

Tema 3. Estructuras de control.

Estructuras de selección

1. Escribe un programa *EstanEnOrden* que lea tres enteros y diga si están ordenados de menor a mayor, de mayor a menor o desordenados.
2. Escribe un programa *Binario* que lea tres números, en tres entradas independientes desde el teclado (del bit más significativo al menos significativo). Estos números deberán ser 0 o 1, de modo que si el usuario introduce otro número distinto a 0 o 1 el programa finalizará mostrando al usuario un mensaje de error. El programa debe tratar estos números como si se tratasen de un número binario de tres bits y mostrar su valor en decimal.

Nota: El número binario $b_2b_1b_0$ con $b = 0, 1$ en decimal se calcula como $2*2*b_2 + 2*b_1 + b_0$.

3. Cuatro números enteros entre 0 y 100 representan las puntuaciones de un estudiante de informática a lo largo del curso. Escribe un programa llamado *MiNota* que dados cuatro enteros entre 0 y 100, correspondientes a las notas de un estudiante de Informática a lo largo del curso, calcule la media de estas puntuaciones y muestre la nota calificación de la asignatura de acuerdo a los siguientes rangos:

Media	Calificación
90-100	A
80-89	B
70-79	C
60-69	D
0-59	E

4. Escribe un programa llamado *OperacionesAritmeticas* que genere dos números enteros aleatorios A y B entre 0 y 10 (ambos inclusive) y muestre las 5 operaciones aritméticas básicas +, -, *, / y % que pueden hacerse entre ellos en el sentido A op B. Utiliza para generar los números aleatorios el método `random()` de la clase `Math`.
5. Escribe un programa llamado *EcuacionCuadratica* que resuelva la ecuación de segundo grado $ax^2 + bx + c = 0$, donde $a \neq 0$.

Estructuras de iteración

1. Escribe un programa llamado *Primo* que muestre si un número dado por el usuario es primo o no.
2. Escribe un programa llamado *TablaAscii* que mediante un bucle imprima por pantalla los 128 primeros caracteres ASCII con el siguiente formato:

....

65:A

66:B

...

3. Escribe un programa llamado *Not* que reciba por teclado un número en binario de 8 bits y muestre por pantalla su complemento en forma binaria. El programa debe mostrar el resultado tanto en binario como en decimal. Por ejemplo:

NOT(10010001) = 01101110

4. Escribe un programa llamado *InvertirEntero* que invierta los dígitos de un entero dado por el usuario. Para ello convierte el número a cadena.
5. Escribe un programa llamado *MCD* que lea desde el teclado dos números enteros positivos y que calcule el máximo común divisor de ambos.

Para el cálculo del MCD utilizar el "algoritmo de Euclides", que consiste en dividir el mayor número por el menor, quedarse con el resto y transformar dicho resto en el número menor y el menor de antes en el mayor de ahora. Repetir el proceso hasta que el resto sea cero. en ese momento el menor número será el máximo común divisor.

Ejemplo MCD(532, 112):

- Paso 1: 532 / 112 → resto = 84
- Paso 2: 112 / 84 → resto = 28
- Paso 3: 84 / 28 → resto = 0 → FIN: MCD = 28

Métodos de utilidad para resolver los ejercicios planteados:

- **int parseInt(String s):** transforma la cadena s en entero (int).
- **String Sttring.valueOf(int i):** transforma el entero i en cadena.
- **char charAt(int i):** devuelve el carácter de la posición i de la cadena actual.
- **int length():** devuelve la longitud de la cadena actual

6. Escribe un programa *Triangulo1* que muestre lo siguiente:

```
1
1 2
1 2 3
1 2 3 4
```

7. Escribe un programa *Triangulo2* que muestre lo siguiente:

```
1 2 3 4
1 2 3
1 2
1
```

8. Escribe un programa *Triangulo3* que muestre lo siguiente:

```
1
1 2
1 2 3
1 2 3 4
1 2 3
1 2
1
```

9. Escribe un programa *Variaciones* que imprima por pantalla todas las posibles variaciones con repetición que pueden hacerse con 4 números. Serían las siguientes (con los números 1 2 3 4):

```
1 1 1 1
1 1 1 2
1 1 1 3
... ..
4 4 4 2
4 4 4 3
4 4 4 4
```