Subject: Re: number of seconds since 1/1/1970
Posted by Jackel on Tue, 14 Mar 1995 02:28:22 GMT
View Forum Message <> Reply to Message

In article <1995Mar13.174212.23087@newsserver.rrzn.uni-hannover.de>
knipp@ipi.uni-hannover.de (K Knipp) writes:

> I need a function to convert any given date (not the actual one)
> to the number of seconds elapsed since Jan 1st, 1970.

A while back I put together something like this. It does the
same thing as the UNIX function "gmtime", which seems to
be what you want. I'm including it below, please let me know
if there are any problems.

```
; Brian Jackel, University of Western Ontario Jackel@Canlon.Physics.UWO.CA
; All comments/bug reports/suggestions cheerfully accepted
;+
; NAME:  GMTIME
;
; PURPOSE: This function converts the time in seconds since
;  January 1 1970 to something more useful.
;
; CATEGORY: Timing
;
; CALLING SEQUENCE: Result= GMTIME( Timeinseconds )
;
; INPUTS:
; Timeinseconds a scalar value or array containing the number of
;   seconds since Jan 1 1970 (This should be a long
;   integer) OR a structure or array of structures of
;   type {GMTIME} (see below)
;
; KEYWORD PARAMETERS:
; ASCTIME  If this keyword is set, then the returned value
;   is a string (or array of strings) containing the
;   time in standard format (see OUTPUTS).
;
;
; OUTPUTS: The default behaviour for this function is to return
;  a structure of the form:
;
; {GMTime, sec:0B,  ;seconds 0-59
;   min:0B,  ;minutes 0-59
;   hour:0B,  ;hours 0-23
;   mday:0B,  ;day of month 1-31
```

```
;   mon:0B,  ;month 0-11
;   year:0,  ;years since 1900
;   wday:0B,  ;day of week 0-6 ,Sunday is 0
;   yday:0,  ;day of year 0-356
;   long_sec:0L} ;seconds since Jan 1 1970
;
; unless the keyword /ASCTIME was set, in which case the result will be
; of type string, with the form
;
;    Thu Jan 01 00:00:00 1970
;
; RESTRICTIONS: Won't work for negative times, ie. nothing before 1970.
;  No attempt is made to correct for daylight savings time,
;   or offset from UT.
;
; EXAMPLE:
; time0= SYSTIME(0)  ;current time, date etc.
; time1= SYSTIME(1)  ;time in seconds since Jan 1 1970
; gm= GMTIME( time1,/ASCTIME ) ;this should be within a second of
;     ;time0, NOT INCLUDING OFFSET FROM UT!
;
; MODIFICATION HISTORY: Brian Jackel March 27 1994
;   University of Western Ontario
;-


function GMTIME,Timeinseconds,ASCTIME=asctime

ON_ERROR,2
IF KEYWORD_SET(ASCTIME) THEN asctime=1 ELSE asctime=0

  gm= {gmtime, sec:0B, min:0B, hour:0B, mon:0B, year:0, wday:0B, mday:0B, yday:0, long_sec:0L
}

  siz= SIZE(Timeinseconds)
  CASE siz(siz(0)+1) OF
  0:MESSAGE,'Error- input value undefined'
  1:MESSAGE,'Warning- input value was of type BYTE',/INFORMATIONAL
  6:MESSAGE,'Error- input value was complex'
  7:MESSAGE,'Error- input value was a string'
  8:BEGIN
 structname= TAG_NAMES(Timeinseconds,/STRUCT)
 IF (structname EQ 'GMTIME') THEN BEGIN
  asctime=1
  gm= Timeinseconds
  goto,ASCTIME
 ENDIF ELSE MESSAGE,'Error- input value was a structure, not of type GMTIME'
    END
```

```
  ELSE:dummy=0
  ENDCASE

  IF (siz(0) EQ 0) THEN gm={gmtime} ELSE   $
   gm= MAKE_ARRAY( DIMENSION= siz( 1:siz(0) ), VALUE=gm )

  gm.long_sec= LONG(Timeinseconds)
;
;Strip off hours, minutes, seconds, since the start of a day
;
  secondsinday= 86400L   ;60 seconds/minute *60 minutes/hour *24 hours/day
  temp= gm.long_sec MOD secondsinday ;seconds since the start of a day
  gm.sec= temp MOD 60L
  temp= temp / 60L    ;full minutes since the start of a day
  gm.min= temp MOD 60L
  gm.hour= temp / 60L   ;full hours since the start of the day



;
;Strip off days, months, years
;
  temp= gm.long_sec / secondsinday  ;days since Jan 1 1970
  gm.wday= (temp+4) MOD 7   ;day of week (Jan 1 1970 was a Thursday=4)

  daysin4years= 1461L   ;3 regular years + 1 leap year = 365*3 + 366
  gm.year= 4 * FIX(temp / daysin4years) +70 ;70 years since 1900 plus however many 4 year
blocks
  temp= temp MOD daysin4years  ;days since the start of a 4 year period

  w= WHERE( temp GE 365 ,nw )  ;first year in a block will always have 365 days
  IF (nw GT 0) THEN BEGIN   ; ie. 1970, 1974, 1978..
   gm(w).year= gm(w).year +1
   temp(w)= temp(w) - 365

   w= WHERE( temp GE 365 ,nw )  ;second year will also have 365 days
   IF (nw GT 0) THEN BEGIN   ; ie. 1971, 1975, 1979...
    gm(w).year= gm(w).year +1
    temp(w)= temp(w) - 365

    w= WHERE( temp GE 366 ,nw )  ;but third year will be a leap year,
    IF (nw GT 0) THEN BEGIN   ; with 366 days
     gm(w).year= gm(w).year +1  ; ie. 1972, 1976, 1980...
     temp(w)= temp(w) - 366
    ENDIF

   ENDIF
  ENDIF
  gm.yday= temp  ;number of days since January 1 of the current year (0 to 365)
```

```
 daysinmonthlist= [31,28,31,30,31,30,31,31,30,31,30,31]
 month= 0
 daysinmonth= daysinmonthlist(month)
 w= WHERE( temp GT daysinmonth ,nw )
 WHILE (nw GT 0) DO BEGIN
   month= month+1
   temp(w)= temp(w)- daysinmonth
   gm(w).mon= month
   daysinmonth= daysinmonthlist(month)
   IF (month EQ 1) THEN daysinmonth= daysinmonth + ((gm.year MOD 4) EQ 0)
   w= WHERE( temp GT daysinmonth ,nw )
 ENDWHILE
 gm.mday= temp+1     ;day of the month (1 to 31)


ASCTIME: IF (asctime EQ 1) THEN BEGIN
 IF (siz(0) EQ 0) THEN timestring=' ' ELSE   $
  timestring= MAKE_ARRAY( DIMENSION= siz( 1:siz(0) ), VALUE=' ' )
 days= ['Sun','Mon','Tue','Wed','Thu','Fri','Sat']
 months=  ['Jan','Feb','Mar','Apr','May','Jun','Jul','Aug','Sep','Oct' ,'Nov','Dec']
 timestring= STRING(days(gm.wday),FORMAT="(A3)") +  $
  STRING(months(gm.mon),FORMAT="(X,A3)") +  $
  STRING(gm.mday,FORMAT="(X,I2.2)") +  $
  STRING(gm.hour,FORMAT="(X,I2.2,':')") +  $
  STRING(gm.min,FORMAT="(I2.2,':')")+  $
  STRING(gm.sec,FORMAT="(I2.2)")+   $
  STRING(gm.year+1900,FORMAT="(X,I4)" )
 return,timestring
ENDIF ELSE return,gm


END
```