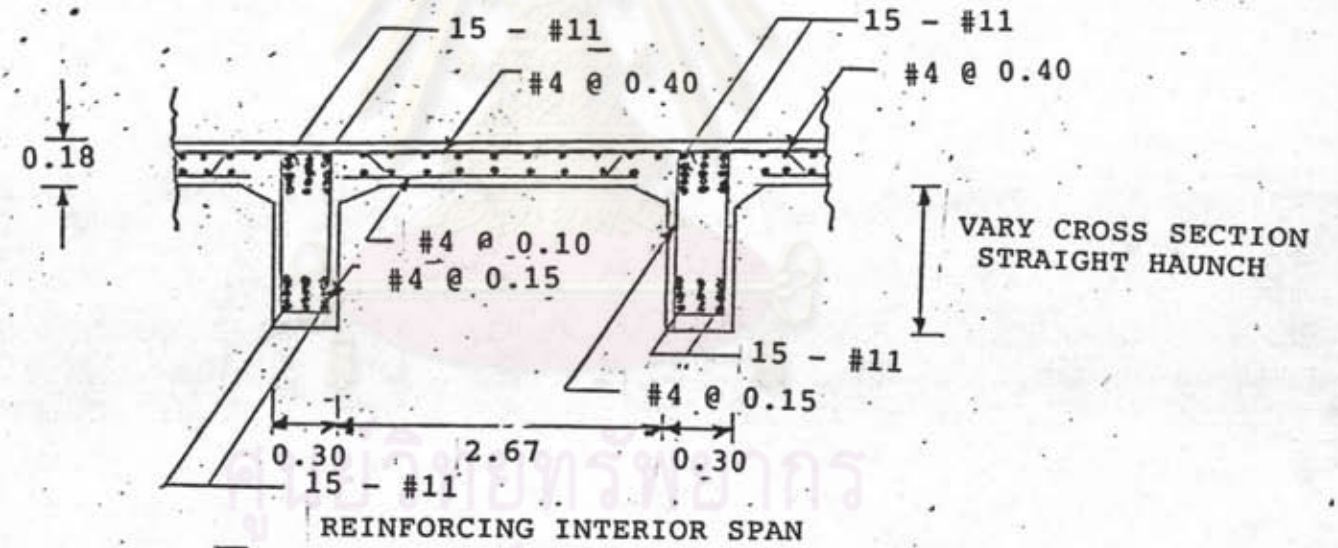
---

REACTION AT SUPPORT NO. 1= 140 TON.
REACTION AT SUPPORT NO. 2= 523 TON.
REACTION AT SUPPORT NO. 3= 523 TON.
REACTION AT SUPPORT NO. 4= 139 TON.

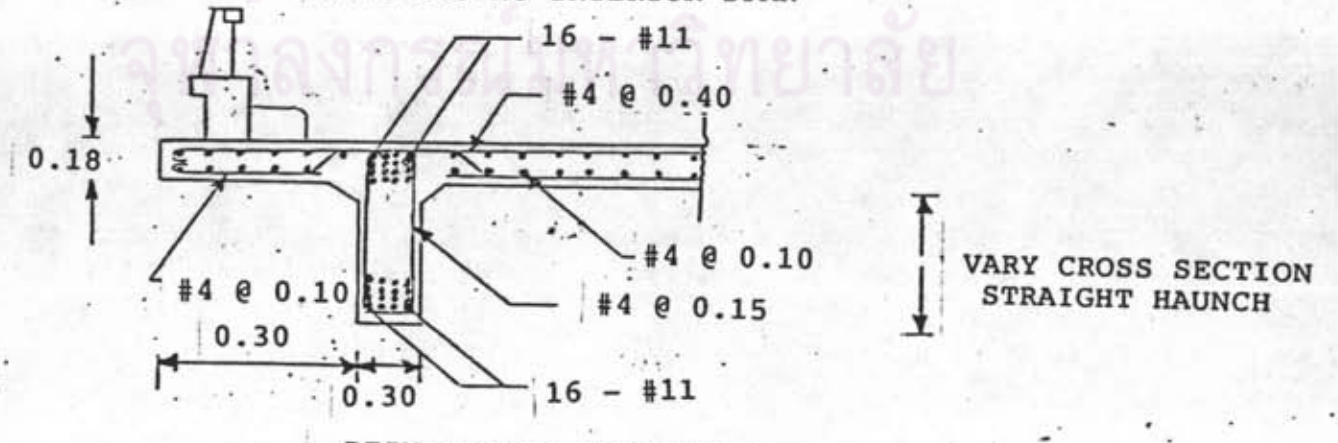
---



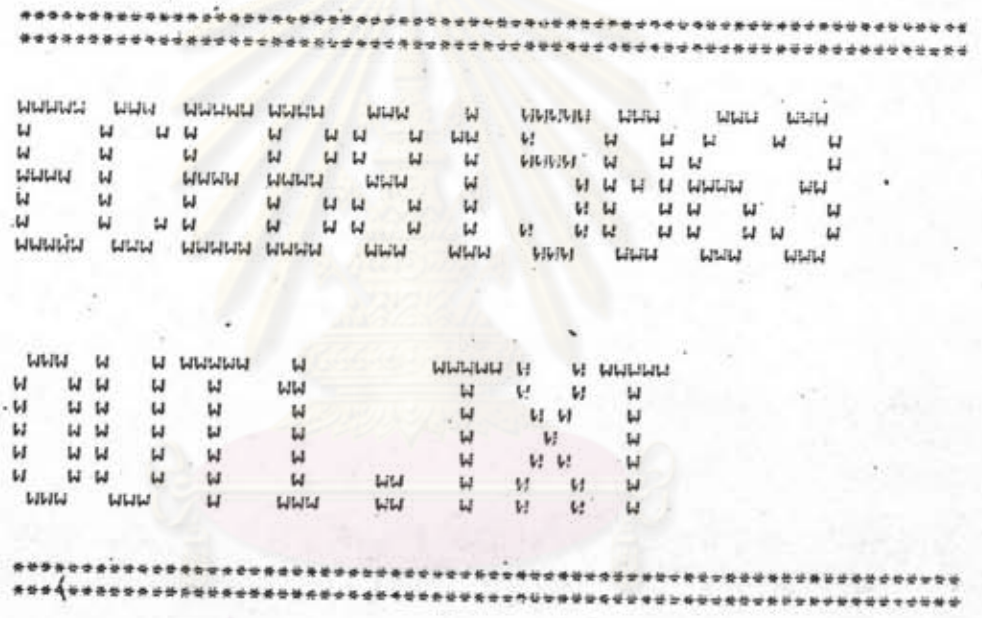

TYPICAL CROSS SECTION




REINFORCING INTERIOR SPAN




REINFORCING EXTERIOR SPAN



**********
********** TITLE HEADING LIST **********
**********

An Existing Title is shown below

CONTINUOUS R.C. BEAM T-UP BRIDGE DESIGN

---

**********
********** MATERIAL PROPERTIES **********
**********

1> Ultimate Strength of concrete 180.00 ksc.

2> Yield Strength of Steel 2500.00 ksc.

