APPENDIX A3.1
FORTRAN PROGRAM FOR CALCULATING EAA BASIN FLOWS AND PHOSPHORUS LOADS

```
program eaatpld
c modified August, 2000 for various ECP elements
c modified may 1999 for STA-5 inflows from Miami Canal (G350B, G349B)
c modified october 1998 for STA-5 & STA-2
c modified march 1998 for STA-6
c utilizes all composite samples
c compute eaa tp load  10-96 - additional comments added  10-3-96
c useage:g
c         >eaatpld eaa.job
c eaa.job = input ascii file specifying case conditions
c subroutines in subr.for
c maximum dimensions
c number of days = 20000 = 52  + years  ~(1978-2030)
c number of grab samples = 4000 per station
c number of composite samples = 4000 per station

c array dimensions increased to handle maximum of 70 terms
      integer*4 dgrab,dcomp,dlast,dbase,dbase0,d0
      character*64 title
      character*32 ofile1,ofile2,ofile3,ofile4,cfile,qfile,ofile0
      character*32 ofile5
      character*32 blank /' '/
      character*8 slab,dum8,qlab,ulab,usave(70),mname(4)
      common /a/ flowu(20000),wcomp(20000),wuse(20000),wusec(20000)
      common /b/ wgrab(20000)
      common /d/ dgrab(4000),dcomp(4000),cgrab(4000),ccomp(4000),
     & x(4000),iym(700),qsave(700,70),wsave(700,70),isgn(70),
     & wcsave(700,70),sumd(6),sumw(6),y(4000),prb(4000),ratio(2),
     & wc(2),wg(2),ncg(2)
character*32 confile

c array definitions
c    flowu()  = daily flow
c    wgrab()  = daily load computed from grab samples
c    wcomp()  = daily load computed from composite samples
c    wuse()   = daily load used in final result
c    wusec()  = daily load computed from composite samples
c    cgrab()  = grab-sample concentration
c    dgrab()  = grab-sample date
c    ccomp()  = composite sample concentration
c    dcomp()  = composite sample date
c    qsave,wsave,wcsave(month,station)
c             = storage of monthly flow, load, & composite load

c number of load calc methods
      data nmeth/3/
      data mname/'noflow','compos','  grab','  miss' /
```

```
c qfac: convert cfs*days to output units = cfs-days
      data qfac/1./

c scale factor to convert input sample concs (ppm) to (ppb)
      data sf/1000./

c factor: convert cfs*ppb to kg/day; sig: level of outliers
c     factor=24.*3600.*(0.3048**3)/1.e6
      factor=24.*3600/3.28**3/1.e6

c grab/composite ratio
c iratio = 0 compute r1 & r2 separately (original algorithm)
c iratio = 1 set r2 = r1
      data iratio/0/

c read input file [eaa.job] to get station labels and input parameters
      open(7,file=' ',status="old")

c read control parameters
      read(7,*) title,qfile,dum8,cfile,dum8,
     &nmaxc,dum8,dbase0,dum8,dbase,dum8,sig,dum8

c title = problem title
c qfile = input daily flow file
c cfile = input sample concentration file
c nmaxc = maximum duration of composite samples
c dbase0 = first day of base period yyyymmdd = 19781001
c dbase  = last day of base period yyyymmdd = 19910930
c sig = significance level for outlier screening in base period

cc
cc March 98 Modification - Look for Composite Samples NAFTER days beyond last flow date
cc
      nafter = nmaxc
cc
cc end of modification
cc

c read date range
      read(7,*) iymd1,dum8,iymd2,dum8,idchk,dum8
      write(*,*) 'sample date range =',iymd1,iymd2
      read(7,*) ofile0,dum8,ofile5,dum8,ofile1,dum8,ofile2,dum8,
     &    ofile3,dum8,ofile4,dum8
c output files (* = optional)
c ofile0 - sample inventory
c ofile5 - totals by term & time period (base pd & after)
c *ofile1 - daily results
c *ofile2 - monthly results for each term
c *ofile3 - monthly crosstab   (term x month)
c ofile4 - monthly totals (sum of all terms)
c
```

```
      read(7,*)

c jdatei()  converts yyyymmdd to julian dates (days from Jan 1, 1900)
      jdbase=jdatei(dbase)
      jymd1=jdatei(iymd1)
      jymd2=jdatei(iymd2)
      jdchk=jdatei(idchk)
      d0=jymd1-1

c open output file for sample statistics
      open(17,file=ofile0)
      write(17,171) idchk
 171      format( 'QLEFT = FLOW (CFSD) BETWEEN LAST GRAB',
     &    ' SAMPLE DATE WITH POSITIVE FLOW &'
     &    i9,' NOT COVERED BY COMPOSITE SAMPLE'/
     &    '              COMPOSITE SAMPLES        GRAB SAMPLES'/
     &    'STATION       N   DFIRST    DLAST',
     &    ' NTOT NOUT NUSE   DFIRST    DLAST',
     &    '  RATIO1  RATIO2    QLEFT')

c open input flow file
      open (8, file=qfile,status='old')

c open daily output file
      if(ofile1.ne.blank) then
          open(10,file=ofile1,status="unknown")
          write(10,"(a64)") title
          write(10,  2)
      endif
   2          format('station   date   ip mth     flow',
     &        '     load   cgrab   ccomp   cused c/g ratio')

c open monthly output file
      if(ofile2.ne.blank) then
          open(11,file=ofile2,status="unknown")
          write(11,*) title
          write(11,*)
     &         'station  mnth  days flow(csd)  load(kg) conc(ppb)
