DATESUBTRACT

Use this simple routine to calculate the difference between two dates and times.

DateSubtract takes the guesswork out of time-difference calculations. You can determine your billing cycle or initiate a Doomsday program. On the lighter side, you can figure out your age to the minute or set up your biorhythm. The answers are easy to find when you use this simple date subtraction routine.

USING THE PROGRAM

Listing 1 is a demonstration of my date subtraction routine. To use it, just enter the date and time for two dates using the form indicated. Be careful to use valid dates, since this demonstration doesn't check them. The program will display the number of days, hours and minutes that separate the two times.

ENTERING THE PROGRAM

Enter the BASIC program **Listing 1** and save it with the command:

```
SAVE DATESUBTRACT
```

For help with entering *Nibble* listings, see the Typing Tips section.

HOW THE PROGRAM WORKS

Program Flow

The program begins by dimensioning the array DE% and inserting the number of days in each month into the appropriate places of this array. Control then passes to the Data Entry routine at **line 460**. When the two dates and times are obtained, the Subtraction subroutine at **line 230** is called. This returns with variables D, H and M containing the days, hours and minutes, respectively, of the time difference to be put on-screen by **lines 520-600**.

Now, let's examine each part in turn.

Data Entry

Lines 460-500 accept two dates as input for strings A1$ and A2$ (earlier and later dates, respectively) and takes times on those dates as strings B1$ and B2$. A default null string for time is accepted if you are interested in days' difference only, and **line 350** checks for this before subtracting times. The entry routine checks only the length of the date string. You can add code to check against the valid ranges for the month and day. Actually, characters three and six of the date string can be anything except a colon or comma and are ignored in all calculations. You may prefer to insert your dates as DD-MM-YY or DD-MM-YYYY, in which case make the changes noted in **line 620**.

Times must be in 24-hour clock format. Illegal numbers such as 2500 for a time or 13 for a month are not checked, but you could easily add a check if you don't trust your initial entry.

Subtraction Subroutine

The subtraction is done in **lines 230-440**. First, the number of days' difference between each date and the beginning of its respective year is determined and assigned to variable A1 or A2. These figures are found by calling another subroutine in **lines 130-210**, where the number of days in previous months are added first, followed by the number of days in the current month. One day is added if the year is a Leap Year and the month is later than February (see the Leap Years section).

Back in the Subtraction subroutine, if the two dates are in different years (**line 250 or 260**), it's assumed that they're in adjacent years, and 365 days are added to A2 plus one more if it's a Leap Year. A2 now represents the number of days that Date 2 is from the beginning of year 1 in the assumed adjacent years. If the dates really are in adjacent years (tested by **line 310**, or by **line 420** if you used YYYY format), we pass on to **line 350** for subtraction. If they're not in adjacent years, then each year in between must be added to A2 in **lines 320-340**. This slows down the calculation considerably, as each year must be individually tested to discover whether it's a Leap Year before the program decides to add either 365 or 366 days. And the more years between the two dates, the longer it will take.

After this, A1 and A2 represent the number of days that Date 1 and Date 2 are from the beginning of the earlier year, and A1 is subtracted from A2 in **line 350**. If no times were entered (B1$ = ""), the program advances to **line 400** and returns from this subroutine to **line 520**.

Subtraction of times in **lines 360-400** is accomplished by first subtracting the hours (LEFT$ (B1$,2)), then the minutes (RIGHT$ (B1$,2)), with appropriate adjustments to days and hours, respectively, if negative differences occur.

The Leap Year

Determination of a Leap Year is needed in three places: First, in assigning the number of days from the beginning of the year for each date; second, in adding a year to Date 2 if the two dates are not in the same year; and last, in adding the days for each intervening year. This is determined by the subroutine in **line 110**. Since Leap Years occur every four years when the year is evenly divisible by four, the test is positive if YEAR/4 = INT (YEAR/4). There are some exceptions, however, because this rule assumes the earth orbits the sun every 365.25 days. Actually, the orbit is every 365.242 days, which would give us an excess of 3.2 days every 400 years. Our calendar accounts for this by including a Leap Year at the turn of the century only if the year is evenly divisible by 400 (this still leaves a discrepancy of 0.2 days for each 400 years, which remains uncorrected); hence, the second test in **line 110** for a year ending in "00". Note that if you chose to enter your year as YY instead of YYYY, this test will not be applied, and the century will be counted as a Leap Year.

Figure out your age to the minute or set up your biorhythm. . .

When exiting the Leap Year subroutine, Y = 0 if the year is not a Leap Year, and Y = 1 if it is a Leap Year; therefore, the addition of the 'Leap Year Determination' Y to 365 as in **line 330** leads to the correct number of days being added.