---

**********
********** LOAD CLASS LISTING **********
**********

=====> by now Load Class is HS20

---

```
**********************************************
**********    SETTLEMENT   LISTING    **********
**********************************************

   1> Settlement at Joint  #  1  =     0.00  m.

   2> Settlement at Joint  #  2  =     0.01  m.

   3> Settlement at Joint  #  3  =     0.01  m.

   4> Settlement at Joint  #  4  =     0.00  m.
```

```
**********************************************
**************    SLAB   DESIGN    **************
**********************************************

  PLOAD =     16.0000
  SPAN eff. =     95.1450
  IMPACT MOMENT =     14297.4
  DEAD WEIGHT =     15.6250
  D.L. MOMENT =     1178.72
  TOTAL MOMENT =     63134.1
  d =     5.82725
  REAL TOTAL DEPTH =     8.50000
  DEAD WEIGHT =     121.875
  D.L. MOMENT =     9194.01
  TOTAL MOMENT =     71149.4
  d =     6.18510
  REAL TOTAL DEPTH =     8.50000
   Actual  d =     6.25000

  ========= TRANSVERSE  STEEL ==========

                USE # 4     Spacing :    0.08 m.

  ======== LONGITUDINAL STEEL ==========

                USE # 3     Spacing :    0.05 m.

  ========= TEMPERATURE STEEL ==========

                USE # 4     Spacing :    0.46 m.
```

```
**********  ************************************
**********  EDGE  DIMENSION  LISTING  **********
**********  ************************************

 1>  Numb.  : Traffic Lane                    2
 2>  Lane   -idth                          3.00 m.
 3>  Oute.  .ension Appart Ex-Gir.         8.00 m.
 4>  Numb.  - Girder per section              4
 5>  Over   .ng Slab from Girder           0.30 m.
 6>  Area   Parapet Family                 0.20 sq. m.
 7>  Numb   : span                            3
 8>  Curb   :a                             0.10 sq. m.
 9>  Cros   n  area                        0.10 sq. m.

----------  ------------------------------------
```

```
********  ************************************
********  BEAM  DIMENSION  LISTING  **********
********  ************************************

      1:  m  Width              =     0.30 m.
      2:  gth  of  Interval     =     5.03 m.

----------  ---------  ------------------------
```



```
**********  ************************************
**********  >  SPAN  NUMBER  1  ****************
**********  ************************************

  ====>  , now  Code of Beam Shape  =  2

  Code :      1 =  Straight haunch
```

2 = Parabolar haunch

3 = Constant Cross Section

| | | | |
|---|---|---|---|
| 1> | Span Length | = | 30.18 m. |
| | Number of Section Consider | = | 7 sect. |
| 2> | Left side Depth | = | 2.00 m. |
| 3> | Centre Depth | = | 1.80 m. |
| 4> | Right side Depth | = | 2.00 m. |
| 5> | Left side Length | = | 6.71 m. |
| 6> | Right side Length | = | 6.71 m. |

---

**********
********** SPAN NUMBER 2 **********
**********

====> by now Code of Beam Shape = 2

Code : 1 = Straight haunch

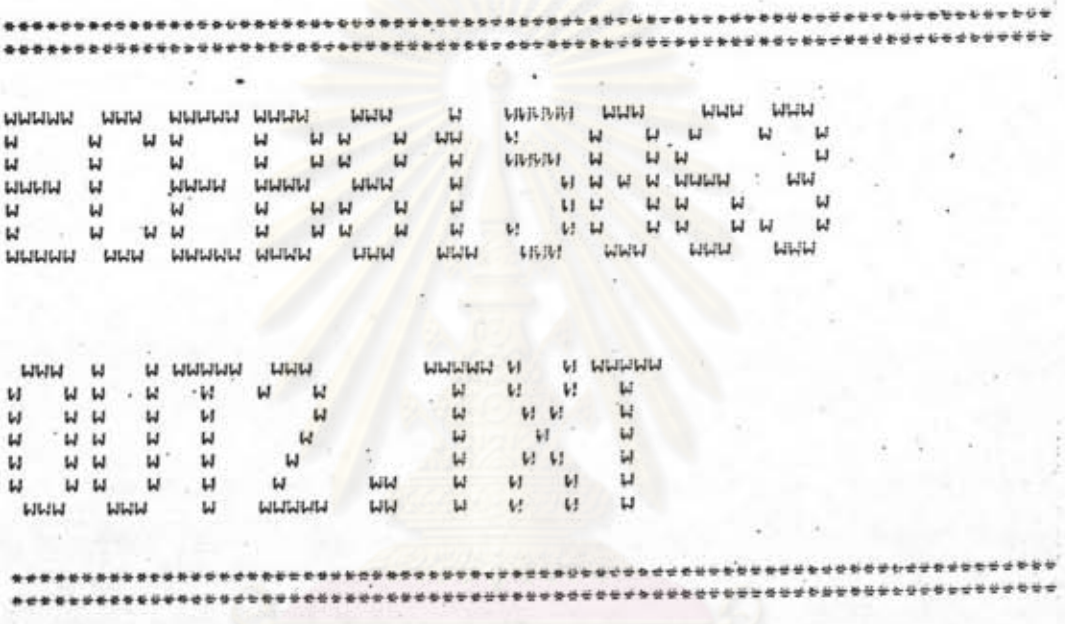
2 = Parabolar haunch

3 = Constant Cross Section

| | | | |
|---|---|---|---|
| 1> | Span Length | = | 40.20 m. |
| | Number of Section Consider | = | 9 sect. |
| 2> | Left side Depth | = | 2.00 m. |
| 3> | Centre Depth | = | 1.80 m. |
| 4> | Right side Depth | = | 2.00 m. |
| 5> | Left side Length | = | 6.71 m. |
| 6> | Right side Length | = | 6.71 m. |