     &         compos(kg)'
      endif

c nsta = number of stations (terms)
      nsta=0

c *********** for each station (term) in job file *************
   10 nsta=nsta+1

      read(7,*,end=500) ulab,slab,qlab,ipos,icomp,isgn(nsta)
c ulab = output label for mass-balance term
c slab = sample station code
c qlab = flow station code
c ipos = flow sign indicator (1 = use positive flows, -1 = use negative flows)
```

```
c icomp = composite sample indicator
c     0 = ignore composite samples
c     1 = use composite samples
c     2 = use comp. samples, force comp./grab ratio = 1.0 (option not used)
c isgn = sign of term in computing total outflow volume and load
c        1 = outflow term from EAA
c        0 = ignore term
c       -1 = inflow or thruflow term

c capitalize labels
      CALL CONCAP(SLAB,8)
      CALL CONCAP(QLAB,8)
      CALL CONCAP(ULAB,8)
      write(*,*)
      write(*,*) 'term = ',ulab
      write(*,*) 'sample station = ',slab
      write(*,*) 'flow label = ',qlab
      usave(nsta)=ulab

c *********** read daily flows for current station ***********

      call flowread(8,jymd1, jymd2,qlab,nq,flowu)
c file start date must be <= jymd1
c jymd2 is adjusted to reflect end of file
c flow data set should contain no missing values

      if(nq.le.0) go to 999
      write(*,*) 'flow dates =',kdate(jymd1),kdate(jymd2)

c ********** load sample data ***********

      ngrab=0
      ncomp=0

c fixed format input
      open(16,file=cfile,status="old")
      do i=1,4
          read(16,*)
      enddo

c read next sample
 40   read(16,41,end=60) dum8,dd,tt,conc
 41       format(a8,2x,10f10.0)
      itype=jfix(tt)

c convert yymmdd to days from Jan 1, 1900
      idd=dd
      jdd=jdate(dd)

c check stations
      CALL CONCAP(DUM8,8)
      if(dum8 .ne. slab) go to 40
```

```
c check date

cc modified march 1998
cc
cc    if(jdd.lt.jymd1.or.jdd.gt.jymd2) go to 40
cc
      if(jdd.lt.jymd1.or.jdd.gt.jymd2+nafter) go to 40
cc
cc end of modification
cc
c check for valid sample value
      if(conc.eq.0.) go to 40

c rescale concentration and set to absolute value (negative values < detection limit)
      conc=sf*abs(conc)

c check for composite vs. grab sample
c sample dates must be in increasing order
      if(itype.eq.7.or.itype.eq.24) then
c process composite sample
          ncomp=ncomp+1
          ccomp(ncomp)=conc
          dcomp(ncomp)=jdd
          if(ncomp.gt.1.and.dcomp(ncomp).lt.dcomp(ncomp-1)) then
              write(*,*) 'compos sample out of sequence: ',idd
              stop
          endif
      else
c process grab sample
          ngrab=ngrab+1
          cgrab(ngrab)=conc
          dgrab(ngrab)=jdd
          if(ngrab.gt.1.and.dgrab(ngrab).lt.dgrab(ngrab-1)) then
              write(*,*) 'sample date out of sequence: ',idd
              stop
          endif
      endif

      go to 40

c end of sample file
   60 continue
      if(ngrab.gt.0) write(*,*) 'grab samples =     ',ngrab,
     &               kdate(dgrab(1)),kdate(dgrab(ngrab))
      if(ncomp.gt.0) write(*,*) 'composite samples =',ncomp,
     &               kdate(dcomp(1)),kdate(dcomp(ncomp))
      close(16)

c calculate average concentrations by date
      call xred(dgrab,cgrab,ngrab)
      call xred(dcomp,ccomp,ncomp)
      write(*,*) 'daily-avg grab samples =     ',ngrab
      write(*,*) 'daily-avg composite samples =',ncomp

c scratch composite samples if switch indicates so
      if(icomp.le.0) ncomp=0

c assign daily flows in cfs
      do 70 j=1,nq
          if(ipos.eq.1) then
              flowu(j)=amax1(flowu(j),0.)
          else
              flowu(j)=abs(amin1(flowu(j),0.))
          endif
          wgrab(j)=0.
          wcomp(j)=0.
   70     wuse(j)=0.

c calculate loads from composite samples
      dlast=0.
      do i=1,ncomp

c date range to apply composite-sample concentration
          j2=dcomp(i)-d0
          j1=max0(1,j2-nmaxc)
          if(j1.le.dlast) j1=dlast+1
          if(j1.gt.j2) j1=j2
          do j=j1,j2
              wcomp(j)=flowu(j)*ccomp(i)*factor
          enddo
          dlast=j2
      enddo

c eliminate grab-samples collected on days with no flow
      mgrab=0
      do i=1,ngrab
          if(flowu(dgrab(i)-d0).gt.0.) then
              mgrab=mgrab+1
              dgrab(mgrab)=dgrab(i)
              cgrab(mgrab)=cgrab(i)
          endif
      enddo
      ngrabt=ngrab
      ngrab=mgrab
      write(*,*) 'grab samples on days with positive flow =',ngrab
      if(ngrab.gt.0) write(*,*) 'date range =',kdate(dgrab(1)),
     &               kdate(dgrab(ngrab))

c screen base-period  grab samples for outliers
c based upon log(c) vs. log(q) regression
c (Snedecor & Cochran, Statistical Methods, 1980, pp. 167-168)
      if(sig.gt.0.) then
          ngt=ngrab
```

```
  110     j=0
          do i=1,ngrab
              prb(i)=1.
