



UNIDAD 2 - ESTRUCTURAS BÁSICAS DE CONTROL

Ejercicios de condicionales

1. Crea una función (método) que reciba como parámetros tres números enteros y devuelva el mayor de los tres números. Utilizar esa función para el calcular el mayor de cuatro números enteros. Sería así:

```
public int mayor (int a, int b, int c) { ... }
```

2. Programa que lea un número por teclado e indique si es múltiplo de 2, 3, 5 o 7.
3. Programa que lea un número positivo de un dígito por teclado y diga en letra a qué dígito corresponde (uno, dos, tres, cuatro, ...). Ejemplo: si se introduce el 1, se mostrará por pantalla el “uno”; si se introduce el 11, se mostrará el mensaje “Error: no es un número de un dígito”.
4. Crea una función que no devuelva nada (void) que, dado un importe en euros, nos indique el mínimo número de billetes y la cantidad sobrante que se pueden utilizar para obtener dicha cantidad. Ejemplo: 362 euros → 1 billete de 200, 1 billete de 100, 1 billete 50, un billete de 10, sobran 2 euros. Pista: hay que ir sacando la división entera, y el resto se lleva al siguiente billete.

Ejercicios de bucles

5. Programa que pida al usuario un número repetidamente hasta que esté entre 1 y 10.
6. Programa que sume los 100 primeros números enteros impares.
7. Función que calcule el **factorial** de un número entero positivo pasado como parámetro.
8. Programa que imprima la tabla de multiplicar de cualquier número entre el 1 y el 10 introducido por teclado.
9. Crea un método que, dado un número entero pasado como parámetro, devuelva cuántos dígitos lo forman. P.ej.: el número 54326 consta de 5 dígitos. Pista: cuántas veces se puede dividir entre 10.
10. Programa (utilizando **bucles**) que muestre por pantalla la siguiente figura. Debe funcionar para n = número de líneas, que se pida por teclado.

```
*
**
***
****
```

11. Programa (utilizando **bucles**) que muestre por pantalla la siguiente figura. Debe funcionar para n = número de líneas, que se pida por teclado.

```
  *
 **
***
****
```

12. Programa (utilizando bucles) que muestre por pantalla la siguiente figura. Debe funcionar para n = número de líneas, que se pida por teclado.

```
  *
 ***
*****
*****
```



Ejercicios con bucles y condicionales

13. Programa que muestre por pantalla los números del 1 al 100 sin mostrar los que sean múltiplos de 5.
14. Programa que lea dos números positivos e imprima por pantalla todos los números pares que estén entre los dos números dados. También debe indicar cuántos números pares hay en ese intervalo.
15. Programa que lea números hasta que le den el cero, y que diga cuál ha sido el mayor y cuál el menor de los números introducidos (sin tener en cuenta el cero).
16. Programa que lea por teclado una serie de números enteros hasta que introduzca el -1 y obtenga su media. Deberá mostrarla por pantalla.
17. Programa que lea un entero positivo introducido por el usuario y que muestre por pantalla todos sus divisores.
18. Diseña un método que indique si un número entero pasado como parámetro es o no un número primo (devuelve un boolean). Un número primo es aquel que tiene por divisores el 1 y el propio número, únicamente.
19. Programa que, dado un número entero positivo introducido por teclado, visualice por pantalla todos los números primos menores que él. Utiliza la función del ejercicio anterior. Ejemplo:

```
Dime un número: 15  
> Los primos menores de 15 son: 1,2,3,5,7,11,13
```

Ejercicios con control de excepciones (hay que controlarlas en la petición de datos de teclado)

20. Pedir un número entero positivo N. A continuación, introducir por teclado N números enteros. Mostrar la media de los N números introducidos, el mayor y el menor.
21. Pedir un número entre 1 y 999 y mostrarlo con las cifras al revés.
22. Pedir un número entre 1 y 9999 y decir si es capicúa.
23. Pedir el día, mes y año de una fecha y mostrar la fecha del día siguiente (sin años bisiestos). Hacer primero una **función** para determinar que la fecha introducida es correcta.
24. **Búsqueda binaria.** El usuario piensa un número, y el programa debe averiguarlo. Pide al usuario que piense un número entre el 1 y el 100. El programa mostrará un número por pantalla y preguntará al usuario si el número que ha pensado es menor, igual o mayor al número mostrado. Este proceso lo repetirá hasta encontrar el número que ha pensado el usuario. Al finalizar el proceso, mostrará el número de intentos probados hasta encontrar el número.

Ejercicios de ampliación

25. Escribe un método, de nombre **esFechaValida**, que reciba como parámetros tres valores enteros con el día, mes y año de una fecha y devuelva un valor booleano que indique si se trata de valores válidos para una fecha. Si puedes comprueba también que el año sea bisiesto.
26. Diseña un método que devuelva un String con un **password** adecuado generado aleatoriamente. Debe tener más de 8 caracteres, al menos una mayúscula y al menos un símbolo (#\$%&/()=?¿!)