---

```
**********************************************************
*************   SPAN NUMBER  3   *************************
**********************************************************

   ====>   by now  Code of Beam Shape  =  2

   Code :        1 =  Straight haunch

                 2 =  Parabolar haunch

                 3 =  Constant Cross Section

   1>  Span  Length                    =      30.18 m.

       Number of Section Consider      =        7 sect.

   2>  Left side  Depth                =       2.00 m.

   3>  Centre     Depth                =       1.80 m.

   4>  Right side Depth                =       2.00 m.

   5>  Left  side Length               =       6.71 m.

   6>  Right side Length               =       6.71 m.

-------------------------------------------------------------

**********************************************************
*************   BAR SIZE  LISTING   **********************
**********************************************************

Bar #        1   2   3   4   5   6   7   8   9  10  11
Diam.(mm.)   4   6   9  12  16  19  25  28  30  33  36

   1>  BEAM Bar Size                #          11

   2>  BEAM Stirrup  Bar Size       #           4

   3>  SLAB Bar Size                #           4

-------------------------------------------------------------
```



```
**************************************************
  CONTINUOUS  R.C.  BEAM  FOR  BRIDGE  DESIGN
**************************************************

NUMBER  OF  SPANS  =    3

SPAN NO.     1     INTERVAL =    6
                   LENGTH   =   99.0157

SPAN NO.     2     INTERVAL =    8
                   LENGTH   =  131.9100

SPAN NO.     3     INTERVAL =    6
                   LENGTH   =   99.0157


***************
EXTERIOR-GIRDER
***************

  UNIFORM DEAD LOAD INTERVAL  =    2507.71    KG/M
  UNIFORM LIVE LOAD INTERVAL  =    6661.367   KG/M
  CONCENTRATED LOAD EQUIVALA  =    8776.58    KG

                    GIRDER BENDING MOMENT-EGUIVALENT LOADING (TON-M)

    DEAD-LOAD      LIVE-LOAD      CONCEN.-LOAD      TOTAL LL.      IMPACT LOAD     SETTLE      TOTAL
 1   88.3459   37.0868   -9.4174   59.6729   0.0000   59.7597   -9.4174   13.3383   0.0000   -0.1207   161.3232   0.0000
 2  119.1744   57.2582  -18.8348   80.4711   0.0000   90.7493  -18.8346   20.2551   0.0000   -0.2413   229.9374   0.0000
 3   92.8832   60.5141  -28.2521   80.0400   0.0000   94.3541  -28.2521   21.0597   0.0000   -0.3620   207.9350   0.0000
 4    9.4722   46.8546  -37.6695   56.1774  -13.1076  72.9819  -50.7771   16.2895  -10.5565   -0.4826    98.2610  -52.3439
 5 -131.0585   19.7800  -50.5874   33.5679  -19.1428  33.3499  -69.7302    7.4437  -14.4968   -0.6033     0.0000 -215.8587
 6 -329.1071   11.4841  -99.1995    5.0504  -38.4674  16.5426 -137.6669    3.6923  -28.6207   -0.7239     0.0000 -496.1156
 7 -128.8393   16.2285  -44.7543   14.7608  -20.3809  30.9822  -65.1352    6.0280  -13.5415   -0.7239     0.0000 -208.2400
 8   13.9252   44.9634  -31.2111   27.4763  -12.8399  72.4397  -44.0510   14.0940   -9.1581   -0.7239    99.7349  -40.0079
 9   99.5841   70.3303  -31.2111   38.7179   -9.8569 106.5502  -41.0679   20.7306   -8.5380   -0.7239   226.1408    0.0000
10  128.1374   78.7859  -31.2110   39.7602    0.0000 118.0191  -31.2110   22.9620    0.0000   -0.7239   268.3945    0.0000
11   99.5852   70.3302  -31.2110   38.7179   -9.8568 106.5501  -41.0679   20.7305   -8.5380   -0.7239   226.1419    0.0000
12   13.9274   44.9633  -31.2110   27.4766  -12.8399  72.4399  -44.0509   14.0940   -9.1581   -0.7239    99.7374  -40.0056
13 -128.8360   16.2286  -44.7545   14.7608  -20.3808  30.9824  -65.1354    6.0280  -13.5415   -0.7239     0.0000 -208.2369
14 -329.1078   11.4842  -99.1997    5.0504  -38.4674  16.5426 -137.6671    3.2186  -28.6208   -0.7239     0.0000 -496.1196
15 -131.0594   19.7800  -50.5877   33.5679  -19.1428  33.3499  -69.7305    7.4437  -14.4969   -0.6033     0.0000 -215.8900
16    9.4716   46.8546  -37.6697   56.1774  -13.1076  72.9820  -50.7773   16.2895  -10.5565   -0.4826    98.2604  -52.3449
17   92.8828   60.5142  -28.2523   80.0400    0.0000  94.3542  -28.2523   21.0597    0.0000   -0.3620   207.9347    0.0000
18  119.1743   57.2583  -18.8349   80.4711    0.0000  90.7493  -18.8349   20.2551    0.0000   -0.2413   229.9374    0.0000
19   88.3460   37.0869   -9.4174   59.6729    0.0000  59.7598   -9.4174   13.3383    0.0000   -0.1207   161.3235    0.0000


UNIFORM  DEAD  LOAD INTERVAL=    2507.71    KG/M
UNIFORM  LIVE  LOAD INTERVAL=    6661.367   KG/M
CONCENTRATED   LOAD INTERVAL=    8776.58    KG
```

GIRDER SHEAR-EQUIVALENT LOADING (TON)

| | DEAD-LOAD | LIVE-LOAD | | CONCEN. -LOAD | | TOTAL LL. | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 17.5584 | 7.3709 | -1.8717 | 6.3609 | 0.0000 | 13.6797 | -1.8717 | 3.0979 | -0.4178 | 0.0787 | 34.6147 | 0.0000 |
| 2 | 6.1270 | 4.7268 | -2.5895 | 4.1073 | 0.0000 | 9.5341 | -2.5895 | 2.1280 | -0.5760 | 0.0787 | 17.8678 | 0.0000 |
| 3 | -5.2253 | 2.7740 | -3.9986 | 3.2262 | 0.0000 | 6.0123 | -3.9986 | 1.3419 | -0.8925 | 0.0787 | 2.2076 | -10.0177 |
| 4 | -16.5776 | 1.4586 | -6.0451 | 1.8751 | -5.9785 | 3.3338 | -12.0236 | 0.7441 | -2.6837 | 0.0787 | 0.0000 | -31.2062 |
| 5 | -27.9299 | 0.6969 | -8.6452 | 0.7771 | -7.3416 | 1.4760 | -15.9869 | 0.3294 | -3.5682 | 0.0787 | 0.0000 | -47.4064 |
| 6 | -39.3613 | 0.3804 | -11.6906 | 0.2450 | -8.4377 | 0.6224 | -20.1283 | 0.1389 | -4.4926 | 0.0787 | 0.0000 | -60.9036 |
| 7 | 39.8122 | 13.1129 | -1.3463 | 7.4018 | 0.0000 | 20.5947 | -1.3463 | 4.0069 | -0.2619 | 0.0000 | 64.4139 | 0.0000 |
| 8 | 28.3809 | 10.0736 | -1.6689 | 5.4902 | -0.7942 | 16.5639 | -2.4632 | 3.2227 | -0.4792 | 0.0000 | 48.1674 | 0.0000 |
| 9 | 17.0286 | 7.4372 | -2.3944 | 3.3407 | -1.7858 | 12.7859 | -4.1802 | 2.4876 | -0.8133 | 0.0000 | 32.3021 | 0.0000 |
| 10 | 5.6762 | 5.2644 | -3.5835 | 1.7810 | 0.0000 | 9.4025 | -3.5835 | 1.8294 | -0.6972 | 0.0000 | 16.9081 | 0.0000 |
| 11 | -5.6761 | 3.5835 | -5.2644 | 0.9273 | -4.9946 | 6.5108 | -10.2591 | 1.2668 | -1.9960 | 0.0000 | 2.1015 | -17.9211 |
| 12 | -17.0284 | 2.3944 | -7.4372 | 0.4168 | -6.2053 | 4.1802 | -13.6425 | 0.8133 | -2.6543 | 0.0000 | 0.0000 | -33.3252 |
| 13 | -28.3807 | 1.6689 | -10.0737 | 0.1666 | -7.3468 | 2.5255 | -17.4205 | 0.4914 | -3.3894 | 0.0000 | 0.0000 | -49.1906 |
| 14 | -39.8130 | 1.3463 | -13.1129 | 0.1666 | -8.3384 | 2.2029 | -21.4514 | 0.4286 | -4.1736 | 0.0000 | 0.0000 | -65.4380 |
| 15 | 39.3613 | 11.6906 | -0.3804 | 7.4769 | 0.0000 | 19.1876 | -0.3804 | 4.2826 | -0.0849 | -0.0787 | 62.7528 | 0.0000 |
| 16 | 27.9299 | 8.6452 | -0.6969 | 5.4009 | -0.7791 | 15.0461 | -1.4760 | 3.3583 | -0.3294 | -0.0787 | 46.2557 | 0.0000 |
| 17 | 16.5776 | 6.0451 | -1.4586 | 3.3378 | 0.0000 | 11.0829 | -1.4586 | 2.4737 | -0.3256 | -0.0787 | 30.0555 | 0.0000 |
| 18 | 5.2253 | 3.9986 | -2.7740 | 1.4648 | 0.0000 | 7.4674 | -2.7740 | 1.6667 | -0.6192 | -0.0787 | 14.2808 | 0.0000 |
| 19 | -6.1270 | 2.5895 | -4.7268 | 0.7672 | 0.0000 | 4.3567 | -4.7268 | 0.9724 | -1.0550 | -0.0787 | 0.0000 | -11.9875 |
| 20 | -18.7407 | 1.8717 | -7.3709 | 0.2407 | 0.0000 | 2.8124 | -7.3709 | 0.6277 | -1.6452 | -0.0787 | 0.0000 | -27.8354 |