              if(dgrab(i).le.jdbase) then
                  j=j+1
                  x(j)=alog(flowu(dgrab(i)-d0))
                  y(j)=alog(cgrab(i))
              endif
          end do
          call outlyr(x,y,j,sig,prb,nrej)
          if(nrej.gt.0) then
              m=0
              do 150 i=1,ngrab
              if(prb(i).gt.sig) then
                  m=m+1
                  dgrab(m)=dgrab(i)
                  cgrab(m)=cgrab(i)
              else
                  write(*,140) kdate(dgrab(i)),cgrab(i),prb(i)
  140                 format(' ***outlier: date =',i9,
     &                      ',  conc = ', f10.1, ',  prob =',f8.3)
              endif
  150         continue
              ngrab=m
c repeat screen until no outliers are found
              go to 110
          endif
              ngout=ngt-ngrab
      endif

c calculate daily loads from grab samples by interpolation
      do  i=1,ngrab
          x(i)=dgrab(i)-d0
      enddo
      call eint3(ngrab,x,cgrab,nq,wgrab)

cc end of mod
      do i=1,nq
          wgrab(i)=wgrab(i)*flowu(i)*factor
      enddo

c ratio = load computed from composite samples / load computed from grab samples
c calculate load ratio for days with both composite and grab samples
c calc separate ratios for base period (ratio(1)) and after (ratio(2))
      do i=1,2
          wg(i)=0.
          wc(i)=0.
          ncg(i)=0
      end do
      do 220 i=1,nq
          if(wgrab(i).gt.0.and.wcomp(i).gt.0.) then
              if(i+d0.gt.jdbase) then
```

```
                  j=2
              else
                  j=1
              endif
              wg(j)=wg(j)+wgrab(i)
              wc(j)=wc(j)+wcomp(i)
              ncg(j)=ncg(j)+1
          endif
  220     continue
      do j=1,2
          ratio(j)=ratv(wc(j),wg(j))
c set to 1 if composite samples are ignored
c or if icomp=2
          if(icomp.le.0.or.icomp.eq.2) ratio(j)=1.
      end do
c if missing, set ratio(2)=ratio(1)
      if(ratio(2).le.0.) ratio(2)=ratio(1)

c sample inventory
      if(ncomp.le.0) then
          jc1=0
          jc2=0
      else
          jc1=dcomp(1)
          jc2=dcomp(ncomp)
      endif
      if(ngrab.le.0) then
          jg1=0
          jg2=0
      else
          jg1=dgrab(1)
          jg2=dgrab(ngrab)
      endif
c qdang = total flow between last grab sample date used and last flow date
      qdang=0.

c final load
c sumd = total days
c sumw = total load
c lq = 1 no flow, 2=composite, 3=grab
      do lq=1,5
          sumd(lq)=0.
          sumw(lq)=0.
      end do

c loop around days
      do i=1,nq
          jdd=i+d0

c wusec tracks loads computed from composite samples
          wusec(i)=0.
          if(i+d0.gt.jdbase) then
```

```
              ipd=2
          else
              ipd=1
          endif

c meth=1 no flow
          if(flowu(i).le.0.) then
              wuse(i)=0.
              meth=1

c meth=2 use composite load
          else if(wcomp(i).gt.0.) then
              wuse(i)=wcomp(i)
              wusec(i)=wcomp(i)
              meth=2

c meth=3 use grab load
          else if(wgrab(i).gt.0.) then

c iratio = 0 use separate values
c iratio = 1 use base period values only
              if(iratio.eq.0) then
                  rr=ratio(ipd)
              elseif(iratio.eq.1) then
                  rr=ratio(1)
              endif
              if(rr.eq.0.) rr=1.
              meth=3
              wuse(i)=wgrab(i)*rr

c  diagnostic - flow after last grab sample used in calc loads
              if(jdd.gt.jg2.and.jdd.le.jdchk)
     &        qdang=qdang+flowu(i)
          endif

          sumw(meth)=sumw(meth)+wuse(i)
          sumd(meth)=sumd(meth)+1.

c output daily results on days with positive flow
          if(ofile1.ne.blank.and.meth.gt.1.and
     &    .flowu(i).gt.0.) then
              write(10,280) ulab,kdate(jdd),ipos,
     &        meth,flowu(i),wuse(i),
     &        ratv(wgrab(i),flowu(i))/factor,
     &        ratv(wcomp(i),flowu(i))/factor,
     &        ratv(wuse(i),flowu(i))/factor,ratio(ipd)
  280             format(a8,1x,i8,i3,i3,f9.1,f9.2,3f8.1,f10.3)
          endif
      end do
c end of date loop

c log file
```

```
      write(17,172) ulab,ncomp,kdate(jc1),kdate(jc2),ngrabt,
     &ngout,ngrab,kdate(jg1),kdate(jg2),
     &ratio(1),ratio(2),qdang
 172      format(1h",a8,1h",i5,2i9,3i5,2i9,2f8.4,f9.1)

      write(*,235)
  235     format(' station    ncomp   ngrab',
     &    '   days1  ratio1   days2  ratio2')
cc   &    '   days1  ratio1   days2  ratio2 usedratio') changed 2/27/98
      write(*,245) ulab,ncomp,ngrab,ncg(1),ratio(1),
     &ncg(2),ratio(2)
cc   &ncg(2),ratio(2),rr   changed 2/27/98
  245     format(1x,a8,3i8,f8.5,i8,2f8.5)

c  method summary
      write(*,305) (mname(i),i=1,nmeth)
  305     format(' breakdown of load estimation methods:'/
     &              ' method:  ',6a10)
      do i=1,nmeth
          sumd(nmeth+1)=sumd(nmeth+1)+sumd(i)
          sumw(nmeth+1)=sumw(nmeth+1)+sumw(i)
      enddo
      write(*,"(' days% :',6f10.1)")
     &(100.*ratv(sumd(i),sumd(nmeth+1)),i=1,nmeth)
      write(*,"(' load% :',6f10.1)")
     &(100.*ratv(sumw(i),sumw(nmeth+1)),i=1,nmeth)

      m=0
      nk=3

      kd= kdate(jymd1)/100

      do k=1,nk
          x(k)=0.
