

PROGRAM LUMP

```
*****
* Lumps an x-ordered x-y file. From standard input it reads:      *
* input file name                                                *
* MIN MAX INC                                                    *
* MIN, MAX and INC are the minimum, maximum and incremental value of x.*
* The resulting x y file flushed to standard output.            *
* It lumps data-points in the new intervals with weighted averages *
* over x and y. Points that fall outside the available data-set are *
* omitted. If the new grid is actually smaller than the original one *
* or if some datapoints are missing, a linear interpolation is made *
* between the two neighboring original data-points.              *
*****
```

```
parameter(nphys=100000)
real eps,sx,sy,x,y,t,xx(nphys),yy(nphys)
*      ,xmx,xmn,xlast,xnext,ylast,ynext,wlow,whigh,tmn,tmx,tsp
CHARACTER*40 flin
INTEGER I,J,n,mm,ilast
```

```
read(*,'(a40)') flin
read(*,*) tmn,tmx,tsp
open(23,FILE=flin)
mm=100000
xmx=-1e20
xmn=1e20
do 10 i=1,mm
  READ(23,*,END=11) xx(i),yy(i)
  if (xx(i).lt.xmn) xmn=xx(i)
  if (xx(i).gt.xmx) xmx=xx(i)
10  continue
11  mm=i-1
close(23)
```

```
*****sort with respect to x-value
call SORT2(mm,xx,yy)
```

```
*****lump to tsp intervals
*****assume that the x-values are ordered
```

```
  j=0
  ilast=1
  eps=1e-8
  do 31 t=tmn,tmx-tsp+eps,tsp
    sy=0
    n=0
    sx=0
    do 20 i=ilast,mm
      if (xx(i).lt.tmn) goto 20
      if (xx(i).gt.(t+tsp)) goto 21
      sy=sy+yy(i)
      sx=sx+xx(i)
      n=n+1
20    continue
21    ilast=i
      j=j+1
      if (n.eq.0) then
***** first omit all points that are outside interval
      if ((t.lt.xmn).or.(t.gt.xmx)) then
        goto 31
```

```

***** otherwise take appropriate weighted average between
***** last point and first subsequent point
      else
        xlast=xx(ilast-1)
        xnext=xx(ilast)
        ylast=yy(ilast-1)
        ynext=yy(ilast)
        whigh=(t-xlast)/(xnext-xlast)
        wlow=1.-whigh
        x=xlast*wlow+xnext*whigh
        y=ylast*wlow+ynext*whigh
      endif
      else
        y=sy/real(n)
        x=sx/real(n)
      endif
      write(*,*) x,y
31  continue
      end

```

```

SUBROUTINE SORT2(N,RA,RB)
DIMENSION RA(N),RB(N)
L=N/2+1
IR=N
10  CONTINUE
      IF(L.GT.1)THEN
        L=L-1
        RRA=RA(L)
        RRB=RB(L)
      ELSE
        RRA=RA(IR)
        RRB=RB(IR)
        RA(IR)=RA(1)
        RB(IR)=RB(1)
        IR=IR-1
        IF(IR.EQ.1)THEN
          RA(1)=RRA
          RB(1)=RRB
          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)
          I=J
          J=J+J
        ELSE
          J=IR+1
        ENDIF
      GO TO 20
    ENDIF

```

```
RA(I)=RRA  
RB(I)=RRB  
GO TO 10  
END
```