**********
INTERIOR-GIRDER
**********

UNIFORM DEAD LOAD INTERVAL = [illegible] KG/M
UNIFORM LIVE LOAD INTERVAL = [illegible] KG/M
CONCENTRATED LOAD EQUIVALA = [illegible] KG

GIRDER BENDING MOMENT-EQUIVALENT LOADING (TON-M)

| | DEAD-LOAD | LIVE-LOAD | | CONCEN. -LOAD | | TOTAL LL. | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 86.1854 | 37.0868 | -9.4174 | [illegible] | 0.0000 | 59.7597 | -9.4174 | 13.3383 | 0.0000 | -0.1207 | 159.1627 | 0.0000 |
| 2 | 116.2599 | 57.2582 | -18.8348 | [illegible] | 0.0000 | 90.7493 | -18.8348 | 20.2551 | 0.0000 | -0.2413 | 227.0230 | 0.0000 |
| 3 | 90.6117 | 60.5141 | -28.2521 | [illegible] | 0.0000 | 94.3541 | -28.2521 | 21.0597 | 0.0000 | -0.3620 | 205.6635 | 0.0000 |
| 4 | 9.2406 | 46.0546 | -37.6695 | [illegible] | -13.1076 | 72.9819 | -50.7771 | 16.2895 | -10.5565 | -0.4826 | 98.0294 | -62.5756 |
| 5 | -127.8534 | 19.7800 | -50.5874 | [illegible] | -19.1428 | 33.3499 | -69.7302 | 7.4437 | -14.4968 | -0.6033 | 0.0000 | -212.6837 |
| 6 | -321.0588 | 11.4841 | -99.1995 | [illegible] | -38.4674 | 16.5426 | -137.6669 | 3.6923 | -28.6207 | -0.7239 | 0.0000 | -488.0703 |
| 7 | -125.6885 | 16.2285 | -44.7543 | [illegible] | -20.3809 | 30.9822 | -65.1352 | 6.0280 | -13.5415 | -0.7239 | 0.0000 | -205.0892 |
| 8 | 13.5846 | 44.9634 | -31.2111 | [illegible] | -12.8399 | 72.4397 | -44.0510 | 14.0940 | -9.1581 | -0.7239 | 99.3944 | -40.3485 |
| 9 | 97.1487 | 70.3303 | -31.2111 | [illegible] | -9.8569 | 106.5502 | -41.0679 | 20.7306 | -8.5350 | -0.7239 | 223.7055 | 0.0000 |
| 10 | 125.0038 | 78.7859 | -31.2110 | [illegible] | 0.0000 | 118.0191 | -31.2110 | 22.9620 | 0.0000 | -0.7239 | 265.2609 | 0.0000 |
| 11 | 97.1499 | 70.3302 | -31.2110 | [illegible] | -9.8568 | 106.5501 | -41.0679 | 20.7305 | -8.5380 | -0.7239 | 223.7066 | 0.0000 |
| 12 | 13.5868 | 44.9633 | -31.2110 | [illegible] | -12.8399 | 72.4399 | -44.0509 | 14.0940 | -9.1581 | -0.7239 | 99.3968 | -40.3462 |
| 13 | -125.6853 | 16.2285 | -44.7545 | [illegible] | -20.3808 | 30.9824 | -65.1354 | 6.0280 | -13.5415 | -0.7239 | 0.0000 | -205.0862 |
| 14 | -321.0594 | 11.4842 | -99.1997 | [illegible] | -38.4674 | 16.5426 | -137.6671 | 3.2186 | -28.6208 | -0.7239 | 0.0000 | -488.0713 |
| 15 | -127.8543 | 19.7800 | -50.5877 | [illegible] | -19.1428 | 33.3499 | -69.7305 | 7.4437 | -14.4969 | -0.6033 | 0.0000 | -212.6850 |
| 16 | 9.2400 | 46.8546 | -37.6697 | [illegible] | -13.1076 | 72.9820 | -50.7773 | 16.2895 | -10.5565 | -0.4826 | 98.0288 | -62.5769 |
| 17 | 90.6113 | 60.5142 | -28.2523 | [illegible] | 0.0000 | 94.3542 | -28.2523 | 21.0597 | 0.0000 | -0.3620 | 205.6632 | 0.0000 |
| 18 | 116.2599 | 57.2583 | -18.8349 | [illegible] | 0.0000 | 90.7493 | -18.8349 | 20.2551 | 0.0000 | -0.2413 | 227.0230 | 0.0000 |
| 19 | 86.1855 | 37.0869 | -9.4174 | [illegible] | 0.0000 | 59.7598 | -9.4174 | 13.3383 | 0.0000 | -0.1207 | 159.1630 | 0.0000 |

UNIFORM DEAD LOAD INTERVAL= 2446.161 KG/M
UNIFORM LIVE LOAD INTERVAL= 660.367 KG/M
CONCENTRATED LOAD INTERVAL= 8776.06 KG

GIRDER SHEAR-EQUIVALENT LOADING (TON)

| | DEAD-LOAD | LIVE-LOAD | | CONCEN.-LOAD | | TOTAL LL. | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 17.1290 | 7.3709 | -1.8717 | 6.5009 | 0.0000 | 13.8797 | -1.8717 | 3.0979 | -0.4178 | 0.0787 | 34.1853 | 0.0000 |
| 2 | 5.9772 | 4.7268 | -2.5895 | 4.0073 | 0.0000 | 9.5341 | -2.5895 | 2.1280 | -0.5780 | 0.0787 | 17.7179 | 0.0000 |
| 3 | -5.0975 | 2.7740 | -3.9986 | 3.2482 | 0.0000 | 6.0123 | -3.9986 | 1.3419 | -0.8925 | 0.0787 | 2.3334 | -9.9100 |
| 4 | -16.1722 | 1.4586 | -6.0451 | 1.8751 | -5.9785 | 3.3336 | -12.0236 | 0.7441 | -2.6837 | 0.0787 | 0.0000 | -30.8008 |
| 5 | -27.2469 | 0.6969 | -8.6452 | 0.7771 | -7.3416 | 1.4760 | -15.9869 | 0.3294 | -3.5682 | 0.0787 | 0.0000 | -46.7234 |
| 6 | -38.3987 | 0.3804 | -11.6906 | 0.2470 | -8.4377 | 0.6224 | -20.1283 | 0.1389 | -4.4926 | 0.0787 | 0.0000 | -62.9410 |
| 7 | 38.8384 | 13.1129 | -1.3463 | 7.4018 | 0.0000 | 20.5947 | -1.3463 | 4.0069 | -0.2619 | 0.0000 | 63.4403 | 0.0000 |
| 8 | 27.6868 | 10.0736 | -1.6689 | 6.4902 | -0.7942 | 16.5639 | -2.4632 | 3.2227 | -0.4792 | 0.0000 | 47.4734 | 0.0000 |
| 9 | 16.6121 | 7.4372 | -2.3944 | 5.3407 | -1.7858 | 12.7859 | -4.1802 | 2.4876 | -0.8133 | 0.0000 | 31.8857 | 0.0000 |
| 10 | 5.5374 | 5.2644 | -3.5835 | 4.1280 | 0.0000 | 9.4025 | -3.5835 | 1.8294 | -0.6972 | 0.0000 | 16.7693 | 0.0000 |
| 11 | -5.5373 | 3.5835 | -5.2644 | 2.9273 | -4.9946 | 6.5108 | -10.2591 | 1.2665 | -1.9960 | 0.0000 | 2.2403 | -17.7923 |
| 12 | -16.6120 | 2.3944 | -7.4372 | 1.7868 | -6.2053 | 4.1802 | -13.6425 | 0.8133 | -2.6543 | 0.0000 | 0.0000 | -32.9088 |
| 13 | -27.6867 | 1.6689 | -10.0737 | 0.8666 | -7.3468 | 2.5255 | -17.4205 | 0.4914 | -3.3894 | 0.0000 | 0.0000 | -48.4965 |
| 14 | -38.8394 | 1.3463 | -13.1129 | 0.1666 | -8.3384 | 2.2029 | -21.4514 | 0.4286 | -4.1736 | 0.0000 | 0.0000 | -64.4644 |
| 15 | 38.3987 | 11.6906 | -0.3804 | 7.4769 | 0.0000 | 19.1876 | -0.3804 | 4.2826 | -0.0849 | -0.0787 | 61.7902 | 0.0000 |
| 16 | 27.2469 | 8.6452 | -0.6969 | 6.4009 | -0.7791 | 15.0461 | -1.4760 | 3.3583 | -0.3294 | -0.0787 | 45.5727 | 0.0000 |
| 17 | 16.1722 | 6.0451 | -1.4586 | 5.0378 | 0.0000 | 11.0829 | -1.4586 | 2.4737 | -0.3256 | -0.0787 | 29.6501 | 0.0000 |
| 18 | 5.0975 | 3.9986 | -2.7740 | 3.4608 | 0.0000 | 7.4674 | -2.7740 | 1.6667 | -0.6172 | -0.0787 | 14.1530 | 0.0000 |
| 19 | -5.9772 | 2.5895 | -4.7268 | 1.7672 | 0.0000 | 4.3567 | -4.7268 | 0.9724 | -1.0550 | -0.0787 | 0.0000 | -11.8377 |
| 20 | -18.2824 | 1.8717 | -7.3709 | 0.9407 | 0.0000 | 2.8124 | -7.3709 | 0.6277 | -1.6452 | -0.0787 | 0.0000 | -27.3771 |

