

## Previo Examen

Hacer el siguiente programa

1. Un programa que calcule la multiplicación de dos matrices como se muestra en el ejemplo. El usuario ingresa los valores de las matrices. Los resultados deben mostrar las matrices ingresadas y la multiplicación. Números enteros

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}_{2 \times 2} ; B = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{pmatrix}_{2 \times 3}$$

$$A \cdot B = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}_{2 \times 2} \cdot \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{pmatrix}_{2 \times 3} = \begin{pmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \end{pmatrix}_{2 \times 3}$$

$$A \cdot B = \begin{pmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \end{pmatrix}_{2 \times 3} = \begin{pmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} & a_{11}b_{13} + a_{12}b_{23} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} & a_{21}b_{13} + a_{22}b_{23} \end{pmatrix}_{2 \times 3}$$

1. Enviar código fuente por correo antes de las 10 pm del jueves 26 de octubre.
2. Traerlo impreso el día del examen, con líneas de código numeradas.

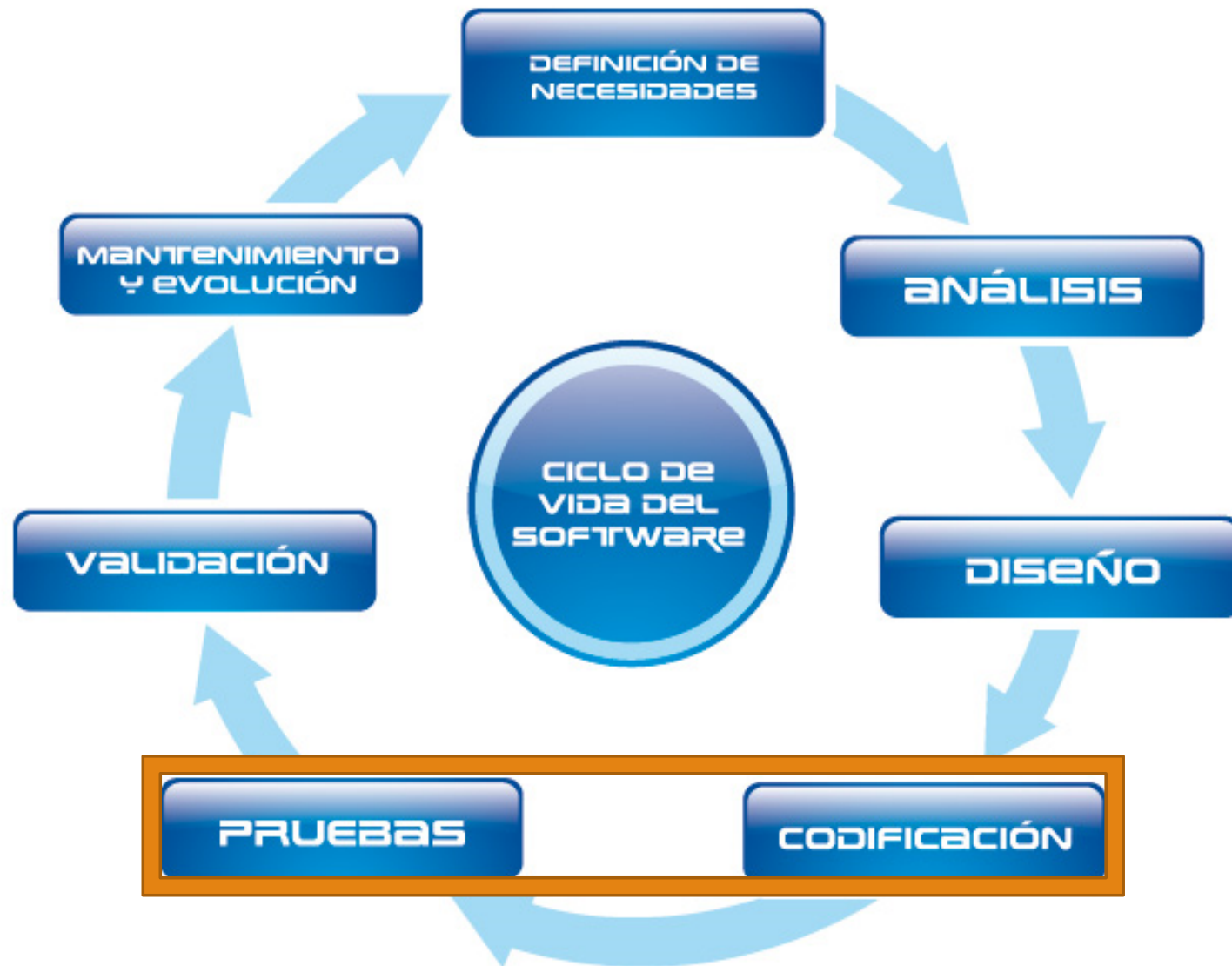
Cuenta 30% del examen.

