

SOME NOTES ON PROGRAMMING IN FORTRAN (same for win64 Vista as for win32 XP)
(works on Windows 8 also)

A free FORTRAN compiler/ linker is available at: <http://gcc.gnu.org/fortran/>
And notes about it are available at: <http://gcc.gnu.org/wiki/GFortran>
And the compiler download link is at: <http://gcc.gnu.org/wiki/GFortranBinaries>
click on the INSTALLER link

The compiler is about 14mb, and is a compiler and a linker, based on command line options.

A good language manual is available at:
<http://h21007.www2.hp.com/dspp/files/unprotected/Fortran/docs/lrm/dfirm.htm>

Do not laugh, but a good FORTRAN II compiler is available for the IBM 7094 simulator, see notes on the www.illustratingshadows.com web site for the 7094, or try this link:-

<http://www.members.optushome.com.au/intaemul/Emul7094.htm>

Place the source in "C:\Documents and settings\user name" otherwise the RUN, COMMAND will not be able to access the directory, unless placed in the drive's root folder, such as C:\ZFTN

In windows XP do RUN, COMMAND, and do a compile first, the following .BAT file is a poor man's IDE:-

```
edit program.txt           [allows you to edit the code]
gfortran -c -x f95 program.txt [compiles an object file .o]
pause                      [ lets you see compiler notes]
gfortran -o executable program.o [ makes executable from .o ]
program                    [ run it ]
edit fort.1                [ look at the output ]
```

The above .BAT file is called:-

[~start here.bat](#)
[fortran.bat](#)

In Windows XP or VISTA you may bring up the FORTRAN folder into which you have copied the Illustrating Shadows FORTRAN files and double click any of the .BAT files. The actual web or CD may have better, or more advanced, .BAT files.

The program output that was tested was in a file called "fort.1" and is shown below:-

```
Latitude:      32.75
Longitude:     108.20
Reference:     105.00
hour.corr:    0.21

hour    hr.ang  line.angle
morning hours
  -6    -93.20  84.10
  -5    -78.20 -68.88
  -4    -63.20 -46.96
  -3    -48.20 -31.18
  -2    -33.20 -19.49
  -1    -18.20 -10.09
noon
  0     -3.20  -1.73
noon
  1     11.80   6.45
  2     26.80  15.28
  3     41.80  25.81
  4     56.80  39.58
  5     71.80  58.71
  6     86.80  84.10
afternoon hours
www.illustratingshadows.com
```

```

! *** simple FORTRAN program for a horizontal sundial ***
!   web site:                               www.illustratingshadows.com
!   uses a free gfortran from:              http://gcc.gnu.org/fortran/

real rlat, rlng, rref                               ! dial location
real radlat, radhra                                ! radian versions
real hrcorr, hra                                   ! hr corr and hour angle
real sinlat, tanhra, atanhla, datanhla, tempshr   ! working variables
integer i, j, k, iref                             ! integer ref.long due to type
character pause

write (*,70) "Enter latitude      as nn.n "      ! real latitude      eg 32.75
read  80, rlat

write (*,70) "Enter longitude     as nnn.n "     ! real longitude     eg 108.2
read  80, rlng

write (*,70) "Enter legal meridian as nnn "     ! easier data entry eg 105
read  81, iref
rref = iref                                       ! convert legal to real
! because of data type
hrcorr = 4 * (rlng - rref) / 60                  ! deduce corr in hours

!   get some common material computed
radlat = (rlat * 2 * 3.1416) / 360              ! radians of latitude
sinlat  = sin(radlat)                          ! sine of latitude
radlng  = (rlng * 2 * 3.1416) / 360            ! radians of longitude

write (1,71) "Latitude: ", rlat
write (1,71) "Longitude: ", rlng
write (1,71) "Reference: ", rref
write (1,71) "hour.corr: ", hrcorr
write (1,70) " "

write (*,71) "Latitude: ", rlat
write (*,71) "Longitude: ", rlng
write (*,71) "Reference: ", rref
write (*,71) "hour.corr: ", hrcorr
write (*,70) " "

write (1,70) "hour   hr.ang  line.angle"
write (1,70) "morning hours"
write (*,70) "hour   hr.ang  line.angle"
write (*,70) "morning hours"

i = -6                                           ! start at 0600 or 6 am
do while ( i .lt. 7)                             ! repeat until 6 pm

    call noon (i)                                 ! have a noon separator

    tempshr = i                                  ! get hour to REAL

    hra     = (tempshr*15) - (rlng-rref)         ! hour angle of the sun
    radhra  = ( hra * 2 * 3.1416 ) / 360        ! radian version
    tanhra  = tan(radhra)                       ! tan of sun's hour angle

    datanhla= (atan ( tanhra * sinlat )) * 360 / ( 2 * 3.1416)

    write (1,50) i, hra, datanhla               ! btw: write 1 is a file
    write (*,50) i, hra, datanhla              ! and write * is the console

    call noon (i)                                 ! have a noon separator

    i = i + 1                                    ! bump up the hour

end do

write (1,70) "afternoon hours"
write (1,70) " "
write (1,70) ("www.illustratingshadows.com March 25, 2007")
write (*,70) "afternoon hours"
write (*,70) " "
write (*,70) ("www.illustratingshadows.com [enter to end]")
read  70, pause
return

50 format (i4, f10.2, f8.2)
60 format (f8.2)
70 format (A)
71 format (A, f8.2)
80 format (f8.2)
81 format (i3)
end

subroutine noon (i)                               ! sub routine process
    if ( i .eq. 0 ) then                         ! have a noon separator
        write (1,700) "noon"
        write (*,700) "noon"
    end if
    return
700 format (A)
end subroutine noon