**********
TRUCK EXTERIOR GIRDER
**********

GIRDER BENDING MOMENT-TRUCK LOADING (TON-M)

| | DEAD-LOAD | HI-AXIAL LOAD | | LOW-AXIAL LOAD | | TOTAL LL. | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 88.3459 | 3.5825 | -0.5184 | 1.0108 | -0.2572 | 62.5954 | -9.3240 | 13.9712 | 0.0000 | -0.1207 | 164.7918 | 0.0000 |
| 2 | 119.1744 | 5.2891 | -1.0369 | 2.6301 | -0.5144 | 95.1463 | -18.6481 | 21.2365 | 0.0000 | -0.2413 | 235.3159 | 0.0000 |
| 3 | 92.8832 | 5.3425 | -1.5553 | 2.6612 | -0.7717 | 96.1244 | -27.9721 | 21.4548 | 0.0000 | -0.3620 | 210.1004 | 0.0000 |
| 4 | 9.4722 | 4.1246 | -3.5433 | 2.0493 | -1.7556 | 74.1900 | -63.7153 | 16.5591 | -13.2463 | -0.4826 | 99.7388 | -67.9720 |
| 5 | -131.0585 | 2.1364 | -5.5612 | 1.0573 | -2.7560 | 38.4111 | -100.0039 | 8.5733 | -20.7907 | -0.6033 | 0.0000 | -252.4563 |
| 6 | -329.1071 | 0.8008 | -7.5792 | 0.3971 | -3.7564 | 14.4096 | -136.2925 | 3.2162 | -28.3350 | -0.7239 | 0.0000 | -494.4564 |
| 7 | -128.8393 | 2.3237 | -2.5050 | 1.1475 | -1.2485 | 41.7764 | -45.0739 | 8.1281 | -9.3708 | -0.7239 | 0.0000 | -184.0079 |
| 8 | 13.9252 | 4.3381 | -2.0327 | 2.1508 | -1.0131 | 78.0326 | -36.5762 | 15.1821 | -7.6041 | -0.7239 | 106.4160 | -30.9791 |
| 9 | 99.5841 | 5.7161 | -1.5605 | 2.8105 | -0.7777 | 102.7000 | -28.0786 | 19.9815 | -5.8375 | -0.7239 | 221.5416 | 0.0000 |
| 10 | 128.1374 | 4.1070 | -1.4327 | 2.0162 | -0.6798 | 73.7764 | -25.6427 | 14.3540 | 0.0000 | -0.7239 | 215.5439 | 0.0000 |
| 11 | 99.5852 | 2.4978 | -1.5605 | 1.2619 | -0.7777 | 44.8528 | -28.0785 | 8.7266 | -5.8375 | -0.7239 | 152.4407 | 0.0000 |
| 12 | 13.9274 | 1.8831 | -2.0327 | 0.9303 | -1.0131 | 33.8502 | -36.5761 | 6.5859 | -7.6041 | -0.7239 | 53.6396 | -30.9788 |
| 13 | -128.8360 | 2.3175 | -2.5050 | 1.1311 | -1.2485 | 41.6049 | -45.0738 | 8.0947 | -9.3708 | -0.7239 | 0.0000 | -184.0045 |
| 14 | -329.1078 | 0.8008 | -4.1491 | 0.3971 | -2.0509 | 14.4097 | -74.5899 | 2.8036 | -15.5071 | -0.7239 | 0.0000 | -419.9288 |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 15 | -131.0594 | 2.1364 | -3.4576 | 1.0573 | -1.7091 | 38.4111 | -62.1584 | 8.5733 | -12.9226 | -0.6033 | 0.0000 | -706.7437 |
| 16 | 9.4716 | 4.1246 | -2.7661 | 2.0473 | -1.3673 | 74.1901 | -49.7267 | 16.5591 | -10.3381 | -0.4826 | 99.7381 | -51.0759 |
| 17 | 92.8828 | 5.3425 | -2.0746 | 2.6612 | -1.0255 | 96.1244 | -37.2951 | 21.4548 | 0.0000 | -0.3620 | 210.1000 | 0.0000 |
| 18 | 119.1743 | 5.2891 | -1.3830 | 2.6301 | -0.6836 | 95.1464 | -24.8634 | 21.2365 | 0.0000 | -0.2413 | 235.3159 | 0.0000 |
| 19 | 88.3460 | 3.5825 | -0.6915 | 1.3018 | -0.3418 | 62.5954 | -12.4317 | 13.9712 | 0.0000 | -0.1207 | 164.7920 | 0.0000 |