# Programación básica

---

## PRÁCTICA 12. FUNCIONES

# ¿Dónde estamos en el ciclo de vida del software?



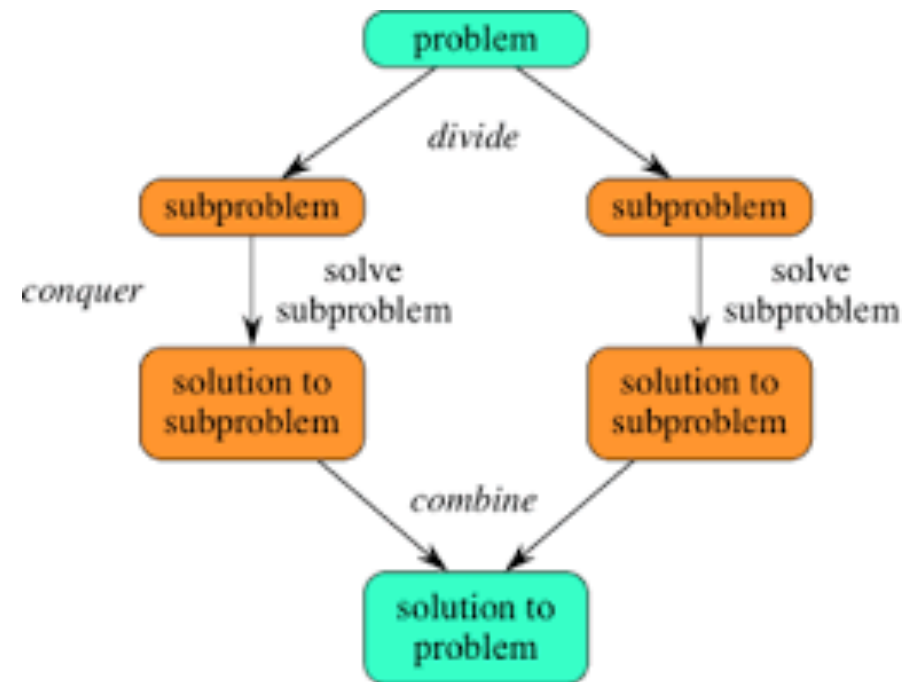
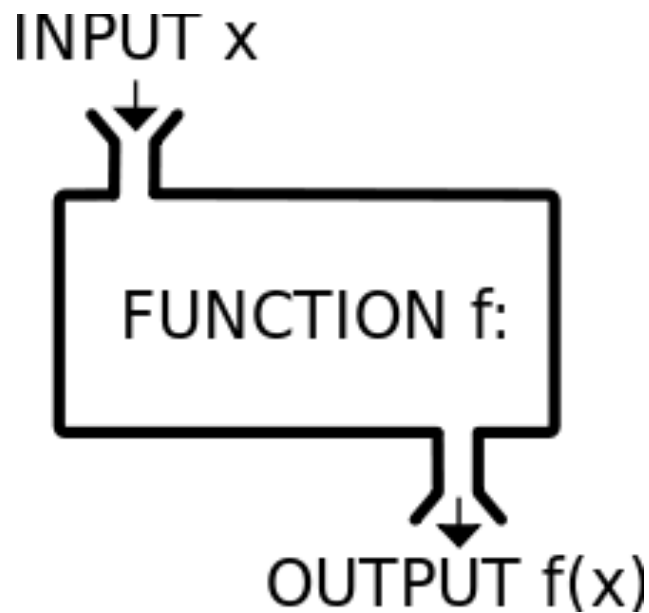
# Objetivo de la práctica

---

- ▶ Elaborar programas en lenguaje FORTRAN que permitan dividir la solución del problema en funciones
- ▶ Actividades
- ▶ Distinguir lo que es el prototipo de una función y la implementación o definición de ella..
- ▶ Utilizar parámetros tanto en la función principal como en funciones secundarias .

# Funciones

## “Divide y vencerás”



# Funciones y subrutinas

Fortran permite tener dentro de un archivo fuente varias funciones,, esto con el fin de dividir las tareas y que sea más fácil la depuración, la mejora y el entendimiento del código.

FORTRAN provee dos maneras de implementar subprogramas:

- creando **funciones** o
- creando **subrutinas**.

Estos subprogramas pueden ser de dos tipos:

- ❖ **intrínsecos**
- ❖ **externos**

# Funciones y subrutinas

Los subprogramas intrínsecos son aquellos que provee el compilador,, algunos de los más comunes son:

abs	valor absoluto
min	valor mínimo
max	valor máximo
sqrt	raíz cuadrada
sin	seno
cos	coseno
tan	tangente
atan	arco tangente
exp	exponente (natural)
log	logaritmo (natural)

# Funciones y subrutinas

Por otro lado,, los subprogramas externos son aquellos escritos por el usuario (o bien pueden formar parte de una biblioteca desarrollados por terceros).

## Función (sintaxis)



valorRetorno nombre ( *parámetros* )  
    ! bloque de código de la función  
END



# Funciones y subrutinas

Tres elementos (prototipo de función):

1. El nombre de la función: El identificador con el que llamaremos a la función
2. Los parámetros que recibe la función: Los datos de entrada a la función. (opcionales)
3. El valor de retorno de la función: El tipo de dato que regresa la función

```
valorRetorno nombre ( parámetros )  
    ! bloque de código de la función  
END
```

---

program principal

! Programa que permite calcular los valores de un polinomio  
! de grado 2, es decir, del tipo  $A*x^2 + B*x + C$

```
real sum, x, coef
integer m

write(*,*) 'Escriba el valor de x'
read(*,*) x

sum = 0.0
do 10 m = 0, 2
    write(*,*) 'Escriba el valor del coeficiente de x', m
    read(*,*) coef
    sum = sum + (coef * potencia(x, m))
10 continue
write (*,*) 'El valor del polinomio valuado en ', x, 'es', sum

stop
end

real function potencia(x,m)
    real x
    integer m
    potencia = 1.0
    do i = 1, m
        potencia = potencia * x
    enddo

    return
end
```

## Actividad de la práctica (en equipo de 4 máximo)

El programa de la multiplicación de dos matrices tiene varios bloques de código que podrían usarse en funciones.

Por ejemplo el pedir la información para llenar una matriz y el imprimir la información que contiene.

Hacer el prototipo y probar la función para el llenado de una matriz y de impresión de una matriz. Modificar su programa para que use estas funciones.

```
valorRetorno nombre ( parámetros )  
    ! bloque de código de la función  
END
```