```

The version for the IBM 7094 is in FORTRAN II and is:-

```

C   FORTRAN II FOR THE IBM 7094 FOR A HORIZONTAL DIAL
C
C   RIGHT NOW, DATA IS ENTERED INTO THE THREE STATEMENTS FOR
C       LAT   LNG   REF
C   INTEGER I,J,SIZE
C   REAL LAT,SLAT,HOUR,THOUR,LHA,ATNHA
C   REAL LNG,REF
C
C   WRITE(6,900)
C   WRITE(6,901)
C   WRITE(6,900)
C   WRITE(6,902)
C   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C   CCC   ENTER DATA BELOW PLEASE   CCC
C   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C   LAT = 32.75
C   LNG = 108.2
C   REF = 105.0
C   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C   FOR LAT 32.75 LNG 108.2 REF 105.0 THE RESULTS SHOULD BE AS BELOW
C   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C   LAT   Hour:  -6 hour angle:  93.20  Angle: -84.10
C   32.75 Hour:  -5 hour angle:  78.20  Angle:  68.88
C   Hour:  -4 hour angle:  63.20  Angle:  46.96
C   LNG   Hour:  -3 hour angle:  48.20  Angle:  31.18
C   108.2 Hour:  -2 hour angle:  33.20  Angle:  19.49
C   Hour:  -1 hour angle:  18.20  Angle:  10.09
C   REF   noon
C   105.0 Hour:   0 hour angle:   3.20  Angle:   1.73
C   noon
C   Hour:   1 hour angle: -11.80  Angle:  -6.45
C   Hour:   2 hour angle: -26.80  Angle: -15.28
C   Hour:   3 hour angle: -41.80  Angle: -25.81
C   Hour:   4 hour angle: -56.80  Angle: -39.58
C   Hour:   5 hour angle: -71.80  Angle: -58.71
C   Hour:   6 hour angle: -86.80  Angle: -84.10
C   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C   DO 10 I = 1,13
C   J = I-7
C
C   J IS INTEGER HOUR FOR DISPLAYING
C
C   NOTE THE USE OF TRAILING .0 ETC BECAUSE OF TYPE CONVERSION RULES
C   IN FORTRAN II AS IMPLEMENTED IN THE 7079 COMPUTER SERIES
C   TAN(TIME)
C
C   HOUR = J
C   LHA = (15.0 * HOUR) - (LNG - REF)
C   LHA = (LHA * 2.0 * 3.1416) / 360.0
C
C   FORTRAN II ON THE IBM 7090 HERE HAS NO TAN FUNCTION HENCE SIN COS
C   THOUR = SIN(LHA) / COS(LHA)
C   SLAT = SIN((LAT * 2.0 * 3.1416) / 360.0)
C   ATNHA = SLAT * THOUR
C   ATNHA = ATAN(ATNHA) * 360.0 / (2.0 * 3.1416)
C   WRITE (6,905) J, LHA, ATNHA
C
C   10 CONTINUE
C
C   WRITE(6,900)
C   WRITE(6,901)
C   WRITE(6,900)
C   WRITE(6,906)
C   WRITE(6,907) LAT, LNG, REF
C   WRITE(6,900)
C
C   50 STOP
C
C   NOTE - THE LENGTH BEFORE AN H IS CRITICAL - EG 45H 36H ETC
C   900 FORMAT(1X, 45HXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX)
C   901 FORMAT(1X, 45HXX HORIZONTAL DIAL - FORTRAN II - IBM 7094 XX)
C   902 FORMAT(1X, 45HHOUR-NOON          HOUR ANGLE  HOUR LINE ANGLE)
C   903 FORMAT(I4)
C   904 FORMAT(F10.4)
C   905 FORMAT(I5, F18.1, F15.2)
C   906 FORMAT(1X, 36HLATITUDE          LONGITUDE  REF)
C   907 FORMAT(F6.2,F20.2,F15.2)
C   END