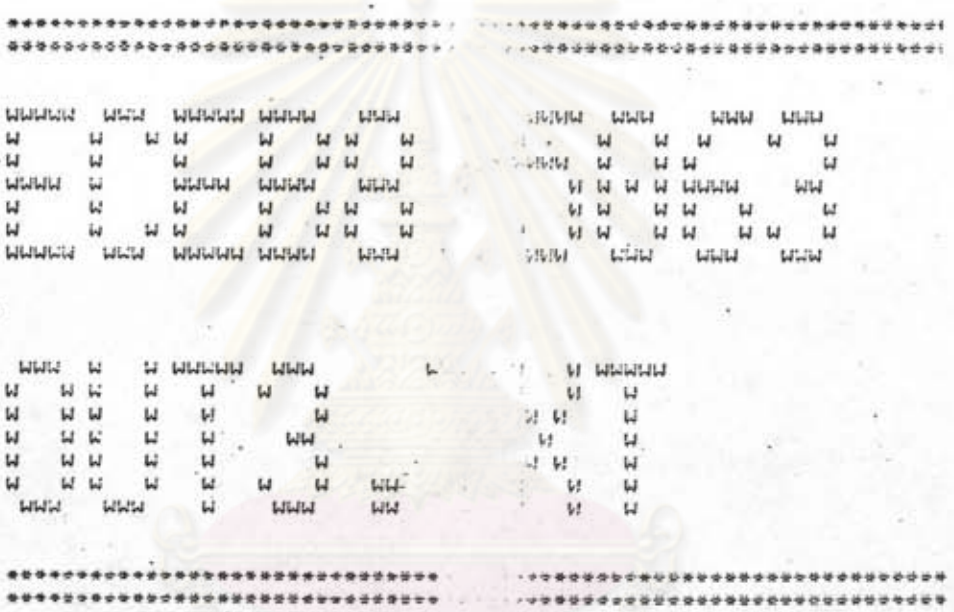
GIRDER SHEAR-TRUCK LOADING (TON)

| | DEAD-LOAD | HI-AXIAL LOAD | | LOW-AXIAL LOAD | | TOTAL LL. | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 17.5384 | 0.7729 | -0.1030 | 0.3697 | -0.0495 | 13.8971 | -1.8466 | 3.1018 | -0.4122 | 0.0787 | 34.6359 | 0.0000 |
| 2 | 6.1270 | 0.5256 | -0.1974 | 0.2578 | -0.0978 | 9.4406 | -3.5501 | 2.1071 | -0.7924 | 0.0787 | 17.7534 | 0.0000 |
| 3 | -5.2253 | 0.3538 | -0.3788 | 0.1717 | -0.1843 | 6.3471 | -6.7973 | 1.4167 | -1.5172 | 0.0787 | 2.6171 | -13.4611 |
| 4 | -16.5776 | 0.2046 | -0.5509 | 0.0968 | -0.2701 | 3.6606 | -9.8943 | 0.8170 | -2.2084 | 0.0787 | 0.0000 | -28.6017 |
| 5 | -27.9299 | 0.1561 | -0.7005 | 0.0766 | -0.3446 | 2.8032 | -12.5858 | 0.6257 | -2.8091 | 0.0787 | 0.0000 | -43.2462 |
| 6 | -39.3613 | 0.1561 | -0.8208 | 0.0766 | -0.4044 | 2.8032 | -14.7510 | 0.6257 | -3.2924 | 0.0787 | 0.0000 | -57.3761 |
| 7 | 39.8122 | 0.8193 | -0.0939 | 0.4036 | -0.0468 | 14.7227 | -1.6893 | 2.8645 | -0.3287 | 0.0000 | 57.3994 | 0.0000 |
| 8 | 28.3809 | 0.7460 | -0.0939 | 0.3700 | -0.0468 | 13.4152 | -1.6893 | 2.6101 | -0.3287 | 0.0000 | 44.4061 | 0.0000 |
| 9 | 17.0286 | 0.5817 | -0.1949 | 0.2870 | -0.0917 | 10.4544 | -3.4850 | 2.0340 | -0.6780 | 0.0000 | 29.5170 | 0.0000 |
| 10 | 5.6762 | 0.0939 | -0.3646 | 0.0468 | -0.1801 | 1.6893 | -6.5535 | 0.3287 | -1.2751 | 0.0000 | 7.6942 | -2.1523 |
| 11 | -5.6761 | 0.0939 | -0.3646 | 0.0468 | -0.1801 | 1.6893 | -6.5535 | 0.3287 | -1.2751 | 0.0000 | 0.0000 | -13.5046 |
| 12 | -17.0284 | 0.0939 | -0.3646 | 0.0468 | -0.1801 | 1.6893 | -6.5535 | 0.3287 | -1.2751 | 0.0000 | 0.0000 | -24.8369 |
| 13 | -28.3807 | 0.0939 | -0.3646 | 0.0468 | -0.1801 | 1.6893 | -6.5535 | 0.3287 | -1.2751 | 0.0000 | 0.0000 | -36.2092 |
| 14 | -39.8130 | 0.0939 | -0.8166 | 0.0468 | -0.4036 | 1.6893 | -14.6791 | 0.3287 | -2.8560 | 0.0000 | 0.0000 | -57.3481 |
| 15 | 39.3613 | 0.8208 | -0.1561 | 0.4044 | -0.0768 | 14.7510 | -2.8040 | 3.2924 | -0.6259 | -0.0787 | 57.3260 | 0.0000 |
| 16 | 27.9299 | 0.7005 | -0.1561 | 0.3446 | -0.0768 | 12.5858 | -2.8040 | 2.8091 | -0.6259 | -0.0787 | 43.2462 | 0.0000 |
| 17 | 16.5776 | 0.5509 | -0.2046 | 0.2701 | -0.0968 | 9.8943 | -3.6606 | 2.2084 | -0.8170 | -0.0787 | 28.6017 | 0.0000 |
| 18 | 5.2253 | 0.3788 | -0.3538 | 0.1863 | -0.1717 | 6.7973 | -6.3471 | 1.5172 | -1.4167 | -0.0787 | 13.4611 | -2.6171 |
| 19 | -6.1270 | 0.1922 | -0.5256 | 0.0971 | -0.2578 | 3.4394 | -9.4406 | 0.7677 | -2.1071 | -0.0787 | 0.0000 | -17.7534 |
| 20 | -18.7407 | 0.0772 | -0.7120 | 0.0368 | -0.3911 | 1.3820 | -12.7965 | 0.3085 | -2.8562 | -0.0787 | 0.0000 | -34.4720 |

**********
TRUCK INTERIOR GIRDER
**********

GIRDER BENDING MOMENT-TRUCK LOADING (TON-M)

| | DEAD-LOAD | HI-AXIAL LOAD | | LOW-AXIAL LOAD | | TOTAL LL. | | IMPACT LOAD | | SETTLE | TOTAL | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 86.1854 | 3.5825 | -0.5184 | 1.3018 | -0.2572 | 62.5954 | -9.3240 | 13.9712 | 0.0000 | -0.1207 | 162.6313 | 0.0000 |
| 2 | 116.2599 | 5.2891 | -1.0369 | 2.6301 | -0.5144 | 95.1463 | -18.6481 | 21.2365 | 0.0000 | -0.2413 | 232.4015 | 0.0000 |
| 3 | 90.6117 | 5.3425 | -1.5553 | 2.6612 | -0.7717 | 96.1244 | -27.9721 | 21.4548 | 0.0000 | -0.3620 | 207.8289 | 0.0000 |
| 4 | 9.2406 | 4.1246 | -3.5433 | 2.0473 | -1.7556 | 74.1900 | -63.7153 | 16.5591 | -13.2463 | -0.4826 | 99.5071 | -68.2036 |
| 5 | -127.8534 | 2.1364 | -5.5612 | 1.0573 | -2.7560 | 38.4111 | -100.0039 | 8.5733 | -20.7907 | -0.6033 | 0.0000 | -249.2312 |
| 6 | -321.0588 | 0.8008 | -7.5792 | 0.3971 | -3.7564 | 14.4096 | -136.2925 | 3.2162 | -28.3350 | -0.7239 | 0.0000 | -456.4102 |
| 7 | -125.6885 | 2.3237 | -2.5050 | 1.1475 | -1.2485 | 41.7764 | -45.0739 | 8.1281 | -9.3708 | -0.7239 | 0.0000 | -180.8971 |
| 8 | 13.5846 | 4.3381 | -2.0327 | 2.1508 | -1.0131 | 78.0326 | -36.5762 | 15.1821 | -7.6041 | -0.7239 | 106.0734 | -31.3147 |
| 9 | 97.1487 | 5.7161 | -1.5605 | 2.8105 | -0.7777 | 102.7000 | -28.0786 | 19.9815 | -5.6375 | -0.7239 | 219.1063 | 0.0000 |
| 10 | 125.0038 | 4.1070 | -1.4327 | 2.0162 | -0.6798 | 73.7764 | -25.6427 | 14.3540 | 0.0000 | -0.7239 | 212.4103 | 0.0000 |
| 11 | 97.1499 | 2.4978 | -1.5605 | 1.2019 | -0.7777 | 44.8528 | -28.0785 | 8.7266 | -5.6375 | -0.7239 | 150.0053 | 0.0000 |
| 12 | 13.5868 | 1.8831 | -2.0327 | 0.9203 | -1.0131 | 33.8502 | -36.5761 | 6.5859 | -7.6041 | -0.7239 | 53.2990 | -31.3174 |
| 13 | -125.6853 | 2.3175 | -2.5050 | 1.1301 | -1.2485 | 41.6049 | -45.0738 | 8.0947 | -9.3708 | -0.7239 | 0.0000 | -180.8939 |



