

! Dolinsek Andreas

! Oster-Datum Berechnung nach Gauss

```
program easterDate
  integer year
  write(*, '(A/) ') 'Oster-Datum Berechnung nach Gauss'
  write(*, '(A$) ') 'Geben Sie ein Jahr ein:'
  read(*, '(I4)', err=10) year ! bei 4 abschneiden - im Jahresbereich bleiben
  call printEasterDate(year) laufruf subroutine
  goto 20
  write (*, '(A) ') 'Inputfehler'
end !programmende
```

! subroutine gibt das osterdatum des uebergebenen jahres aus

```
subroutine printEasterDate(year)
  integer M, N, year, day !redefine jahr
  integer a,b,c,d,e !hilfsvars fuer die Berechnung
  character*5 month ! April oder Maerz - beides 5 buchstaben
  month = 'Maerz' !defaultmaesig Maerz
  if(year .ge. 1700 .and. year .le. 1799) then ! kein select case
    M=23
    N=3
  else if(year .ge. 1800 .and. year .le. 1899) then
    M=23
    N=4
  else if(year .ge. 1900 .and. year .le. 2099) then
    M=24
    N=5
  else if(year .ge. 2100 .and. year .le. 2199) then
    M=24
    N=6
  else
    goto 20 ! kein gueltiges jahr
  end if
  ! Bibliotheksfunktionen verwendet...
  a = mod(year, 19)
  b = mod(year, 4)
  c = mod(year, 7)
  d = mod((19*a+M), 30)
  e = mod((2*b+4*c+6*d+N), 7)
  day = 22+d+e
  if(day .gt. 31) then ! nicht mehr im maerz?
    day = d+e-9 ! dann am April-Tag
    month = 'April'
    ! April hat aber randbedingungen ...
    if(day .eq. 26) day = 19 !26. April ist immer der 19. April zu setzen
    if(day .eq. 25 .and. d .eq. 28 .and. a .gt. 10) day = 18
    !statt 25. April ist dann der 18. April zu setzen, wenn d=28 und a>10.
  end if

  write(*,10) year, day, month !ausgabe mit format
  format('Ostern'1X,I4,1X,'ist am',1X,I2,'.',1X,A) !formatierung
  goto 30
  write(*, '(A//) ') 'Kein gueltiges Jahr!'
```

30 end subroutine