APPLICATIONS

If you want to use this routine in your own programs, you must first have the appropriate values in the array DE%. Then you need three subroutines: the Subtraction subroutine (**230-430**), the Leap Year subroutine (**110-120**), and the Days Assignment subroutine (**130-210**). Basically, this means inserting **lines 90-430** into your program.

For specific purposes, you may be able to shorten the Subtraction subroutine, and thus speed it up. If you always want your years as YY, for example, then all references to YYYY can be deleted (**lines 250, 300, 420, 430**). Conversely, if you always want YYYY, then remove **line 260** and substitute **lines 420** and **430** for **line 300-320**. If your years will never be more than one year apart, then **lines 320-340** are unnecessary. If you do not require times, **lines 360-390** and the test of B1$ in **line 350** can also be omitted.

Once these subroutines are in place, simply enter the Subtraction subroutine with your dates in A1$ and A2$, times in B1$ and B2$, and you will return with days' difference in D, hours' difference in H, and minutes' difference in M.

For a bullet proof program you will need to provide more sophisticated input range checking than this demonstration provides.

LISTING 1: DATESUBTRACT

```
10  REM  ************************
20  REM  *      DATESUBTRACT      *
30  REM  * BY  DAVID CARTWRIGHT *
40  REM  *  COPYRIGHT (C) 1987   *
50  REM  *  BY MICROSPARC, INC   *
60  REM  *  CONCORD,  MA 01742   *
70  REM  ************************
80  REM
90  DIM DE%(12):G$ =  CHR$ (7): POKE 33,40
100  DATA      0,31,28,31,30,31,30,31,31,30,31
     ,30,31: FOR I = 0 TO 12: READ DE%(I): NEXT
     : RESTORE : GOTO 460
110 Y = 0:YR =  VAL (Y$): IF YR / 4 =  INT (Y
     R / 4) THEN Y = 1: IF  RIGHT$ (Y$,2) = "
     00" AND  LEN (Y$) > 2 AND  INT (YR / 400
     ) <  > YR / 400 THEN Y = 0: REM  ** '00'
      IS 'ZERO ZERO'
120  RETURN : REM  ** LEAP YEAR DETERMINANT -
      IF YES Y=1; IF NO Y=0
130 A = 0
140  FOR I = 1 TO  VAL ( LEFT$ (A$,2))
150 A = A + DE%(I - 1): REM   ** ASSIGN VALUE
      FOR NO. OF DAYS EARLIER MONTHS THAT YEA
     R
160  NEXT
170 A = A +  VAL ( MID$ (A$,4,2)): REM   ** A
     DD DAYS THIS MONTH; A = DAYS FROM BEGINN
     ING OF YEAR
180 Y$ =  RIGHT$ (A$,2): IF  LEN (A$) = 10 THEN
     Y$ =  RIGHT$ (A$,4): REM   ** PREPARE FO
     R SUBROUTINE 110
190  GOSUB 110: REM  ** DETERMINE IF LEAP YEA
     R
200  IF  VAL ( LEFT$ (A$,2)) > 2 THEN A = A +
     Y: REM    ** IF MONTH > FEB ADD LEAP YEA
     R DETERMINANT
210  RETURN
220  REM   *** SUBTRACTION SUBROUTINE
230 A$ = A1$: GOSUB 130:A1 = A: REM    **A1 =
      DAYS FROM BEGINNING OF YEAR EARLIER DAT
     E
240 A$ = A2$: GOSUB 130:A2 = A: REM  ** A2 =
     DAYS FROM BEGINNING OF YEAR LATER DATE
250  IF  LEN (A1$) = 10 AND  VAL ( RIGHT$ (A2
     $,4)) >  VAL ( RIGHT$ (A1$,4)) THEN A2 =
     A2 + 365: GOTO 280: REM     ** IF DIFFER
     ENT YEARS ADD 365 TO A2 (LATER DAYS) AND
      GOTO 280
260  IF  VAL ( RIGHT$ (A2$,2)) >  VAL ( RIGHT$
     (A1$,2)) THEN A2 = A2 + 365: GOTO 280: REM
        ** AS FOR 250
270  GOTO 350: REM   ** SAME YEAR
280 Y$ =  RIGHT$ (A1$,2): IF  LEN (A1$) = 10 THEN
     Y$ =  RIGHT$ (A1$,4): REM   ** AS FOR 18
     0
290  GOSUB 110:A2 = A2 + Y: REM   ** LEAP YEA
     R DETERMINANT ADDED TO LATER DAYS
300  IF  LEN (A2$) = 10 THEN 420: REM   ** IF
      YEAR AS '1985' INSTEAD OF '85'
310  IF  VAL ( RIGHT$ (A2$,2)) -  VAL ( RIGHT$
     (A1$,2)) <  = 1 THEN 350: REM   ** IF <
     1 YEAR DIFFERENCE GOTO 350
320  FOR I =  VAL ( RIGHT$ (A1$,2)) + 1 TO  VAL
     ( RIGHT$ (A2$,2)) - 1
330 Y$ =  STR$ (I): GOSUB 110:A2 = A2 + 365 +
     Y: REM  ** ADD A YEAR + LEAP YEAR DETERM
     INANT (Y)
340  NEXT
350 D = A2 - A1: IF B1$ = "" THEN 400: REM
       **** D = TIME DIFFERENCE IN DAYS
360 H =  VAL ( LEFT$ (B2$,2)) -  VAL ( LEFT$
     (B1$,2)): REM    ** SUBTRACT HOURS ; H =
      DIFFERENCE
370  IF H < 0 THEN H = 24 + H:D = D - 1: REM
        ** IF NEG DECREMENT DAYS BY 1 & ADD 2
     4 TO HOURS
380 M =  VAL ( RIGHT$ (B2$,2)) -  VAL ( RIGHT$
     (B1$,2)): REM    ** SUBTRACT MINUTES ; M
      = DIFFERENCE
390  IF M < 0 THEN M = 60 + M:H = H - 1: REM
       ** IF NEG DECREMENT HOURS BY 1 & ADD 6
     0 TO MINUTES
```

