

Adquisición y Tratamiento de Datos I. Estructuras de control

1. Diseñar un programa que lea los coeficientes de un polinomio de segundo grado $ax^2 + bx + c = 0$ y calcule las soluciones del mismo teniendo en cuenta todas las posibles alternativas.

2. Hacer un programa que nos permita convertir grados Celsius (C), Fahrenheit (F), Kelvin (K), y Rankine (R). El programa nos preguntará en que unidades damos la temperatura de entrada y a que escala la queremos convertir. Para ello sabemos que:

$$K = C + 273.15$$

$$R = F + 459,67$$

$$C = \frac{5(F - 32)}{9}$$

3. Haz un programa que lea las coordenadas que definen un rectángulo (esquina superior y esquina inferior opuesta) y las coordenadas de un punto en el espacio. El programa deberá decidir si el punto está o no dentro del rectángulo.

4. Escribe un programa que lea tres números enteros por teclado y que devuelva el mínimo de los tres. Pero se hará considerando 3 posibilidades:

- Usar condicionales simples (“if” sin parte “else”)
- Usar condicionales compuestas (“if + else”) y anidamiento
- Usar sentencia switch

5. Nos hace falta un programa para establecer los descuentos en el precio de los billetes del tren siguiendo los siguientes criterios:

- Según la edad: Los menores de 4 años no pagan. Los que tienen 4 y 7 (ambos inclusive) sólo pagan la mitad del billete. Los mayores de 65 pagan el 40%.
- Los estudiantes pagan el 45% del billete
- Los miembros de familia numerosa pagan el 70%

Se tomarán los datos de entrada: precio del billete, la edad del cliente, si es o no estudiante y si es miembro de familia numerosa. El programa ha de mostrar el precio a pagar teniendo en cuenta que los descuentos no son acumulativos. En el caso de que a un cliente se le pueda aplicar más de un descuento se aplicará aquel que sea más beneficioso para él.

6. Diseñar un programa para la caja de un supermercado que lea un precio desde el teclado y una cantidad (se supone que ambas están por debajo de las 100 euros y que cantidad \geq precio) y obtenga en la pantalla el número mínimo de monedas de 2, 1, 0.5, 0.2 y 0.1 que se deben dar de cambio. Realizar el programa usando bucles y sin usar los operadores división entera ni módulo.

7. Necesitamos un programa para obtener la letra del NIF a partir del número del DNI. Para ello se debe leer el número y obtener el resto de dividirlo por 23. La letra asociada depende del resto obtenido siendo la equivalencia la siguiente:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
T	R	W	A	G	M	Y	F	P	D	X	B	N	J	Z	S	Q	V	H	L	C	K	E

8. Hacer un programa que lea los dígitos de un ISBN y calcule el dígito de control. Para calcular el carácter de control se debe multiplicar cada dígito por su posición (la posición 1 se encuentra a la derecha), sumar los resultados obtenidos y hallar el resto de dividir por 11. El resultado es el número de control; si el resto es 10 entonces se usa como carácter de control la 'X'. Ej.: 84-921873-8-9

9. Diseñar un programa para calcular la suma de los 100 primeros términos de la sucesión:

$$a_n = (-1)^n \frac{n^2 - 1}{2n + 1}$$

10. Realizar un programa para calcular los valores de la función

$$f(x, y) = \frac{\sqrt{x}}{y^2 - 1}$$

para los valores de (x,y) con x=0,...,48, 50 e y=-40,-39,...,39,40

11. Calcular mediante un programa en C la potencia x^n y la función factorial $n!$

12. Realizar un programa que imprima todos los números primos entre 1 y n, siendo n un número introducido por el usuario.

13. Diseñar un programa que calcule los divisores de un valor entero.

14. Diseñar un programa que halle la descomposición en números primos de un valor entero.

15. Leer números enteros por teclado hasta que se introduzcan 10 o hasta que se introduzca un número negativo. Imprimir la media.

16. Diseñar un programa para jugar a adivinar un número entre 0 y 100. El juego tiene que dar pistas de si el número introducido por el jugador está por encima o por debajo. El juego termina cuando se adivina el número o se decide terminar de jugar (por ejemplo al teclear un número negativo).

Nota: la función **int rand(void)** de **stdlib.h** genera un número aleatorio entero entre 0 y la constante **RAND_MAX**. Para generar un entero aleatorio entre 0 y 100, sería **rand()%101**.

17. Realizar el mismo ejercicio de antes pero permitiendo jugar tantas veces como lo desee el jugador.