

# Práctica 1.

## Ejercicio 1.

Implementar un programa para el cálculo del área y el perímetro de un rectángulo. (Hay que preguntar al usuario por la longitud de dos lados).

## Ejercicio 2.

Desarrollar un programa que pida la ficha de un alumno y la muestre por pantalla.

La ficha de un alumno está formada por su nombre, primer apellido, segundo apellido, NIF, sexo (H/M), edad y número de créditos matriculados.

Nótese que el número de créditos puede contener decimales. Ejemplo: 13.5 créditos.

El programa deberá declarar variables del tipo adecuado para almacenar cada uno de los componentes de la ficha, asignarles el valor que el usuario introduzca y por último mostrar toda la información almacenada.

## Ejercicio 3.

Una empresa de venta de recambios de automóviles necesita un programa que calcule y muestre el precio de venta en euros de los productos. Para ello, se debe aplicar la siguiente fórmula:

$$ganancia = precioCompra \cdot \frac{margen}{100}$$

$$precioVenta = precioCompra + ganancia$$

El precio de compra en euros y el margen de ganancias en tanto por ciento que desea obtener la empresa para el producto se introducirán por teclado. Como salida se deberá mostrar por pantalla el precio de venta del producto.

## Ejercicio 3.

Escribir un programa en lenguaje C que pida al usuario por teclado las coordenadas (x,y) de dos puntos (x1,y1), (x2,y2) y que calcule la distancia geométrica que hay entre ellos.

La distancia geométrica  $d$  entre dos puntos cumple:

$$d^2 = (x_2 - x_1)^2 + (y_2 - y_1)^2$$

Para calcular la raíz cuadrada de un número hay que incluir la cabecera de la librería matemática colocando al inicio del programa

```
#include <math.h>
```

La función para calcular raíces cuadradas es `sqrt(valor)`, que devuelve la raíz cuadrada de "valor". (Sugerencia: plantear primero el algoritmo en papel)

## Ejercicio 6.

Escribir un programa en lenguaje C que pida al usuario por teclado y por separado tres cadenas de texto: nombre, primer apellido y segundo apellido, para dos personas. Seguidamente deberá preguntar por el nombre de su hijo/a y con estos datos visualizar por pantalla el nombre completo del hijo/a (utilizando los primeros apellidos de sus padres).

## Ejercicio 7.

Desarrollar un programa que dado un carácter muestre su código ASCII y que dado un número entero muestre el carácter que tiene ese código ASCII.

A continuación se muestra un ejemplo de ejecución de este programa:

```
Introduce un caracter: F
```

```
El caracter 'F' tiene el codigo ASCII 70.
```

```
Introduce un codigo ASCII: 80
```

```
El codigo ASCII 80 se corresponde con el caracter 'P'.
```

## Ejercicio 8.

Escribir un programa en C que pida al usuario que elija un número del 1 al 9 y a partir del resultado de una serie de operaciones le diga su edad y el número que pensó.

Las operaciones a realizar son multiplicar la edad por 10 y restarle el número pensado multiplicado por 9. A partir del resultado es fácil obtener los datos de partida. El

número elegido es el dígito de las unidades del resultado, mientras que la edad del usuario se obtiene al sumarle a los otros dígitos este número.

Ejemplo de salida (pensando el número 8 y con una edad de 19):

Piensa un numero del 1 al 9.

Multiplica tu edad por 10 y restale el numero que has pensado multiplicado por 9.

Dame el resultado: 118

Tu edad es 19 y el numero que has pensado es el 8.

## **Ejercicio 9.**

Escribir un programa que pida al usuario una letra minúscula y una letra mayúscula. Posteriormente, se deberá escribir la letra minúscula en mayúsculas y la mayúscula en minúsculas.

Ejemplo de salida por pantalla:

Escribe una letra minuscula: r

Escribe una letra mayuscula: U

La letra 'r' en mayusculas es 'R'.

La letra 'U' en minusculas es 'u'.

## **Ejercicio 10.**

En un colegio se compran todos los años agendas para todos sus alumnos. A cada alumno se le deja elegir el modelo de agenda entre diferentes opciones. Las agendas se venden en cajas grandes (50 unidades), cajas medianas (20 unidades), cajas pequeñas (5 unidades) y sueltas. Como suele ser habitual, en los paquetes más grandes el precio por unidad resulta más económico que en los paquetes más pequeños.

Se desea tener un programa al que se le diga el número de alumnos del colegio que quieren un mismo modelo de agenda y que diga cuántas cajas hay que comprar de

cada uno de los tamaños y cuántas agendas sueltas, minimizando el precio. Para reducir el coste, hay que tratar de coger primero cajas grandes, luego medianas, luego pequeñas y por último agendas sueltas.

A continuación se da un ejemplo de ejecución del programa deseado:

Numero de alumnos para este modelo: 87

Para este modelo hay que comprar:

- cajas grandes: 1
- cajas medianas: 1
- cajas pequeñas: 3
- agendas sueltas: 2