

---

# Funciones y procedimientos.

---

# Contenidos:

---

- 1)Justificación
- 2)Tipos de datos
- 3)Rutina y tipos de rutinas.

# Justificación

---

- Cuando un programa comienza a ser largo y complejo no es apropiado tener un único texto con sentencias una tras otra. => La mayoría de los problemas son de este tipo.
- El principal motivo de su uso es que no se comprende bien qué hace el programa debido a que se intenta abarcar toda la solución a la vez. => El programa se vuelve difícil de leer, de modificar y de mantener.
- Además suelen aparecer trozos de código muy similares entre sí repetidos a lo largo de todo el programa.

# Justificación

---

## ¿Solución?

- Los lenguajes de alto nivel suelen disponer de una herramienta que permite estructurar el programa principal en pequeños subprogramas o **rutinas**, las cuales, resuelven problemas parciales del problema principal.
- A su vez, cada uno de estos subprogramas puede estar resuelto por otro subconjunto de programas o problemas parciales.
- Los **procedimientos y las funciones** son mecanismos de estructuración que permiten ocultar los detalles de la solución de un problema y resolver una parte de dicho problema en otro lugar del código.

# Rutina

---

PseInt permite definir "subrutinas" (o "funciones") dentro del pseudocódigo, desde la versión del 10 de octubre de 2012.

En su caso, se llaman "subprocesos" o "subalgoritmos".

**SubProceso** ejemploRutina

**FinSubProceso**

**SubAlgoritmo** ejemploRutina2

**FinSubAlgoritmo**

# Rutina: Ejemplo

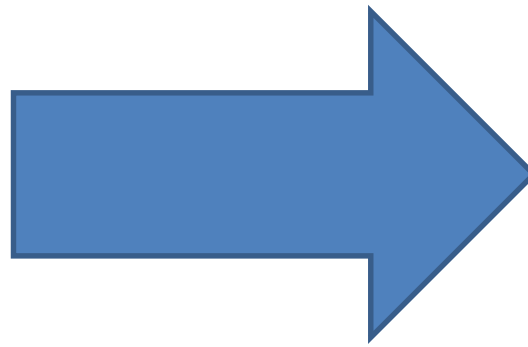
**Proceso** LasFuncionesReducenElPrograma

```
Definir x Como Entero;
Definir cantidad como Entero;
Leer x;

cantidad <- x;
Escribir "Primera repetición";
Mientras cantidad >0 Hacer
    Escribir '*';
    cantidad <- cantidad - 1;
FinMientras

cantidad <- x;
Escribir "Segunda repetición";
Mientras cantidad >0 Hacer
    Escribir '*';
    cantidad <- cantidad - 1;
FinMientras

cantidad <- x;
Escribir "Tercera repetición";
Mientras cantidad >0 Hacer
    Escribir '*';
    cantidad <- cantidad - 1;
FinMientras
FinAlgoritmo
```



**Algoritmo** LasFuncionesReducenElPrograma

```
Definir x Como Entero;
Leer x;
Escribir "Primera repetición";
Asteriscos(x);

Escribir "Segunda repetición";
Asteriscos(x);
...
Escribir "Tercera repetición";
Asteriscos(x);
FinAlgoritmo

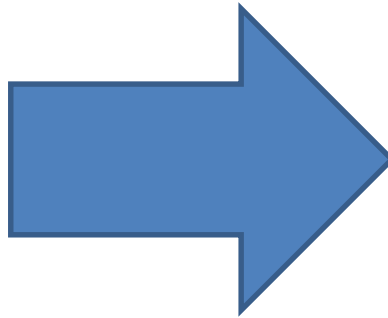
SubProceso Asteriscos (cantidad)
    Escribir "Esto es una rutina";
    Mientras cantidad >0 Hacer
        Escribir '*';
        cantidad <- cantidad - 1;
    FinMientras
FinSubProceso
```

# Rutina: Parámetros

---

En la mayoría de lenguajes de programación se puede indicar detalles adicionales ("**parámetros**") para que se puedan utilizar desde dentro de esa subrutina.

```
SubProceso  Asteriscos (cantidad)
  Escribir  "Esto es una rutina";
  Mientras cantidad > 0 Hacer
    Escribir '*';
    cantidad <- cantidad - 1;
  FinMientras
FinSubProceso
```



Un **parámetro**, generalmente es un valor o una variable de un tipo dado que se define en nivel superior o anterior. Pueden ser utilizados dentro de la rutina, sin verse afectado su valor externamente.

# Rutina: Retorno

---

Podemos crear subprocesos que realicen ciertas operaciones aritméticas y **devuelvan un resultado**. En PseInt se puede hacer con la misma palabra "subproceso" que hemos empleado hasta ahora, pero muchos lenguajes de programación distinguen entre un "procedimiento" o "subrutina".

Cuando se da una serie de pasos pero no se devuelve ningún valor => **procedimiento**  
Y en caso contrario => **función**



# Rutina: Retorno

---

**Algoritmo** LasFuncionesReducenElPrograma

Definir A Como Entero

Definir B Como Entero

Leer A

Leer B

Escribir "La suma de ",A," + ",B," es: ",Suma(A,B)

**FinAlgoritmo**

**SubProceso** retorno <- Suma ( n1, n2)

Definir retorno Como Entero

retorno <- n1 + n2

**FinSubProceso**

# Ejercicios

---

- 1.- Escribe una función en pseudocódigo que devuelva el área de un círculo dado su radio. Recuerda que para calcular el área, la fórmula es  $\Pi * r^2$ .
- 2.- Escribe una función en pseudocódigo que devuelva el resultado de un número elevado a un exponente dado.
- 3.- Escribe una función en pseudocódigo que devuelva el factorial de un número N (siendo N un número entero positivo). El factorial de 5 es el resultado de  $5*4*3*2*1$ .

Haz después un algoritmo principal que pida por teclado dos números y escriba en pantalla el factorial de cada número comprendido entre los dos pedidos.

# Ejercicios

---

**4.-** Escribe una función en pseudocódigo llamada `ConversionMoneda (...)` que permita convertir una cantidad de dinero dada en dólares, libras o yenes a euros. Estas son las equivalencias aproximadas:

- 1 libra=1,22 euros
- 1 dolar=0,75 euros
- 1 yen=0.009 euros

**5.-** Los empleados de una fábrica trabajan en dos turnos, Diurno y Nocturno. Se desea calcular el jornal diario de acuerdo a con las siguientes reglas:

La tarifa de las horas diurnas es de 10 €

La tarifa de las horas nocturnas es de 13,5 €

En caso de ser festivo, la tarifa se incrementa en un 10% en caso de turno diurno y en un 15% para el nocturno.

Escribe una función en pseudocódigo llamada `jornal(...)` que tome como parámetros el nº de horas, el turno y el tipo de día ("Festivo", "Laborable") y nos devuelva el sueldo a cobrar. Escribe también un algoritmo principal que pida el nombre del trabajador, el día de la semana, turno y nº de horas trabajadas, nos escriba el sueldo a cobrar usando la función anterior.

Ten en cuenta, que en la función nos pide el tipo de día pero en el algoritmo le pedimos al día es decir, que debemos saber si el día que introduce el usuario es festivo o no. Utiliza notación flexible.