      enddo
      mm=0
      do i=1,nq
          jd=kdate(i+jymd1-1)/100
          if(jd.ne.kd) then
c output monthly totals for current station
              m=m+1
              cc=ratv(x(2),x(1))*qfac/factor
              if(ofile2.ne.blank)
     &        write(11,350) ulab,kd,mm,(x(k),k=1,2),cc,x(3)
  350             format(a8,i8,i4,2f10.1,f10.1,f10.1)
              qsave(m,nsta)=x(1)
              wsave(m,nsta)=x(2)
              wcsave(m,nsta)=x(3)
              iym(m)=kd
              do k=1,nk
                  x(k)=0.
              enddo
              mm=0
              kd=jd
          endif
          mm=mm+1
          x(1)=x(1)+flowu(i)*qfac
          x(2)=x(2)+wuse(i)
          x(3)=x(3)+wusec(i)
      end do

      m=m+1
      if(ofile2.ne.blank) then
          cc=ratv(x(2),x(1))*qfac/factor
          write(11,350) ulab,kd,mm,(x(k),k=1,2),cc,x(3)
      endif
      iym(m)=kd
      qsave(m,nsta)=x(1)
      wsave(m,nsta)=x(2)
      wcsave(m,nsta)=x(3)

c end loop around stations
      go to 10

c end of station list
  500 continue

c weighted sum over all stations
      usave(nsta)='Total'
      do i=1,m
          qsave(i,nsta)=0.
          wsave(i,nsta)=0.
          wcsave(i,nsta)=0.
          do j=1,nsta-1
              qsave(i,nsta)=qsave(i,nsta)+qsave(i,j)*isgn(j)
              wsave(i,nsta)=wsave(i,nsta)+wsave(i,j)*isgn(j)
              wcsave(i,nsta)=wcsave(i,nsta)+wcsave(i,j)*isgn(j)
          end do
      end do

c output monthly cross-tab
      if(ofile3.ne.blank) then
          open(12,file=ofile3,status="unknown")
          write(12,"(a64)") title
          write(12,*) 'flows in cfs-days'
          write(12,"(a6,2x,50a10)") 'month',(usave(i),i=1,nsta)
          do 530 i=1,m
  530         write(12,"(i6,50f10.1)") iym(i),(qsave(i,k),k=1,nsta)
          write(12,*)
          write(12,*) 'loads in kg'
          write(12,"(a6,2x,50a10)") 'month',(usave(i),i=1,nsta)
          do 540 i=1,m
  540         write(12,"(i6,50f10.1)") iym(i),(wsave(i,k),k=1,nsta)
          close(12)
      endif

c output totals before & after base period
      if(len_trim(ofile5).gt.0) then
c convert cfsd to kac-ft
          qqfac=24.*3600./43560./1000.
          open(12,file=ofile5)
          write(12,39) title,dbase
          do i=1,nsta
              x(1)=0.
              x(2)=0.
              y(1)=0.
              y(2)=0.
              tb=0
              ta=0
              do j=1,m
                  if(iym(j).gt.dbase/100) then
                      k=2
                      ta=ta+1
                  else
                      k=1
                      tb=tb+1
                  endif
                  x(k)=x(k)+qsave(j,i)
                  y(k)=y(k)+wsave(j,i)
              enddo
              ta=ta/12
              tb=tb/12
              write(12,38) usave(i),isgn(i),
     &        qqfac*x(1)/tb,y(1)/tb,ratv(y(1),x(1))*qfac/factor,
     &        qqfac*x(2)/ta,y(2)/ta,ratv(y(2),x(2))*qfac/factor
          enddo

 38       format(1h",a8,1h",i4,2(2f12.3,f10.1))
 39       format(a64/'Yearly Averages for Each Term & Time Period'/
     &    '           In Base Period <=',i8,8x,
     &    '       After Base Period'/
     &    'Term      Sign Flow(kaf/y)  Load(kg/y) Conc(ppb) '
     &    ' Flow(kaf/y) Load(kg/y)  Conc(ppb)')
      endif

c output monthly totals across all stations
      if(ofile4.ne.blank) then
          open(13,file=ofile4)
          write(13,"(a64)") title
          write(13,*) 'totals'
          write(13,567)
 567          format('month  flow(cfsd)    load(kg) conc(ppb)',
     &        ' grab(out) comp(out)  grab(in)  comp(in) comp(%)')

c loop around months
          do k=1,4
```

```
              y(k)=0.
          end do
          do i=1,m
              do k=1,4
                  x(k)=0.
              enddo
              do j=1,nsta-1
                  if(isgn(j).lt.0) then
c grab & composite inflows
                      x(3)=x(3)+wsave(i,j)-wcsave(i,j)
                      x(4)=x(4)+wcsave(i,j)
                  elseif(isgn(j).gt.0) then
c grab & composite outflows
                      x(1)=x(1)+wsave(i,j)-wcsave(i,j)
                      x(2)=x(2)+wcsave(i,j)
                  endif
              enddo
c composite as % of total absolute value
              x(5)=ratv(x(2)+x(4),x(3)+x(4)+x(1)+x(2))*100.
