```fortran
C     ================================================================
C     SUBROUTINE RESPON_FEA        :       CALCULATE TOATAL MASS
c                                          CALCULATE MAXIMUM von Mises stress
c                                          for current gauss point
C
C     ================================================================
c     Copyright (c) 2010-2014 [Xuchun Ren], All Rights Reserved.




      SUBROUTINE RESPON_FEA(X0,Y,IFLAG_Y)
      USE DFLIB
c      use msflib
      use IFPORT
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C      IMPLICIT NONE


      REAL*8 X(20),Z(20),XMU(20),SIG(20), NMU(20),NSIG(20),DX(20),
     1  X0(20)
      REAL*8 Y
      INTEGER NYQ, N,IITER, IITER1, IITER_TOTAL,I_SQP
        COMMON /SR02/ XMU,SIG, NMU, NSIG, NYQ, N
        COMMON /XR01/ N_F, N_GF, N_GC, N_EF,N_EC
      INTEGER N_PREFEA, N_FEA, NO_GAUSS
      COMMON /XR05/ IITER, IITER1, N_PREFEA, N_FEA, IITER_TOTAL, I_SQP
      COMMON /XR06/ NO_GAUSS
      COMMON /XR09/ I_EVAL


        INTEGER(2) n1, n2, n3, n4, n5, n6, n7, n8,n9, status,
     1   MESH_ALGORITHM,M
        CHARACTER(320) buf, FND,FNP,FN_INP, FN_HM, FN_MASS, FN_TCL,
     1  FN_BAT, FN_BAT0, FN_INP0, FN_HM0, FN_MASS0, FN_PY0, FN_PY,
     2  FN_ODB, FN_ODB0, FN_MISES, FN_MISES0, JOBNAME, FUNC_NAME
        CHARACTER(320) MSG,MSG1,IDR,GDR, ELSETNAME, CTMP,CTMP1,CTMP2,
     1  IDR0
        CHARACTER(320) PATH, NAME, EXE_HM, EXE_ABAQUS
        CHARACTER(320) WORKFOLDER
        CHARACTER*320 CMD_TCL, CMD_PY, CMD_DOS
        LOGICAL ex
        LOGICAL(4) l_result
        REAL*8 TOTAL_MASS, MAX_MISES
        REAL*8 A(20), RTEMP, RTEMP2
        INTEGER  N_RENUM, N_INP_OK, N_ODB_OK, N_INP_BEST
        COMMON /XR07/ PATH, WORKFOLDER
C         MSG = 'PROGRAM NBAS(<10) NGAUSS(<10) NVAR_PDD(1,2,3) FN
C     1   IBOX(0,1,2,...) '
c        MSG1 = 'IID = 0(INDEPENDENT), 1(IID), 2(DEPENDNENT) '

        N_INP_OK = 0
        N_ODN_OK = 0
        N_INP_BEST = 0
        PSTRESS = 800.0
c        PSTRESS = 2200.0
        MSG = 'PROGRAM PATH NAME'
        FN_TCL = "3hole_bracket-17.tcl"
        FN_BAT0 = "RUN_TCL.BAT"
        EXE_HM = "C:\Altairwin64\hw9.1\hm\bin\win64\hmbatch.exe -tcl "
        EXE_ABAQUS = "abaqus  cae noGUI="
        FN_INP0 = '3H_BRACKET.INP'
        FN_HM0 = '3H_BRACKET.hm'
        FN_MASS0 = '3HB_MASS.DAT'
        FN_PY = 'submit-07.py'
        FN_PY0 = 'submit-07.py'
        ELSETNAME = 'ELEMENTS4STRESS'
c$$$        FN_ODB = '3H_BRACKET.ODB'
c$$$        FN_ODB0 = '3H_BRACKET.ODB'
        FN_MISES = 'maxMises.DAT'
        FN_MISES0 = 'maxMises.DAT'

        n1 = 1
        n2 = 2
        PI = 3.1415926
      Z = 0.0
      DO I = 1,N
      Z(I) = NMU(I) + NSIG(I)*X0(I)
      ENDDO
C  -----------   CALCULATE TRUNCATED DISTANCE FROM NMU AND NSIG  -------
      A = 0.0D0

      A(1) = (NMU(2)-NMU(1)-1.0)/2.0
      RTEMP = (NMU(7) - NMU(2)-2.0)/2.0
