

```

    program glue
    parameter(nf=100000,nm=100000)
*****
* This program glues two files. From standard input it reads:
*   low frequency filename
*   high frequency filename
*   MINGLUE   MAXGLUE   FLAG
* MINGLUE and MAXGLUE are the lower and higher limit of the  glued part.
* FLAG tells the program whether or not, and how, to scale:
*  -1: Low frequency part is scaled to the high frequency part.
*   0: No rescaling takes place.
*   1: High frequency part is scaled to the low frequency part.
* The glued result file is flushed to standard output.
* The files may have different and irregular frequency intervals.
* The program cheques whether the sets of data points of both files
* include the region of overlap.
* In the overlapping region the smaller distance between points of both
* files taken on output. Everwhere else the original x-points are taken
* on output.
*****
    real lowmx,higmn,flow,fhig,x,y,w,s0,sx,sy,sxx,sxy,sw,al,bl,ah,bh
    *   ,xlow(nf),ylo(nf),xhig(nf),yhig(nf),xm(nm),ym(nm),wm(nm),dx,t
    integer i,it,ilast,nlow,nhig,nmerge,flag,mrgl,mrgh,mrg
    character*400 fillow,filhig

    read(*,'(a400)') fillow
    read(*,'(a400)') filhig
    read(*,*) higmn,lowmx,flag
    open(12,file=fillow)
    open(13,file=filhig)
    open(18,file='glue.log')
    mrg=0
    mrgl=0
    nlow=0
    s0=0
    sx=0
    sy=0
    sxy=0
    sxx=0
    do 10 i=1,nf
        read(12,*,END=11) x,y
        if (x.gt.lowmx) then
            goto 10
        else if (x.lt.higmn) then
            nlow=nlow+1
            xlow(nlow)=x
            ylow(nlow)=y
        else
            w=(lowmx-x)/(lowmx-higmn)
            s0=s0+w
            sx=sx+w*x
            sy=sy+w*y
            sxx=sxx+w*x*x
            sxy=sxy+w*x*y
            mrg=mrg+1
            mrgl=mrgl+1
            xm(mrg)=x
            ym(mrg)=y
    end do

```

```

        wm(mrg)=w
    endif
10  continue
11  close(12)
    if (mrg1.le.1) then
        write(18,*) 'Warning. Process stopped: '
        write(18,*) 'No. points in low frequency overlap region < 2'
        goto 61
    else
*****  we determine the weighted average of y and dy/dx
        al=(s0*sxy-sx*sy)/(s0*sxx-sx*sx)
        bl=(sy*sxx-sx*sxy)/(s0*sxx-sx*sx)
    endif

    mrgh=0
    nhig=0
    s0=0
    sx=0
    sy=0
    sxy=0
    sxx=0
    do 20 i=1,nf
        read(13,*,END=21) x,y
        if (x.lt.higmn) then
            goto 20
        else if (x.gt.lowmx) then
            nhig=nhig+1
            xhig(nhig)=x
            yhig(nhig)=y
        else
            w=(x-higmn)/(lowmx-higmn)
            s0=s0+w
            sx=sx+w*x
            sy=sy+w*y
            sxx=sxx+w*x*x
            sxy=sxy+w*x*y
            mrg=mrg+1
            mrgh=mrgh+1
            xm(mrg)=x
            ym(mrg)=y
            wm(mrg)=w
        endif
20  continue
21  close(13)
    if (mrgh.le.1) then
        write(18,*) 'Warning. Process stopped: '
        write(18,*) 'No. points in high frequency overlap region < 2'
        goto 61
    else
*****  we determine the weighted average of y and dy/dx
        ah=(s0*sxy-sx*sy)/(s0*sxx-sx*sx)
        bh=(sy*sxx-sx*sxy)/(s0*sxx-sx*sx)
    endif

    bl=bl+(lowmx+higmn)*al*0.5
    bh=bh+(lowmx+higmn)*ah*0.5
    write(18,*) 'low/hig = ',bl/bh
    write(18,*) 'ratio (al*bh)/(ah*bl) ',(al*bh)/(ah*bl)

```

```

if (flag.eq.1) then
  fhig=bl/bh
  flow=1.
else if (flag.eq.-1) then
  fhig=1.
  flow=bh/bl
else
  fhig=1.
  flow=1.
endif

do 30 i=1,mrgl
  ym(i)=ym(i)*flow
  wm(i)=wm(i)/mrgl
30 continue
do 35 i=mrgl+1,mrg
  ym(i)=ym(i)*fhig
  wm(i)=wm(i)/mrgh
35 continue

call sort3(mrg,xm,ym,wm)

do 40 i=1,nlow
  write(*,*) xlow(i),ylow(i)*flow
40 continue

*****lump to dx intervals
nmerge=max0(mrgl,mrgh)
dx=(lowmx-higmn)/nmerge
ilast=1
do 55 it=0,nmerge
  t=higmn+it*dx
  sy=0
  sx=0
  sw=0
  do 50 i=ilast,mrg
    if (xm(i).gt.(t+dx)) goto 51
    sy=sy+ym(i)*wm(i)
    sx=sx+xm(i)*wm(i)
    sw=sw+wm(i)
50 continue
51 ilast=i
  if (sw.le.0) goto 55
  write(*,*) sx/sw,sy/sw
55 continue

do 60 i=1,nhig
  write(*,*) xhig(i),yhig(i)*fhig
60 continue

close(18)
61 end

```

```

*****
SUBROUTINE SORT3(N,RA,RB,RC)
real RA(N),RB(N),RC(N)

```

```

integer l,ir,i,j
real rra,rrb,rrc
L=N/2+1
IR=N
10 CONTINUE
  IF(L.GT.1)THEN
    L=L-1
    RRA=RA(L)
    RRB=RB(L)
    RRC=RC(L)
  ELSE
    RRA=RA(IR)
    RRB=RB(IR)
    RRC=RC(IR)
    RA(IR)=RA(1)
    RB(IR)=RB(1)
    RC(IR)=RC(1)
    IR=IR-1
    IF(IR.EQ.1)THEN
      RA(1)=RRA
      RB(1)=RRB
      RC(1)=RRC
      RETURN
    ENDIF
  ENDIF
  I=L
  J=L+L
20 IF(J.LE.IR)THEN
  IF(J.LT.IR)THEN
    IF(RA(J).LT.RA(J+1))J=J+1
  ENDIF
  IF(RRA.LT.RA(J))THEN
    RA(I)=RA(J)
    RB(I)=RB(J)
    RC(I)=RC(J)
    I=J
    J=J+J
  ELSE
    J=IR+1
  ENDIF
  GO TO 20
ENDIF
RA(I)=RRA
RB(I)=RRB
RC(I)=RRC
GO TO 10
END

```