              write(13,560) iym(i),qsave(i,nsta),wsave(i,nsta),
     &        ratv(wsave(i,nsta),qsave(i,nsta))*qfac/factor,
     &        (x(k),k=1,5)
 560              format(i6,2f12.1,5f10.1,f8.1)
c sum over all months
              do k=1,4
                  y(k)=y(k)+x(k)
              enddo
              qsave(m+1,nsta)=qsave(m+1,nsta)+qsave(i,nsta)
              wsave(m+1,nsta)=wsave(m+1,nsta)+wsave(i,nsta)
              wcsave(m+1,nsta)=wcsave(m+1,nsta)+wcsave(i,nsta)
          enddo
          y(5)=ratv(y(2)+y(4),y(3)+y(4)+y(1)+y(2))*100.
          write(13,570) qsave(m+1,nsta),wsave(m+1,nsta),
     &    ratv(wsave(m+1,nsta),qsave(m+1,nsta))*qfac/factor,
     &    (y(k),k=1,5)
 570          format(/'total ',2f12.1,5f10.1,f8.1)
          close(13)
      endif
 999 close(10)
      end

      subroutine flowread(ifile,ibdate,iedate,clab,nq,values)

c modified March 2017 to include C10, C12, C12A, C4A, S236 and epd07
c modified Feb 2016 for A-1 FEB outflow structure g722
c modified Sept 2004 for STA3/4 inflows and outflows
c modified June 2004 for addition of g507, g204-g206
c modified August 2000 for various ECP elements
c reads daily flows - modified for STA-6 march 1998
c modified for STA-2 & STA-5 may 1999
c missing values not allowed in flow file
      character*8 clab
      character*8 labs(70)
      real values(1)

c these labels correspond to flow station labels in control file
      data labs /"s5a+s5aw",  "hgs5",     "wpbthru",  "s6",
     &           "s2/s6",     "hilthru",  "s7",       "s150",
     &           "s2/s7",     "thrulake", "thrus7",   "thrus150",
     &           "s8",        "s3",       "g88",      "g136",
     &           "holey",     "miathru",  "g250",     "g600",
     &           "g605",      "g606",     "g344a",    "g344b",
     &           "g344c",     "g344d",    "g328",     "g349b",
     &           "g350b",     "ebps",     "esps",     "g410",
     &           "g402a",     "g402b",    "g402c",    "g402d",
     &           "g404"       "g357",     "g204",     "g205",
     &           "g206",      "g507",     "g370",     "g372",
     &           "g376abc",   "g376def",  "g379abc",  "g379de",
     &           "g381ab",    "g381cdef", "ssdd",     "sfcd",
     &           "g371",      "g373" ,    "g373bc",   "g434",
     &           "g435",     "g722thru",   “c10”,     “c12a”,
     &           “c12”,      “c4a”,        “s236”,    “epd07”/

c number of daily flows in input file
      data nqin /70/
      rewind ifile
      do i=1,4
          read(ifile,*)
      enddo
      nq = 0
      do I=1,nqin
          call CONCAP(LABS(I),8)
      enddo
  90  read(ifile,222,end=100) dd,qhgs5,qs5as5aw,qs2,
     &qs6, qs7, qs150, qs3, qs8, qg88, qg136, qholey,
     &qg250,qg600,qg605,qg606,qg344a,qg344b,
     &qg344c,qg344d,qg328,qg349b,qg350b,
     &qebps,qesps, qg410,qg402a,
     &qg402b, qg402c, qg402d,qg404, qg357,qg204,qg205,
     &qg206, qg507, qg370, qg372, qg376a, qg376d, qg379a,
     &qg379d, qg381a, qg381c, qssdd, qsfcd, qg371,
     &qg373, qg373bc, qg434, qg435, qg722,
     &qc10, qc12a,qc12,qc4a,qs236,qepd07


c Modify East Beach, South Florida and South Shore flows to account for the portion of
c these basins that was previously in the EAA.

      qebps = 0.813 * qebps
      qssdd = 0.966 * qssdd
      qsfcd = 0.799 * qsfcd


 222      format(100f10.0)
```

```
c convert yymmdd to julian
      jfdate=jdate(dd)
      if(jfdate.lt.ibdate) then
          goto 90
      elseif(jfdate.gt.iedate) then
          return
      elseif(nq.eq.0.and.jfdate.ne.ibdate) then
          write(*,*) 'flow file starting date too late: ',jfix(dd)
          stop
      elseif(nq.gt.0.and.jfdate-jflast.ne.1) then
          write(*,*) 'flow file dates out of sequence: ',jfix(dd)
          stop
      endif
      nq=nq+1
      jflast=jfdate

c split s2 outflow between s6 (hillsboro qs2h) and S7 (nnriver qs2n) basins
      qs2n =(qs2 / (1.534769))
      qs2h = qs2 - qs2n

c Adds STA3/4 outflows to total North New River inflows
c by adding g722 as an inflow to EAA

      qin = amax1(0., qs2n) - amin1(0.,qs7) - amin1(0., qs150)
     & + amax1(0.,qg376a) + amax1(0.,qg376d) + amax1(0.,qg379a)
     & + amax1(0.,qg379d) + amax1(0.,qg722)

c total flow thru in north new river canal
c Combines G370 flow through to S7 (9/28/04)
c Add G371 to flow through term for WY06

      ft = amin1(qin, amax1(0., qs7)+amax1(0., qs150))+ amax1(0., qg370)-
     & amax1(0.,qg376a) - amax1(0.,qg376d)-amax1(0.,qg379a)-
     & amax1(0.,qg379d)+ amax1(0., qg371) + amax1(0.,qg434)
     & + amax1(0.,qg435))

      do i = 1, nqin
          if(clab .eq. labs(i)) then
              ind = i
              go to 200
          endif
      end do
      write(*,*) 'flow station label not found:', clab
      stop
      go to 29
 200  goto (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,
     &20,21,22,23,24,25,26,27,28,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,
     &45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65), ind

c s5a+s5aw    outflow
 1    x = qs5as5aw
```

```
      go to 29

c hgs5 outflow
 2    x = qhgs5
      go to 29

c s5athru west palm beach canal flowthru
 3    if(qhgs5 .le. 0 ) then
          x = 0.
