ALGORITHM 199
CONVERSIONS BETWEEN CALENDAR DATE AND JULIAN DAY NUMBER

ROBERT G. TANTZEN
Air Force Missile Development Center, Holloman AFB, New Mex.

procedure $JDAY$ (d,m,y,j);
integer d,m,y,j;
comment $JDAY$ converts a calendar date, Gregorian calendar, to the corresponding Julian day number j. From the given day d, month m, and year y, the Julian day number j is computed without using tables. The procedure is valid for any valid Gregorian calendar date. When transcribing $JDAY$ for other compilers, be sure that integers of size 3×10^6 can be handled;
begin integer c, ya;
  if $m > 2$ **then** $m := m - 3$
    else begin $m := m + 9$; $y := y - 1$ **end**;
  $c := y \div 100$; $ya := y - 100 \times c$;
  $j := (146097 \times c) \div 4 + (1461 \times ya) \div 4 + (153 \times m + 2) \div 5 + d + 1721119$
end $JDAY$

procedure $JDATE$ (j,d,m,y);
integer j,d,m,y;
comment $JDATE$ converts a Julian day number j to the corresponding calendar date, Gregorian calendar. Since j is an integer for this procedure, it is correct astronomically for noon of the day. $JDATE$ computes the day d, month m, and year y, without using tables. The procedure is valid for any valid Gregorian calendar date. When transcribing $JDATE$ for other compilers, be sure that integers of size 3×10^6 can be handled:
begin $j := j - 1721119$;
  $y := (4 \times j - 1) \div 146097$; $j := 4 \times j - 1 - 146097 \times y$;
    $d := j \div 4$;
  $j := (4 \times d + 3) \div 1461$; $d := 4 \times d + 3 - 1461 \times j$;
    $d := (d + 4) \div 4$;
  $m := (5 \times d - 3) \div 153$; $d := 5 \times d - 3 - 153 \times m$;
    $d := (d + 5) \div 5$;
  $y := 100 \times y + j$; **if** $m < 10$ **then** $m := m + 3$
    else begin $m := m - 9$; $y := y + 1$ **end**;
end $JDATE$

procedure $KDAY$ (d,m,ya,k);
integer d,m,ya,k;
comment $KDAY$ converts a calendar date, Gregorian calendar, to the corresponding serial day number k. From the given day d, month m, and the last two decimals of the year, ya, the serial day number k is computed without using tables. The procedure is valid from 1 March 1900 $(k=1)$ to 31 December 1999 $(k = 36465)$. To obtain the Julian day number j (valid at noon) use $j = k + 2415079$;
begin if $m > 2$ **then** $m := m - 3$
    else begin $m := m + 9$; $ya := ya - 1$ **end**;
  $k := (1461 \times ya) \div 4 + (153 \times m + 2) \div 5 + d$
end

procedure $KDATE$ (k,d,m,ya);
integer k,d,m,ya;

comment $KDATE$ converts a serial day number k to the corresponding calendar date, Gregorian calendar. It computes day d, month m, and the last two decimals of the year, ya, without using tables. The procedure is valid from $k = 1$ (1 March 00) to $k = 36465$ (31 December 99) for any one century. For the 20th Century the relation between k and the Julian day number j (at noon) is $j = k + 2415079$;
begin $ya := (4 \times k - 1) \div 1461$; $d := 4 \times k - 1 - 1461 \times ya$;
  $d := (d + 4) \div 4$; $m := (5 \times d - 3) \div 153$;
    $d := 5 \times d - 3 - 153 \times m$;
  $d := (d + 5) \div 5$;
  if $m < 10$ **then** $m := m + 3$
    else begin $m := m - 9$; $ya := ya + 1$ **end**;
end $KDATE$

CERTIFICATION OF ALGORITHM 199 [Z]
CONVERSIONS BETWEEN CALENDAR DATE AND JULIAN DAY NUMBER [Robert G. Tartzen, *Comm. ACM 8* (Aug. 1963), 444].

DAVID K. OPPENHEIM (Recd. 10 Jul. 64 and 27 Jul. 64)
System Development Corp., Santa Monica, Calif.

Algorithm 199 was translated into JOVIAL J3 and tested on the Philco 2000. Input was generated with a random number generator that produced uniformly distributed dates between the years 1583 and 2583. The results were checked for 50 different dates in that range.

The procedures as written place unnecessary restrictions on some of the parameters. Expressions cannot always be used as inputs to the procedures. Also, the original input to $JDAY$, $JDATE$ and $KDAY$ will be modified during the operation of the respective procedures. It should also be noted that in many implementations of ALGOL the use of parameters called by name may be more expensive than those called by value. The call by name is a far more powerful tool than is necessary for most of the parameters of these procedures. For these reasons the following changes are suggested:

1. In **procedure** $JDAY$
   change: **integer** d, m, y, j;
   to: **value** d, m, y; **integer** d, m, y, j;
2. In **procedure** $JDATE$
   change: **integer** j, d, m, y; to: **value** j; **integer** j, d, m, y;
3. In **procedure** $KDAY$
   change: **integer** d, m, ya, k;
   to: **value** d, m, ya; **integer** d, m, ya, k;
4. In **procedure** $KDATE$
   change: **integer** k, d, m, ya;
   to: **value** k; **integer** k, d, m, ya;