      RTEMP2 = (NMU(4) - NMU(2) - 2.0)/2.0
      A(2) = MIN(RTEMP,A(1), RTEMP2)
      IF (A(2) .LE. 0.4) A(2) = 0.4
      RTEMP = (NMU(3)-2.0)/2.0
      RTEMP2 = 30.0-NMU(3)-2.0
      A(3) =  MIN(RTEMP, RTEMP2)
      RTEMP = (NMU(7)-NMU(4)-2.0)/2.0
      RTEMP2 = (NMU(4)-NMU(1)-2.0)/2.0
      A(4) = MIN(RTEMP, RTEMP2)
      A(5) = (NMU(5)-11.0)/2.0
      A(6) = (NMU(6)-11.0)/2.0
      RTEMP = (NMU(7)-NMU(4)-2.0)/2.0
      RTEMP2 = (150.0 - NMU(7)-5.0)/2.0
      A(7) = MIN(RTEMP, RTEMP2)

      RTEMP = (27.57 + NMU(8) - NMU(9)-2.0)/2.0
c      RTEMP2 = (150.0 - NMU(7)-5.0)/2.0
      A(8) = RTEMP
      IF (A(8) .LE. 0.4) A(8) = 0.4
      RTEMP = (27.57 + NMU(8) - NMU(9)-2.0)/2.0
      RTEMP2 = (14.912 + NMU(9)-2.0)/2.0
      A(9) =  MIN(RTEMP, RTEMP2)
      IF (A(9) .LE. 0.4) A(9) = 0.4

      DO I = 1,N
      IF (A(I) .LE. 0.4) THEN
      PRINT*, 'A(I) < 0.2, I = ', I
      PRINT*, 'A(I) = ', A(I)
      PRINT*, 'NMU(I) = ', NMU(I)
      A(I) = 0.4
      ENDIF
      ENDDO


c      A = 0.6

      CALL ToTruncatedG(Z, A, X)

        IF (IFLAG_Y .EQ. 1) THEN
        FUNC_NAME = 'OBJ'
        ELSEIF (IFLAG_Y .EQ. 2) THEN
        FUNC_NAME = 'CONST'
        ELSE
        PRINT*, 'WRONG IFLAG_Y!'
        STOP
        ENDIF

      WRITE(IDR0,"(T1, I10)") I_SQP
      WRITE(IDR,"(T1, I10)") IITER
      IDR0 = ADJUSTL(IDR0)
      IDR = ADJUSTL(IDR)
      IDR = TRIM(WORKFOLDER)//'\'//TRIM(IDR)//'-'//TRIM(IDR0)
      INQUIRE (DIRECTORY = TRIM(IDR), EXIST=ex)
      IF (.NOT. ex) THEN
      l_result = MAKEDIRQQ (TRIM(IDR))
      IF (.NOT. l_result) THEN
         PRINT*, 'FAIL TO CREATE FOLDER :', TRIM(IDR)
         STOP
      ENDIF
      ENDIF

        WRITE(GDR,*) I_EVAL
        WRITE(CTMP,*) NO_GAUSS
        GDR = ADJUSTL(GDR)
        GDR = TRIM(IDR)//'\'//TRIM(GDR)//'-'//TRIM(FUNC_NAME)

        INQUIRE (DIRECTORY = TRIM(GDR), EXIST=ex)
        IF (.NOT. ex) THEN
          l_result = MAKEDIRQQ (TRIM(GDR))
          IF (.NOT. l_result) THEN
             PRINT*, 'FAIL TO CREATE FOLDER :', TRIM(GDR)
             STOP
          ENDIF

        ENDIF
        GDR = TRIM(GDR)//'\'//TRIM(ADJUSTL(CTMP))
        INQUIRE (DIRECTORY = TRIM(GDR), EXIST=ex)
        IF (.NOT. ex) THEN
          l_result = MAKEDIRQQ (TRIM(GDR))
          IF (.NOT. l_result) THEN
             PRINT*, 'FAIL TO CREATE FOLDER :', TRIM(GDR)
             STOP
          ENDIF

        ENDIF

C----------  COPY TCL FILE   -----------
        CMD_TCL = "copy "//TRIM(PATH)//'\'//TRIM(FN_TCL)//" "
     1   //TRIM(GDR)
        l_result = systemqq (TRIM(CMD_TCL))
        IF (.NOT. l_result) THEN
           PRINT*, 'FAIL TO RUN COMMAND: ', TRIM(CMD_TCL)
           STOP
        ENDIF
C----------  CREATE DESIGN.DAT FILE   -----------



        M = 0
201     FND =  TRIM(GDR)//'\'//'DESIGN.DAT'
        IF (M .GE. 1) THEN
        WRITE(CTMP,*) M-1