      else
          x = amin1(qhgs5, amax1(qs5as5aw+qg250-qebps, 0.))
      end if
      go to 29

c s6 outflow
 4    x = qs6
      go to 29

c s2/s6 s2 outflow to lake from hillsboro basin
 5    x = qs2h
      go to 29

c s6thru hillsboro canal flowthru
 6    if(qs2h .le. 0) then
          x = 0.
      else
          x = amin1(qs2h, amax1(qs6-qesps, 0.))
      end if
      go to 29

c s7 outflow
 7    x = qs7
      go to 29

c s150 outflow
 8    x = qs150
      go to 29

c s2/s7 outflow to lake from s7 basin
 9    x = qs2n
      go to 29

c thrulake - nnriver flowthru from lake
 10   if(qin .eq. 0) then
          x = 0.
      else
          x = amax1(0., qs2n) * ft / qin
      end if
      go to 29

c thrus7 - nnriver flowthru from s7
 11   if(qin .eq. 0) then
```

```
          x = 0.
      else
          x = -amin1(0., qs7) * ft / qin
      end if
      go to 29

c thrus150 - nnriver flowthru from s150
 12   if(qin .eq. 0) then
          x = 0.
      else
          x = -amin1(0., qs150) * ft / qin
      end if
      go to 29

c s8 outflow
 13   x = qs8
      go to 29

c s3 outflow
 14   x = qs3
      go to 29

c g88 inflow
 15   x = qg88
      go to 29

c g136 inflow
 16   x = qg136
      go to 29

c holeyland
 17   x = qholey
      go to 29

c s8 miami canal flowthru
 18   if(qs3 .le. 0) then
          x = 0.
      else
          x = amin1(qs3, amax1(0.,
     &    qs8-qg88-qg136+qholey-qg606-qg605+qg349b+qg350b-qg344a-
     &    qg344b-qqg344c-g344d-qg402a-qg402b-qg402c-qg402d+qg410+
     &    qg404 + qg357-qg204-qg205-qg206+qg507+qg372- qg381a - qg381c-
     &    qssdd - qsfcd + qg373 + qg373bc)
      endif
      goto 29

c enr inflow  - eaa outflow
 19   x=qg250
      goto 29

c sta6 inflow
 20   x=qg600
```

```
      goto 29

c sta6 bypass
 21   x=qg605
      goto 29

c sta6 outflow
 22   x=qg606
      goto 29

c sta5 outflows
 23   x=qg344a
      goto 29
 24   x=qg344b
      goto 29
 25   x=qg344c
      goto 29
 26   x=qg344d
      goto 29

c sta2 supplementary inflow

 27   x=qg328
      goto 29

c sta5 inflows from miami canal
 28   x=qg349b
      goto 29
 30   x=qg350b
      goto 29

c East Beach outflow - EAA inflow
 31   x=qebps
      goto 29

c East Shore outflow - EAA inflow
 32   x=qesps
      goto 29

c Rotenberger inflow  - EAA outflow
 33   x=qg410
      goto 29

c Rotenberger outflows - EAA inflow
 34   x=qg402a
      goto 29
 35   x=qg402b
      goto 29
 36   x=qg402c
      goto 29
 37   x=qg402d
      goto 29
```

```
c G404 outflow
 38   x=qg404
      goto 29

c G357 outflow
 39   x=qg357
      goto 29
c G204 inflow
 40   x=qg204
       goto 29
c G205 inflow
 41   x=qg205
       goto 29
c G206 inflow
 42   x=qg206
       goto 29
c G507 outflow
 43   x=qg507
       goto 29
c G370 outflow
 44   x=qg370
       goto 29
c G372 outflow
 45   x=qg372
       goto 29
c G376abc inflow
 46   x=qg376a
       goto 29
c G376def inflow
 47   x=qg376d
       goto 29
c G379abc inflow
 48   x=qg379a
       goto 29
c G379de inflow
 49   x=qg379d
       goto 29
c G381ab inflow
 50   x=qg381a
       goto 29
c G381cdef inflow
 51   x=qg381c
       goto 29
c SSDD inflow
 52   x=qssdd
       goto 29
c SFCD inflow
 53   x=qsfcd
       goto 29
c G371 outflow
 54   x=qg371
```

```
       goto 29
c G373 outflow
 55   x=qg373
       goto 29
c G373BC outflow
 56   x=qg373bc
       goto 29
c G434 outflow
 57   x=qg434
      goto 29
c G435 outflow
 58   x=qg435
      goto 29
c G722 A-1 FEB flow through
 59   if(qin .eq. 0) then
          x = 0.