**********
********** BEAM DESIGN **********
**********

**********
EXTERIOR DESIGN
**********

Span # 1

| Section No. | Distance (m.) | Depth (m.) | Top Bar # 11 | Bottom Bar # 11 | Stirrub # 4 (m.) |
|---|---|---|---|---|---|
| 1 | 0.00 | 2.00 | 2 | 2 | 0.40 |
| 2 | 5.03 | 1.81 | 2 | 5 | 37.33 |
| 3 | 10.06 | 1.80 | 2 | 7 | 0.89 |
| 4 | 15.09 | 1.80 | 2 | 7 | 0.40 |
| 5 | 20.12 | 1.80 | 2 | 3 | 0.18 |
| 6 | 25.15 | 1.81 | 7 | 7 | 0.12 |
| 7 | 30.18 | 2.00 | 16 | 16 | |

Span # 2

| Section No. | Distance (m.) | Depth (m.) | Top Bar # 11 | Bottom Bar # 11 | Stirrub # 4 (m.) |
|---|---|---|---|---|---|
| 1 | 0.00 | 2.00 | 16 | 16 | 0.14 |
| 2 | 5.03 | 1.81 | 7 | 5 | 0.18 |
| 3 | 10.06 | 1.80 | 2 | 3 | 0.37 |
| 4 | 15.09 | 1.80 | 2 | 7 | 0.89 |
| 5 | 20.12 | 1.80 | 2 | 9 | 18.33 |
| 6 | 25.15 | 1.80 | 2 | 7 | 0.35 |
| 7 | 30.18 | 1.80 | 2 | 3 | 0.17 |
| 8 | 35.21 | 1.81 | 7 | 5 | |

| | | | | | |
|---|---|---|---|---|---|
| 9 | 40.24 | 2.00 | 16 | 16 | 0.12 |

Span # 3

| Section No. | Distance (m.) | Depth (m.) | Top Bar # 11 | Bottom Bar # 11 | Stirrub # 4 (m.) |
|---|---|---|---|---|---|
| 1 | 0.00 | 2.00 | 16 | 16 | 0.14 |
| 2 | 5.03 | 1.81 | 7 | 5 | 0.19 |
| 3 | 10.06 | 1.80 | 2 | 3 | 0.43 |
| 4 | 15.09 | 1.80 | 2 | 7 | 0.89 |
| 5 | 20.12 | 1.80 | 2 | 7 | 34.25 |
| 6 | 25.15 | 1.81 | 2 | 5 | 0.33 |
| 7 | 30.18 | 2.00 | 2 | 2 | |

**********
INTERIOR DESIGN
**********

Span # 1

| Section No. | Distance (m.) | Depth (m.) | Top Bar # 11 | Bottom Bar # 11 | Stirrub # 4 (m.) |
|---|---|---|---|---|---|
| 1 | 0.00 | 2.00 | 2 | 2 | 0.45 |
| 2 | 5.03 | 1.81 | 2 | 5 | 0.90 |
| 3 | 10.06 | 1.80 | 2 | 7 | 0.89 |
| 4 | 15.09 | 1.80 | 2 | 6 | 0.45 |
| 5 | 20.12 | 1.80 | 2 | 3 | 0.20 |
| 6 | 25.15 | 1.81 | 6 | 7 | 0.13 |
| 7 | 30.18 | 2.00 | 15 | 15 | |

Span # 2

| Section No. | Distance (m.) | Depth (m.) | Top Bar # 11 | Bottom Bar # 11 | Stirrub # 4 (m.) |
|---|---|---|---|---|---|
| 1 | 0.00 | 2.00 | 15 | 15 | |
| 2 | 5.03 | 1.81 | 7 | 4 | 0.15 |
| 3 | 10.06 | 1.80 | 2 | 3 | 0.20 |
| 4 | 15.09 | 1.80 | 2 | 7 | 0.41 |
| 5 | 20.12 | 1.80 | 2 | 8 | 0.89 |
| 6 | 25.15 | 1.80 | 2 | 7 | 0.89 |
| 7 | 30.18 | 1.80 | 2 | 3 | 0.39 |
| 8 | 35.21 | 1.81 | 7 | 4 | 0.19 |
| 9 | 40.24 | 2.00 | 15 | 15 | 0.12 |

Span # 3

| Section No. | Distance (m.) | Depth (m.) | Top Bar # 11 | Bottom Bar # 11 | Stirrub # 4 (m.) |
|---|---|---|---|---|---|
| 1 | 0.00 | 2.00 | 15 | 15 | |
| 2 | 5.03 | 1.81 | 7 | 5 | 0.15 |
| 3 | 10.06 | 1.80 | 2 | 3 | 0.21 |
| 4 | 15.09 | 1.80 | 2 | 6 | 0.49 |
| 5 | 20.12 | 1.80 | 2 | 7 | 0.89 |
| 6 | 25.15 | 1.81 | 2 | 5 | 0.89 |
| 7 | 30.18 | 2.00 | 2 | 2 | 0.36 |

QUANTITY OF CONCRETE (cu. m)

| Slab (cu. m) | Girder (cu. m) | Total (cu. m) |
|---|---|---|
| 153. 81 | 221. 38 | 375. 19 |

QUANTITY OF FORMWORK (sq. m)

| Slab (sq. m) | Girder (sq. m) | Total (sq. m) |
|---|---|---|
| 780. 12 | 1596. 57 | 2376. 70 |

QUANTITY OF STEEL (ton)

| Slab (ton) | Girder (ton) | Total (ton) |
|---|---|---|
| 1. 93 | 43. 85 | 45. 78 |

DEFLECTION EACH SPAN (MIDDLE SPAN)(CM.)

---

DEF. FOR SPAN 1 = 2.887 CM.
DEF. FOR SPAN 2 = 7.172 CM.
DEF. FOR SPAN 3 = 2.887 CM.

