

MODULARIZACIÓN

INTRODUCCIÓN A LA PROGRAMACIÓN

DESCOMPOSICIÓN ALGORÍTMICA: MODULARIZACIÓN

- Los **algoritmos complejos** son difíciles de afrontar
- Es más fácil aplicar *divide y vencerás*: fragmentar un problema en problemas más pequeños (*subalgoritmos*) y componer una solución final
- ¿Cuándo usar **subalgoritmos**?
 - Problemas complejos
 - Si tenemos que repetir operaciones análogas (reutilización)
- Al proceso de descomposición se le llama **modularización**

TIPOS DE SUBALGORITMOS O MÓDULOS

- **Funciones:** es un módulo o subalgoritmo que al ser evaluado devuelve un único resultado (por ejemplo, un valor numérico)
- **Procedimientos:** es un módulo o subalgoritmo que al ser evaluado no devuelve un valor, pero puede producir otros efectos o ayudar a fragmentar el problema (por ejemplo, mostrar las opciones del menú de la aplicación)

FUNCIONES

- Devuelven un **único resultado**
- El resultado puede ser numérico, textual o lógico
- Pueden necesitar algunos **valores de entrada** para realizar su cometido
- Ejemplo: función que devuelva el mcd de dos números
FUNCIÓN mcd(a: numérico, b: numérico): numérico

COMENZAR

PASOS DEL ALGORITMO DE EUCLIDES

DEVOLVER resultado

FIN FUNCIÓN

FUNCIONES

- Estructura general en pseudocódigo

FUNCIÓN nombre(lista de parámetros formales) : tipo de resultado

COMENZAR

Declaración de variables

Acciones

DEVOLVER valor

FIN FUNCIÓN

FUNCIONES

- Ejemplo: devolver el máximo de dos números

FUNCIÓN maximo(num1: numérico, num2: numérico) : numérico

COMENZAR

SI num1 >= num2 ENTONCES

DEVOLVER num1

SI NO

DEVOLVER num2

FIN SI

FIN FUNCIÓN

Parámetros formales



FUNCIONES

- Las funciones se pueden usar como un paso de otras funciones
- *Ejemplo: decidir un trabajo según el salario máximo*

ALGORITMO: “Decidir puesto de trabajo según salario máximo”

COMENZAR

VARIABLES salario1 (numérica), salario2 (numérica), max (numérica)

max = MAXIMO(salario1, salario2) ← Parámetros actuales

IMPRIMIR “Yo me quedaría con el trabajo cuyo salario es” max

FIN

PROCEDIMIENTOS

- Módulo o subalgoritmo que **agrupa** un conjunto de acciones o pasos, dándole un **nombre** por el que se puede **identificar**.
- Es posible que también necesite algunos **valores de entrada** para realizar su cometido.
- En un procedimiento podemos tener llamadas a otros **procedimientos** o **funciones anidadas**.

PROCEDIMIENTOS

- Estructura general en pseudocódigo

PROCEDIMIENTO nombre(lista de parámetros formales) COMENZAR

Declaración de variables

Acciones

FIN PROCEDIMIENTO

PROCEDIMIENTO

- Ejemplo: imprimir la tabla de multiplicar de un número

PROCEDIMIENTO tabla_multiplicar(num: numérico)

COMENZAR

VARIABLES: resultado (numérica)

PARA i DESDE 0 hasta 10 HACER

 resultado = num * i

 IMPRIMIR num "x " i "=" resultado

FIN PARA

FIN PROCEDIMIENTO