        CTMP = TRIM(GDR)//'\'//'DESIGN'//TRIM(ADJUSTL(CTMP))//'.DAT'
        CMD_DOS = 'move '//TRIM(FND)//' '//TRIM(CTMP)
        l_result = systemqq (TRIM(CMD_DOS))

        IF (.NOT. l_result) THEN
           PRINT*, 'FAIL TO RUN COMMAND: ', TRIM(CMD_DOS)
           STOP
        ENDIF
        ENDIF
        OPEN(201,FILE=trim(FND),STATUS='NEW')
        DO J = 1,9
           WRITE(CTMP,*) X(J)
           WRITE(201,'(A)')  TRIM(ADJUSTL(CTMP))
        ENDDO
        M = M+1
        FN_INP = TRIM(GDR)//'\'//TRIM(FN_INP0)
        FN_HM = TRIM(GDR)//'\'//TRIM(FN_HM0)
        FN_MASS = TRIM(GDR)//'\'//TRIM(FN_MASS0)
        WRITE(201,'(A)')  TRIM(ADJUSTL(FN_INP))
        WRITE(201,'(A)')  TRIM(ADJUSTL(FN_HM))
        WRITE(201,'(A)')  TRIM(ADJUSTL(FN_MASS))
        T1 = X(2)-X(1)
        T2 = (X(5)-10.0)
        T3 = (X(6)-10.0)
        T4 = (27.57+X(8)-X(9))
        T5 = (150.0-X(7))
        DENSITY_MESH = MIN(T1/2.0,T2/2.0,
     1   T3/2.0, T4/4.0, T5/2.0, 2.0)
        DENSITY_MESH = MAX(DENSITY_MESH, 2.0)
        MESH_ALGORITHM = 0
        IF (M .GE. 2 .AND. M .LE. 4) THEN
            PRINT *, CHAR(7)
            CALL SLEEP(2)
            IF (DENSITY_MESH .LE. 1.0 .AND.
     1       (T4 .GE. 0 .AND. T1 .GE. 2.0 .AND. T2 .GE. 2.0
     2        .AND. T3 .GE. 2.0 .AND. T5 .GE. 2.0)) THEN
                DENSITY_MESH = DENSITY_MESH * (PI/3.0D0 + 0.4*M)
            ELSE
                DENSITY_MESH = DENSITY_MESH / (PI/3.0D0 + 0.1*M)
            ENDIF
        ELSE IF (M .GE. 5) THEN
            MESH_ALGORITHM = 1

            PRINT*, 'FAIL TO MESH BY CHANGE MESHSIZE, TRYING TO
     1    CHANGE ALGORITHM TO MIXED ....'
        ENDIF


        WRITE(201,'(F10.6)') DENSITY_MESH
        WRITE(201,'(I8)') MESH_ALGORITHM

        CLOSE(201)
C----------  RUN TCL FILE   -----------
C        FN_BAT = TRIM(GDR)//'\'//TRIM(FN_BAT0)
C        OPEN(202,FILE=trim(FN_BAT),STATUS='NEW')
C        WRITE(202,*), "cd "//TRIM(GDR)
C        WRITE(202,*), TRIM(EXE_HM)//TRIM(GDR)//'\'//
C     1   TRIM(FN_TCL)
C        CLOSE(202)
        CMD_TCL = ADJUSTL(GDR(1:2))//" & "//"cd "//TRIM(GDR)//" & "//
     1   TRIM(EXE_HM)//" -tcl "//TRIM(FN_TCL)
        l_result = systemqq (TRIM(CMD_TCL))
        IF (.NOT. l_result) THEN
           PRINT*, 'FAIL TO RUN COMMAND: ', TRIM(CMD_TCL)
           STOP
        ENDIF
        INQUIRE (FILE = TRIM(FN_INP), EXIST=ex)
        IF (.NOT. ex) THEN
            IF (M .GE. 5) THEN
            STOP
            ELSE
            PRINT*, 'FAIL TO GENERATE INP FILE: ', TRIM(FN_INP)
            PRINT*, "CHANGE MESH SIZE, TRYING .... "
            GOTO 201
            ENDIF

         ENDIF
c     ----------   get mass  ------------------
      N_INP_OK = 1
      OPEN(204,FILE=TRIM(FN_MASS),STATUS='OLD')
      READ(204,'(F16.8)') TOTAL_MASS
      CLOSE(204)
      PRINT*, '---------------------------TOTAL MASS  --------------'
      PRINT*, TOTAL_MASS