      else
          x = amax1(0., qg722) * ft / qin
      end if
      go to 29
c East Beach outflow to Lake Okeechobee – New EAA outflow
 60   x=qc10
      goto 29
c Closter outflow to Lake Okeechobee – New EAA outflow
 61   x=qc12a
      goto 29
c East Shore outflow to Lake Okeechobee – New EAA outflow
 62   x=qc12
      goto 29
c South Shore outflow to Lake Okeechobee – New EAA outflow
 63   x=qc4a
      goto 29
c South Florida Conservancy outflow to Lake Okeechobee – New EAA outflow
 64   x=qs236
      goto 29
c South Florida Conservancy outflow to Industrial Canal – New EAA outflow
 65   x=qepd07
      goto 29


 29   values(nq) = x

      go to 90

 100  iedate=jfdate
      return
      end
c subroutines in subr.for


c subroutines for eaa software
c
```

```
c date functions
c
c date sequence number = number of days from Jan 1, 1900 (= Lotus 123 date)
c All reals=real*4, All integers = Integer*4
c function               inputs          returns
c idate(iy,im,id)        iy,im,id        date sequence number
c jdate(d)               yymmdd          date sequence number
c kkdate(d)              yymmdd          yyyymmdd
c jdatei(k)              yyyymmdd        date sequence number
c kdate(j)               date sequence   yyyymmdd
c ddate(j)               date sequence   yymmdd
c sub yymmdd(d,iy,im,id) yymmdd          iy,im,id
c sub iymmdd(k,iy,im,id) yyyymmdd        iy,im,id
c idbt(k1,k2)            2 x yyyymmdd    days between 2 dates, inclusive
c imonth(char3)          character month month number
c mday(iy,im)            iy,im           number of days in month

        function idate(iy,im,id)

       integer mdy(12)
        DATA MDY/0,31,59,90,120,151,181,212,243,273,304,334/

c returns days from Jan 1, 1900 for input iy,im,id
c year in yy format

c years
       jy=iy+1900

c if iy<50 assume turn of century
       if(iy.lt.50) jy=jy+100

        idate=0
c check for valid date
        if(im.le.0.or.im.gt.12) return
       if(id.lt.1.or.id.gt.mday(iy,im)) return

        idate=mdy(im)+(jy-1900)*365.+id+(jy-1897)/4

c add 1 day if leap year and after february
        if(mod(jy,4).eq.0.and.im.gt.2) idate=idate+1
        return
        end

            function jdate(d)
c returns date sequence number for input d in yymmdd format
            call yymmdd(d,iy,im,id)
            jdate=idate(iy,im,id)
            return
            end

       function jdatei(id)
c returns date sequence number for input id in yyyymmdd format
```

```
       j=id-19000000
       jdatei=jdate(float(j))
       return
       end



       function kkdate(d)
c returns yyyymmdd for input in yymmdd
       kkdate=d+19000000
       if(d.le.500000.) kkdate=kkdate+1000000
       return
       end



       function kdate(id)
c returns integer date yyyymmdd for julian date id
       kdate=ddate(id)
       if(kdate.eq.0) then
              return
              elseif(kdate.lt.500101) then
              kdate=kdate+20000000
              else
              kdate=kdate+19000000
              endif
       return
       end

        function ddate(id)

c returns date in yymmdd format for input id =
c      number of days from Jan 1, 1900

             ddate=0.
             if(id.le.0) return

c first find year, roughly
             jy=id/367
 13          if(idate(jy+1,1,1).le.id) then
                     jy=jy+1
                     goto 13
                     endif

c find month
             do 10 jm=2,12
               if(idate(jy,jm,1).gt.id) goto 12
 10          continue
 12          jm=jm-1

c find day
             jd=id-idate(jy,jm,1)+1
```

```
ccc adjust year
            if(jy.gt.99) jy=jy-100

c compute ddate
             ddate=10000.*jy+jm*100.+jd
             return
             end


       subroutine yymmdd(date,iy,im,id)

c convert real date yymmdd to integer year yy, month, day

       iy=0
       im=0
       id=0
       iy=jfix(date/10000.)
       im=jfix((date-iy*10000.)/100.)
       id=jfix(date-iy*10000.-im*100.)
       return
       end

       subroutine iymmdd(idate,iy,im,id)

c convert integer date to integer year, month, day

       iy=0
       im=0
       id=0
       iy=jfix(idate/10000)
       im=jfix((idate-iy*10000)/100)
       id=jfix(idate-iy*10000-im*100)
       return
       end

       function mday(iy,im)

c number of days in current month

       dimension mdy(12)
       data mdy/31,28,31,30,31,30,31,31,30,31,30,31/
       mday=0
       if(im.gt.12.or.im.lt.1) return
       mday=mdy(im)
       if(im.eq.2.and.mod(iy,4).eq.0.) mday=mday+1
       return
       end

          subroutine outlyr(x,y,n,sig,prb,nrej)
c screen for outliers - linear regression y(n) vs. x(n)
c sig = rejection significance level
```

```
c returns prb(n) = significance level for rejection
c nrej = number of screened data points
c snedecor and cochran, p. 157-158
         dimension x(1),y(1),prb(1)
         if(n.le.3) return
         sy=0.
         sy2=0.
         sx=0.
         sx2=0.
         sxy=0.
         nrej=0
         nn=n
c first compute regression
         do 100 i=1,n
             prb(i)=1.
             sy=sy+y(i)
             sx=sx+x(i)
             sy2=sy2+y(i)*y(i)
             sx2=sx2+x(i)*x(i)
             sxy=sxy+x(i)*y(i)
 100         continue
         txy=sxy-sx*sy/n
         tx2=sx2-sx*sx/n
         ty2=sy2-sy*sy/n
         tx=sx/n
         ty=sy/n
         b=txy/tx2
         a=ty-b*tx
c find maximum residual
 10      rmax=0.
