

```

program ej1
!/////////////////////////////////////////////////////////////////
!/
!// Programa para calcular la sucesión:
!// 1)  $x(n+1) = 2*x(n) + x(n-1) + x(n-2)$  si  $x(n)$  es par natural
!// 2)  $x(n+1) = x(n) + x(n-1)$  si  $x(n)$  es impar natural
!// 3)  $x(0) = 0, x(1) = 1$  condiciones iniciales
!// notar que la sucesión comienza con la 2da fórmula
!/
!// x1 impar ----> x (fórmula 2)
!// |x0|x1|x2|x3|...
!// x2 par -----> x (fórmula 1)
!/
!/////////////////////////////////////////////////////////////////

integer :: x
integer :: x0impar, xlimpar,x0par,x1par,x2par !x0par, x1par, x2par son x(n-2), x(n-1), x(n) caso par
                                                ! x0impar, xlimpar, son x(n-1), x(n) caso impar

! -- Instancio las cond. iniciales --
x0impar = 0
xlimpar = 1

x = xlimpar

print *, "           i   x(i)      nuevo x"
print *, "-----"

do i = 1,10

    if (mod(x,2)==0) then
        ! caso par
        x = 2*x2par + x1par + x0par
        ! actualizo x (x(i+1))
        ! debo actualizar las vars pares e impares para la prox iteración
        ! es importante el orden de actualización de las vars, primero
        ! las vars ximpar y luego las xpar iter (sino se pierde x0 y x1 impares)
        ! aparte tambien importa el orden de actualización de las impares (pares)solas
        ! la x0 x1 y x2 pares en ese orden (x0, x1 impares en ese orden)
        ! ver el diagrama de la secuencia arriba para entender las actualizaciones.
    else
        x = xlimpar + x0impar
        x0par = x0impar
        x1par = xlimpar
        x2par = x
        x0impar = xlimpar
        xlimpar = x
        print *, i+1, "es impar ",x
    end if

end do

end program ej1
-----


program ej2
integer :: nmax
real    :: suma , x

nmax = 10
suma = 0.0
x   = 1.0

do i = 0,nmax,2
    suma = suma + i*cos(i*x)
end do

do i = 1,nmax,2
    suma = suma + i*sin(i*x)
end do

print *, "La suma aproximada es: ", suma
end program ej2
-----
```

```

program ej3
!/////////////////////////////////////////////////////////////////
!/
!// programa para calcular la solucion aprox de la ec. dif.    /
!//   x'(t) = x(t)      ( soluciÃ³n: x(t)=exp(t) )           /
!// Una aprox numerica en pasos 'n' discretos del tiempo t     /
!// separados por un valor finito (pequeÃ±o) 'h' es:            /
!//   x(n+1) = x(n) + h*x(n)          t=n*h                  /
!/
!/////////////////////////////////////////////////////////////////
integer :: nmax
real    :: h,x,xExacta

h    = 0.01
nmax = 100
x   = 1.0        ! valor inicial en t=0 , n=0
xExacta = exp(0.0)

print *, "xAprox      xExacta"
print *, "-----"

do i = 1,nmax
  print *, x, xExacta
  x = x + h*x    ! aprox. x(i+1)
  xExacta = exp(i*h)
end do

end program ej3
-----
```