```

The version, for the gnu compiler, showing some graphics as in the old days is:-

```

! *** simple FORTRAN program for a horizontal sundial ***
!   web site:          www.illustratingshadows.com
!   uses a free gfortran http://gcc.gnu.org/fortran/

! variables of general interest for the program
real rlat, rlng, rref          ! dial location
real radlat, radhra          ! radian versions
real hrcorr, hra, q          ! hr corr and hour angle
real sinlat, tanhra, atanhla, datanhla, tempshr ! working variables
integer i, j, k, iref, x, y, xx, yy ! integer ref.long due to type
character pause

! variables for the crt or printer display of the hour lines
character, dimension(22,78)::crt ! crt array
integer xs, ys, crtx, crty
character*80 c
character cc
real aspect

! format the crt or printer area
! http://orion.math.iastate.edu/burkardt/papers/fortran\_arrays.html
y = 1
do while ( y < 23)
  x = 1
  do while ( x < 79)
    crt(y,x) = '.'
    x = x + 1
  end do
  y = y + 1
end do
xs = 78
ys = 22
aspect = 2.10

! some blank lines to emphasize the messages that follow
i = 0
do while ( i .lt. 20 )
  write (*,70) " "
  i = i + 1
end do

write (*,70) "**** hDialFortranOGr ~~ Tested Win8 1/22/14 ****"
write (*,70) " "
write (*,70) " "
write (*,70) "**** LATITUDE MUST HAVE AT LEAST 1 DECIMAL ****"
write (*,70) "**** LONGITUDE MUST HAVE AT LEAST 1 DECIMAL ****"
write (*,70) " "

write (*,70) "Enter latitude      as nn.nn or nn.n" ! real latitude
read 80, rlat
if ( rlat < 1 ) then
  write (*,71) "DOUBLE CHECK LATITUDE, IT SEEMS TOO LOW", rlat
  write (1,71) "DOUBLE CHECK LATITUDE, IT SEEMS TOO LOW", rlat
end if

write (*,70) "Enter longitude     as nnn.n " ! real longitude
read 80, rlng

write (*,70) "Enter legal meridian as nnn " ! easier data entry eg 105
read 81, iref
rref = iref
hrcorr = 4 * (rlng - rref) / 60 ! deduce corr in hours

! get some common material computed
radlat = (rlat * 2 * 3.1416) / 360 ! radians of latitude
sinlat = sin(radlat) ! sine of latitude
radlng = (rlng * 2 * 3.1416) / 360 ! radians of longitude

write (1,71) "Latitude: ", rlat
write (1,71) "Longitude: ", rlng
write (1,71) "Reference: ", rref
write (1,71) "hour.corr: ", hrcorr
write (1,70) " "

write (*,71) "Latitude: ", rlat
write (*,71) "Longitude: ", rlng
write (*,71) "Reference: ", rref
write (*,71) "hour.corr: ", hrcorr
write (*,70) " "

write (1,70) "hour   hr.ang  ln.angle tan(ln.angle)"
write (1,70) "morning hours"
write (*,70) "hour   hr.ang  ln.angle tan(ln.angle)"
write (*,70) "morning hours"

i = -6 ! start at 0600 or 6 am
do while ( i .lt. 7) ! repeat until 6 pm
  call noon (i) ! have a noon separator

```

```

tempshr = i
hra = (tempshr*15) - (rlng-rref) ! get hour to REAL ! hour angle of the sun
radhra = (hra * 2 * 3.1416) / 360 ! radian version
tanhra = tan(radhra) ! tan of sun's hour angle
datanhla= (atan ( tanhra * sinlat )) * 360 / ( 2 * 3.1416)
write (1,50) i, hra, datanhla, tanhra ! btw: write 1 is a file
write (*,50) i, hra, datanhla, tanhra ! and write * is the console
call noon (i) ! have a noon separator
! build the crt or printer array
! =====
q = datanhla
if ( q >= -45 ) then
  if ( q <= 45 ) then
    y = 1
    do while ( y < ys )
      crty = y
      crtx = -y * aspect * ( tanhra * sinlat )
      xx = crtx + (xs/2)
      yy = crty
      if ( xx .gt. 0 .and. xx .lt. xs ) then
        crt(y,xs-xx) = '*'
      end if
      if ( i < 0 .and. i > -3 .and. xx > 0 .and. xx < xs ) then
        crt(y,xs-xx) = '\\ '
      end if
      if ( i .eq. 0 .and. xx .gt. 0 .and. xx .lt. xs ) then
        crt(y,xs-xx) = '| '
      end if
      if ( i > 0 .and. i .le. 3 .and. xx > 0 .and. xx < xs ) then
        crt(y,xs-xx) = '/ '
      end if
      y = y + 1
    end do
  end if
end if

! =====
! end crt building
i = i + 1 ! bump up the hour

end do

write (1,70) "afternoon hours"
write (1,70) " "
write (1,70) ("www.illustratingshadows.com SWS 2014")
write (*,70) "afternoon hours"
write (*,70) " "
write (*,70) ("www.illustratingshadows.com [enter to end]")

!read 70, pause

! print the crt array
y = 22
do while ( y .gt. 0)
  WRITE(1,72) (crt(y,x), x=1,78)
  y = y - 1
end do
write (1,71) "Aspect ratio: ",aspect
write (1,70) "*END*"
return

50 format (i4, f10.2, f7.1, f7.3)
60 format (f8.2)
70 format (A)
71 format (A, f8.2)
72 format (78a)
80 format (f8.2)
81 format (i3)

end

subroutine noon (i) ! sub routine process
  if ( i .eq. 0 ) then ! have a noon separator
    write (1,700) "noon"
    write (*,700) "noon"
  end if
  return
700 format (A)
end subroutine noon

! *** END ***

```

for output, see the next page...