         j=0
         do 200 i=1,n
         if(prb(i).eq.1.) then
            resid=abs(y(i)-b*x(i)-a)
             if(resid.gt.rmax) then
               j=i
               rmax=resid
               endif
            endif
 200     continue
         if(j.le.0) return
c compute regression with point j excluded
         nn=nn-1
         if(nn.le.3) return
         sxy=sxy-x(j)*y(j)
         sx2=sx2-x(j)*x(j)
         sy2=sy2-y(j)*y(j)
         sy=sy-y(j)
         sx=sx-x(j)
         txy=sxy-sx*sy/nn
         tx2=sx2-sx*sx/nn
         ty2=sy2-sy*sy/nn
         tx=sx/nn
         ty=sy/nn
         b=txy/tx2
         a=ty-b*tx
         se2=(ty2-b*b*tx2)/(nn-2)
         if(se2.le.0.) return
         se=sqrt(se2)
c test residual
         resid=y(j)-b*x(j)-a
         sr=se*sqrt( 1.+1./nn + (x(j)-tx)**2/tx2 )
         t=resid/sr
         prb(j)=probt(t,nn-2)*(nn+1)
         if(prb(j).gt.sig) return
         nrej=nrej+1
         go to 10
         end

        subroutine eint3(n,e,x,ni,xi)
c interpolation
c inputs e(i),x(i),i=1,n
c output ei(i),xi(j),j=1,ni
c      ei(j)==j

        dimension x(1),e(1),xi(1)
c
         i=1
         do 100 j=1,ni
            if(j.gt.e(i)) go to 110
              xi(j)=x(i)
              go to 100
 110        if(j.lt.e(n)) go to 120
              xi(j)=x(n)
              go to 100
 120        if(j.le.e(i+1)) go to 125
              i=i+1
              go to 120
 125        f=(j-e(i))/(e(i+1)-e(i))
            xi(j)=(1.-f)*x(i)+f*x(i+1)
 100    continue
        return
        end

          subroutine xred(ix,y,n)
c replaces x() and y() with running means
c for common values of ix()
c length n
c destroys input vectors
          dimension y(1)
          integer ix(1),ixlast
          if(n.le.1) return
          ixlast=ix(1)
          m=1
          k=0
          sum=y(1)
          do 10 j=2,n
          if(ix(j).ne.ixlast) then
             k=k+1
             ix(k)=ixlast
             y(k)=sum/m
             ixlast=ix(j)
             m=0
             sum=0.
             endif
          m=m+1
          sum=sum+y(j)
 10       continue
          k=k+1
          ix(k)=ixlast
          y(k)=sum/m
          n=k
          return
          end


       function ratv(x1,x2)
c divide x1 by x2 or set to 0.
       if(x2.ne.0.) then
           ratv=x1/x2
           else
           ratv=0.
           endif
       return
       end

       function ic8(c1,c2)
c compares strings c1 and c2
c returns 1 if they are identical
c case not significant
       character*8 c1,c2,c3,c4
c
       c3=c1
       call concap(c3,8)
       c4=c2
       call concap(c4,8)

       if(c3.eq.c4) then
            ic8=1
           else
            ic8=0
           endif
        return
        end

       function match(n,label,char)
```

```
c lookup char in label()
       character*8 label(1),char
       match=0
       do 10 i=1,n
       if(ic8(char,label(i)).gt.0) then
         match=i
         return
         endif
 10    continue
       return
       end

        function probg(s,r,z)
c  f statistic
c used with probf and probt
        u=2./9./s
        v=2./9./r
        q=abs((1.-v)*(z**.333333)-1.+ u)/sqrt(v*z**.6666667+u)
        if (r.lt.4) q=q*(1.+.08*(q**4)/(r**3))
        probg=.5/(1.+q*(.196854+q*(.115194+q*(3.44e-04+q*.019527))))**4
        return
        end

        function probt(t,n)
c  two-tailed - modified from "some common basic programs"
        probt=1.0
        if(t.eq.0..or.n.le.0) return
        w=t*t
        if (w.lt..5) then
           s=n
           r=1.
           z=1./w
           else
           s=1.
           r=n
           z=w
           endif
 20     probt=probg(s,r,z)
        if(w.lt..5) probt=1.-probt
        return
        end

       subroutine concap(string,n)
c convert string to caps
       character*1 string(1)
       do i=1,n
       j=ichar(string(i))
       if(j.gt.96.and.j.lt.123) string(i)=char(j-32)
       enddo
       return
       end
```

```
      subroutine pquote(cin,cout)
c returns string cin enclosed in quotes
c     xxxxx ---> "xxxxx"
      character*16 cin,cout,ctemp
      character*1 cc(16)
      equivalence (ctemp,cc(1))
      cout=' '
      ctemp=cin
      n=len_trim(cin)
      cc(n+1)='"'
      write(cout,1) (cc(i),i=1,n+1)
 1     format('"',20a1)
      return
      end


      function idbt(id1,id2)
c days between id1 & id2, inclusive
      idbt=jdatei(id2)-jdatei(id1)+1
      return
      end

      function imonth(c)
c convert character month to integer month
      character*3 c
      character*3 mlab(12) /'JAN','FEB','MAR','APR','MAY','JUN',
     &              'JUL','AUG','SEP','OCT','NOV','DEC'/

      imonth=0
      if(len_trim(c).le.0) return

      call concap(c,3)
      do i=1,12
               if(c.eq.mlab(i)) goto 5
              enddo
      write(*,*) 'Invalid Month =', c
      stop
 5    imonth=i
      return
      end
```