      N_PREFEA = N_PREFEA + 1

      IF (IFLAG_Y .EQ. 1) THEN
        Y = TOTAL_MASS
        RETURN
      ENDIF

C----------  COPY PYTHON FILE   -----------
        CMD_PY = "copy "//TRIM(PATH)//'\'//TRIM(FN_PY0)//" "
     1   //TRIM(GDR)
        l_result = systemqq (TRIM(CMD_PY))
        IF (.NOT. l_result) THEN
           PRINT*, 'FAIL TO RUN COMMAND: ', TRIM(CMD_PY)
           STOP
        ENDIF
C----------  CREATE PATH_X.DAT FILE   ----------

        FND = TRIM(GDR)//'\'//'PATH_X.DAT'

        INQUIRE (FILE = TRIM(FND), EXIST=ex)
        IF (N_INP_BEST == 1 .AND. ex) THEN
        CMD_PY = "del "//TRIM(FND)
        l_result = systemqq (TRIM(CMD_PY))
        IF (.NOT. l_result) THEN
           PRINT*, 'FAIL TO RUN COMMAND: ', TRIM(CMD_PY)
           STOP
        ENDIF
        ENDIF


        OPEN(203,FILE=trim(FND),STATUS='NEW')

C        WRITE(CTMP,*) I_SQP    ---- FOR AIJK
        WRITE(CTMP,*) IITER
        WRITE(CTMP1,*) I_EVAL
        JOBNAME = TRIM(ADJUSTL(CTMP))//'-'//TRIM(ADJUSTL(CTMP1))
        WRITE(203,'(A)') TRIM(JOBNAME)


        FN_ODB = TRIM(GDR)//'\'//TRIM(JOBNAME)//".ODB"
        FN_MISES = TRIM(GDR)//'\'//TRIM(FN_MISES0)
        WRITE(203,'(A)') TRIM(ADJUSTL(FN_INP))
        WRITE(203,'(A)') TRIM(ADJUSTL(FN_ODB))
        WRITE(203,'(A)') TRIM(ADJUSTL(FN_MISES))
        WRITE(203,'(A)') TRIM(ELSETNAME)
        CLOSE(203)

        INQUIRE (FILE = TRIM(FN_ODB), EXIST=ex)
        IF (N_INP_BEST == 1 .AND.  ex) THEN
        CMD_PY = "del "//TRIM(FN_ODB)
        l_result = systemqq (TRIM(CMD_PY))
        IF (.NOT. l_result) THEN
           PRINT*, 'FAIL TO RUN COMMAND: ', TRIM(CMD_PY)
           STOP
        ENDIF
        ENDIF
C----------  RUN PYTHON FILE   -----------
C        FN_BAT = TRIM(GDR)//'\'//TRIM(FN_BAT0)
C        OPEN(202,FILE=trim(FN_BAT),STATUS='NEW')
C        WRITE(202,*), "cd "//TRIM(GDR)
C        WRITE(202,*), TRIM(EXE_HM)//" -tcl "//TRIM(GDR)//'\'//
C     1   TRIM(FN_TCL)
C        CLOSE(202)
        CMD_PY = GDR(1:2)//" & "//"cd "//TRIM(GDR)//" & "//
     1   TRIM(EXE_ABAQUS)//TRIM(FN_PY)
        l_result = systemqq (TRIM(CMD_PY))
        IF (.NOT. l_result) THEN
           PRINT*, 'FAIL TO RUN COMMAND: ', TRIM(CMD_PY)
           STOP
        ENDIF
        INQUIRE (FILE = TRIM(FN_MISES), EXIST=ex)
        IF (.NOT. ex .AND. N_INP_BEST .EQ. 1) THEN
            PRINT*, 'FAIL TO CREAT MISES FILE: ', TRIM(FN_MISES)
            STOP
         ENDIF
        IF (.NOT. ex .AND. N_INP_BEST .EQ. 0) THEN
            M = 4
            N_INP_BEST = 1
            GO TO 201
         ENDIF
c     -------------  get max mises  ------------
      OPEN(205,FILE=TRIM(FN_MISES),STATUS='OLD')
      READ(205,'(F16.8)') MAX_MISES
      CLOSE(205)
      PRINT*, '---------------------------MAX MISES  --------------'
      PRINT*, MAX_MISES

      Y =  PSTRESS - MAX_MISES

      N_FEA = N_FEA + 1

      RETURN

      END
```