---

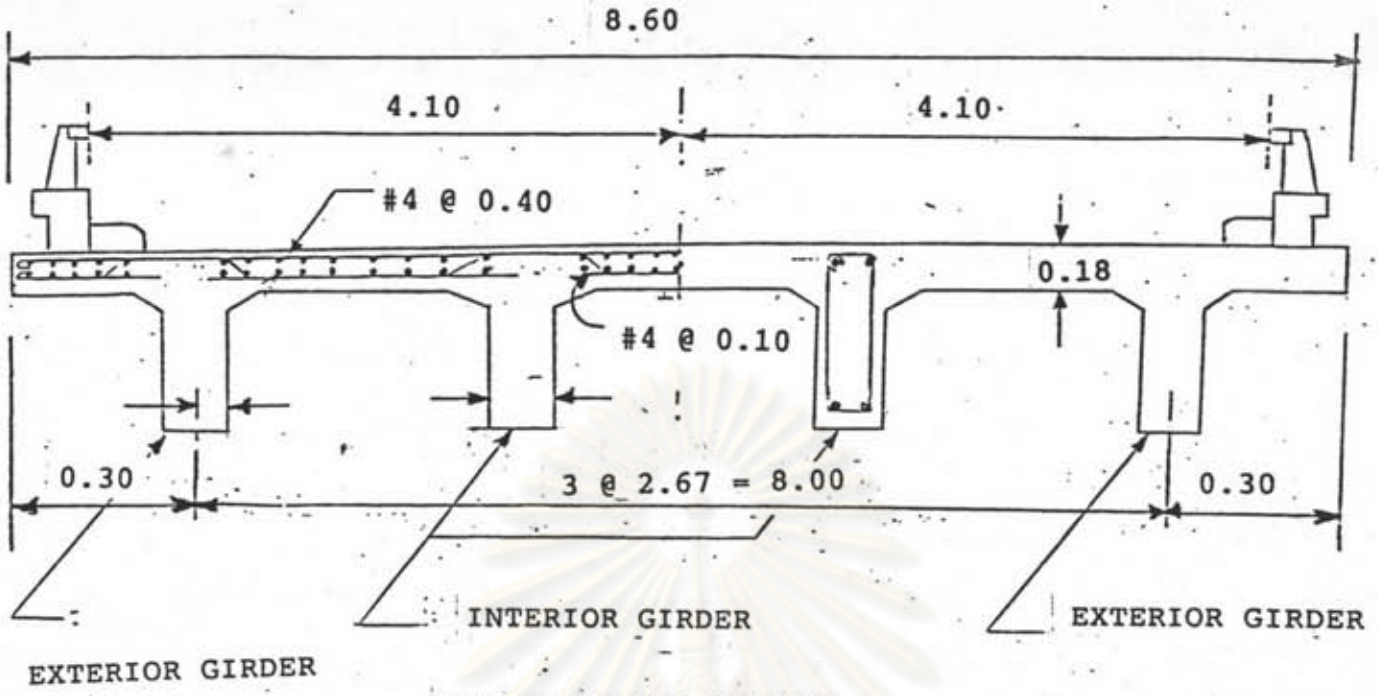
---

REATION AT SUPPORT (TON.)

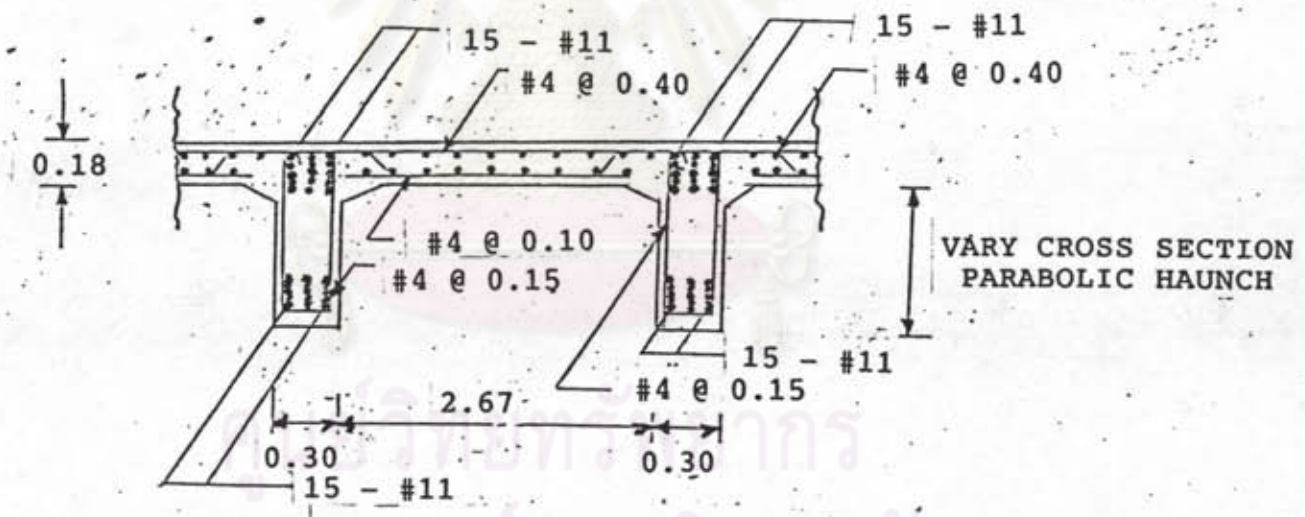
---

REACTION AT SUPPORT NO. 1= 140 TON.
REACTION AT SUPPORT NO. 2= 520 TON.
REACTION AT SUPPORT NO. 3= 520 TON.
REACTION AT SUPPORT NO. 4= 139 TON.

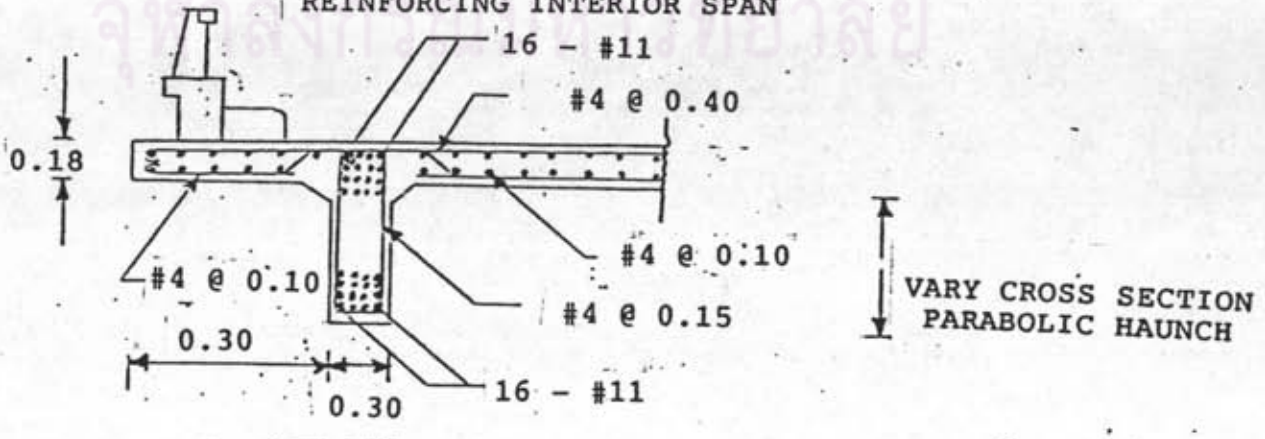
---




TYPICAL CROSS SECTION




REINFORCING INTERIOR SPAN




REINFORCING EXTERIOR SPAN

# ประวัติผู้เขียน

นายชาญวิทย์ อาจสมิติ เกิดวันที่ 15 ตุลาคม พ.ศ. 2504 ที่อำเภอลาดบัวหลวง จังหวัดพระนครศรีอยุธยา สำเร็จการศึกษา ระดับปริญญาตรีวิศวกรรมศาสตร์บัณฑิต สาขาวิศกรรมโยธา คณะวิศวกรรมศาสตร์ จากมหาวิทยาลัยขอนแก่น เมื่อปีการศึกษา 2527