LISTING 1: DATESUBTRACT *(continued)*

```
400  RETURN
410  REM  ***
420  IF  VAL ( RIGHT$ (A2$,4)) -  VAL ( RIGHT$
     (A1$,4)) <  = 1 THEN 350
430  FOR I =  VAL ( RIGHT$ (A1$,4)) + 1 TO  VAL
     ( RIGHT$ (A2$,4)) - 1: GOTO 330
440  REM  ***
450  REM   *** DATA ENTRY
460  HOME : PRINT "DATESUBTRACT BY DAVID CART
     WRIGHT": PRINT "COPYRIGHT 1987": PRINT "
     MICROSPARC,INC.": VTAB 5: CALL  - 958: PRINT
     "ENTER DATES AS MM-DD-YY OR MM-DD-YYYY"
470  VTAB 7: INPUT "ENTER DATE 1 (EARLIER) ";
     A1$: IF  LEN (A1$) <  > 8 AND  LEN (A1$)
      <  > 10 THEN  PRINT G$: GOTO 470
480  VTAB 9: CALL  - 958: INPUT "ENTER TIME O
     N DATE 1 (E.G. 0030) ";B1$: IF B1$ <  >
     "" AND  LEN (B1$) <  > 4 THEN  PRINT G$:
      GOTO 480
490  VTAB 11: CALL  - 958: INPUT "ENTER DATE
     2 ( LATER ) ";A2$: IF  LEN (A1$) <  >  LEN
     (A2$) THEN  PRINT G$: PRINT "MUST BE SAM
     E FORMAT AS EARLIER DATE, ": PRINT "(";A
     1$;")": PRINT : PRINT "RETURN TO CONTINU
     E";: GET A$: PRINT : GOTO 490
500  VTAB 13: CALL  - 958: INPUT "ENTER TIME
     ON DATE 2 (E.G. 1345) ";B2$: IF B2$ <  >
     "" AND  LEN (B2$) <  > 4 THEN  PRINT G$:
      GOTO 500
510  GOSUB 230
520  REM  ***
530  PRINT : PRINT "TIME DIFFERENCE IS :-": PRINT
     : IF D = 0 THEN  HTAB 8: GOTO 560
540  PRINT  TAB( 8)D;: IF D <  > 1 THEN  PRINT
     " DAYS";: GOTO 560
550  PRINT " DAY";
560  IF B1$ = "" THEN 610
570  PRINT ", ";H;: IF H <  > 1 THEN  PRINT "
      HOURS & ";: GOTO 590
580  PRINT " HOUR & ";
590  PRINT M;: IF M <  > 1 THEN  PRINT " MINS
     ": GOTO 610
600  PRINT " MIN"
610  PRINT : PRINT "RETURN TO CONTINUE";: GET
     A$: PRINT : GOTO 460
620  REM   *** FOR DD-MM-YY OR DD-MM-YYYY; CH
     ANGE MID$(A$,4,2) LINE 170 TO LEFT$(A$,2
     ) & CHANGE LEFT$(A$,2) LINES 140 AND 200
      TO MID$(A$,4,2)
```

END OF LISTING 1

KEY PERFECT 5.0
RUN ON
DATESUBTRACT

CODE-5.0	LINE# - LINE#	CODE-4.0
85B40409	10 - 100	73C4
8BC5B6C5	110 - 200	D573
D814D2FD	210 - 300	EA8D
CC8B403C	310 - 400	DA5B
A9EDC7E7	410 - 500	011911
C2BF0F96	510 - 600	564D
CD7CA12D	610 - 620	4380
0944C6F6	= PROGRAM TOTAL =	0A2C