

## Práctica 4 – Pseudocódigo IV

En esta práctica definiremos algoritmos sencillos con modularidad en pseudocódigo y diagramas de flujo.

### Práctica

1. Diseña un algoritmo con una función de nombre esParMultiploTres que retorne verdadero o falso según el número que se le pase como parámetro sea o no un número par y a la vez múltiplo de 3. El mensaje que muestre si es o no par y múltiplo de 3 no lo debe escribir la función, sino el programa principal.
2. Diseña un algoritmo que muestre por pantalla un menú al usuario con cuatro opciones (Introducir números, Sumar números, Restar números y Salir). Si el usuario selecciona la opción “Introducir números”, se ejecutará dos veces una función de nombre introducirNumero que le pedirá que introduzca un número del 1 al 50. Si el usuario selecciona “Sumar números” o “Restar números”, se ejecutará la función sumar o restar respectivamente, que sumará o restará los dos números introducidos por el usuario y mostrará el resultado por pantalla (Si el usuario selecciona estas opciones sin haber introducido anteriormente números, se le mostrará un error indicándole que debe introducirlos antes). El programa finalizará cuando el usuario seleccione la opción “Salir”.
3. Diseña un algoritmo con una función de nombre interesMensualAPagar que calcule los intereses y cuotas que tendrá que pagar un cliente por un préstamo. Los parámetros de la función serán el préstamo que se solicita en euros, el número de meses en que se pretende devolver el préstamo y el TAE, que es el tipo de interés anual en %. La función debe escribir por pantalla la cifra total de intereses a pagar, y retornar el valor de la cuota que deberá pagar cada mes el cliente, que mostrará por pantalla el programa principal.
4. Diseñar un algoritmo con una función que realice la conversión de una cantidad en euros a otras monedas del mundo. La función tendrá de nombre cambioMoneda y recibirá como parámetros la cantidad en euros a convertir y la moneda a la que se desea realizar la conversión, que podrá ser “dólar”, “libra” o “yuan” (*el cambio actual de 1 euro es 1,11 dólares, 0,83 libras y 7,79 yuanes*), retornando el cambio en la moneda que se indique.
5. Diseñar un algoritmo de un juego que pida al usuario que introduzca un número secreto del 1 al 50. A partir de entonces, le da la opción a otro usuario de intentar adivinar el número secreto, indicándole si el número secreto es mayor o menor que el número que ha introducido, y volviendo a pedir que intente adivinarlo. En caso de que acierte el número secreto, se le mostrará un mensaje de *¡Enhorabuena!* indicando en cuántos intentos lo ha conseguido; en caso de que no lo consiga en 4 intentos, se le mostrará por pantalla *Fin de la partida* y se le dirá cuál era el número secreto. Este



algoritmo utilizará dos funciones, la primera de ellas es introducirNumero, que pide al usuario que introduzca un número entre 1 y 50 hasta que lo introduzca correctamente, y que utilizaréis tanto para pedir el número secreto como para pedir posteriormente que intente adivinarlo; la segunda función es comprobarNumero, que recibe como parámetros el número que ha introducido el usuario y el número secreto, y retorna 0 si los dos números son iguales, -1 si el número secreto es menor que el número introducido, y 1 si el número secreto es mayor que el introducido.

6. Diseñar un algoritmo del juego “Piedra, papel o tijera”. El juego consistirá en cinco rondas, donde el usuario tendrá que seleccionar si elige Piedra, papel o tijera, se invocará la función jugarRonda que recibirá como parámetro la elección del jugador y realizará la elección de “la máquina” de manera aleatoria, retornando el ganador de la ronda o si ha sido empate. El programa principal irá mostrando cómo va la partida después de cada ronda y al final de las 5 rondas mostrará quién es el ganador, si el jugador o “la máquina”. Para calcular lo que elige “la máquina”, deberás utilizar una función que no programes tú, incluida en PseInt, que es la función azar (recibe un parámetro numérico, y retorna un número aleatorio entre 0 y el número introducido menos uno – es decir, si llamas a la función azar con un 3, te retornará 0, 1 o 2)
7. Diseñar un algoritmo que dibuje por pantalla una pirámide invertida con asteriscos, con un procedimiento de nombre piramideInvertida que la dibuje tras pasarle los pisos de altura de la pirámide como parámetro.

```
*** Ejecución Iniciada. ***
¿Cuántos pisos debe tener la pirámide invertida?
> 7
*****
*****
*****
*****
*****
***
*
```