

1. Realizad un programa de ordenador en lenguaje **C** que pida al usuario las coordenadas de dos puntos en el espacio, calcule la distancia euclídea entre ambos y la muestre por pantalla.
2. Realizad un programa de ordenador en lenguaje **C** que pida al usuario las coordenadas de un punto en el plano (x,y) e indique el ángulo que forma con el eje de coordenadas la recta que pasa por dicho punto y por el origen de coordenadas.
3. Realizad un programa de ordenador en lenguaje **C** que imprima en pantalla la tabla completa de códigos ASCII (desde 0 a 255). Se imprimirán 10 columnas con el ordinal y el carácter. Como los caracteres correspondientes a los ordinales 7 (campana), 8 (retroceso), 9 (tabulador horizontal), 10 (salto de línea), 11 (tabulador vertical), 13 (retorno) y 26 (subrayado) producen disfunciones, en lugar del carácter se listará, respectivamente, `\c`, `\b`, `\t`, `\r`, `\v`, `\n` y `\s`.
4. Realizad un programa de ordenador en lenguaje **C** que lea desde el teclado tres caracteres que son las cifras que forman un número en base hexadecimal (16). Se debe comprobar que son válidas (0..9, A, B, C, D, E, F, a, b, c, d, e, f) y convertirlo a base decimal, mostrando por pantalla "Hexadecimal xxx = yyyy decimal". El proceso se repetirá hasta dar el hexadecimal 000.
5. Realizad un programa de ordenador en lenguaje **C** que pida un mensaje al usuario (una línea de longitud máxima 100) y la longitud N de una ventana. Posteriormente, el programa imprimirá una serie de líneas por pantalla correspondientes a la ventana de longitud N que mostrará el mensaje desplazándose de izquierda a derecha. Por ejemplo, si introducimos el mensaje:

Vamos a aprobar

y longitud N=5, el programa mostrará por pantalla las líneas:

```
"Vamos"
"amos "
"mos a"
"os a "
"s a a"
" a ap"
"a apr"
" apro"
...
```

6. Realizad un programa de ordenador en lenguaje **C** que lea una serie de números enteros positivos de hasta 4 cifras. El programa dejará de leer cuando se introduzca un número negativo. Una vez leídos, el programa indicará
  - Cuántos números introducidos tenían un 0 entre sus cifras.
  - Cuántos números introducidos tenían un 1 entre sus cifras.
  - ...
  - Cuántos números introducidos tenían un 9 entre sus cifras.
  - El número de cifras 0 aparecidas en todos los números.

■ ...

- El número de cifras 9 aparecidas en todos los números.

Por ejemplo, si introducimos los números 123, 45, 10, 1001 la salida sería:

Hay 2 números con un 0 entre sus cifras.

Hay 3 números con un 1 entre sus cifras.

...

Hay 0 números con un 9 entre sus cifras.

La cifra 0 aparece 3 veces.

La cifra 1 aparece 4 veces.

...

La cifra 9 aparece 0 veces.

7. Realizad un programa de ordenador en lenguaje **C** que lea por teclado una matriz  $N \times M$  (los valores  $N$  y  $M$  son constantes mayores que 2 definidas en el programa) de enteros y que indique por pantalla todas las parejas distintas de filas que tengan algún elemento idéntico en una misma columna. Por ejemplo, si introducimos la matriz:

```
2 3 4 5
1 3 0 2
3 4 1 1
```

Las filas con elementos repetidos en alguna columna son:

Filas 0 y 1

8. Se desea rellenar un vector con la siguiente sucesión:

$$\begin{aligned}a(0) &= 1 \\a(1) &= 1 \\a(n) &= a(n-2) + a(n-1), n > 1\end{aligned}$$

Desarrollad un programa en lenguaje **C** que forme este vector hasta el valor  $n = 20$  (21 valores: desde  $a(0)$  hasta  $a(20)$ ) y luego lo muestre por pantalla, imprimiendo cada elemento de la sucesión en una línea diferente con el siguiente formato:

```
a(0) = 1
a(1) = 1
a(2) = 2
...
```

9. Desarrollad un programa en lenguaje **C** que lea del archivo `DATOS004.txt` pares de valores reales  $X$  e  $Y$  que representan puntos del plano y que escriba en pantalla si cada uno de los puntos está en o no en el semiplano limitado por  $y \geq 1 - x$ , siendo un ejemplo de la salida:

PUNTO	X	Y	Posición
1	3,00	4,00	DENTRO
2	3,00	3,00	DENTRO
3	5,00	5,00	DENTRO
4	5,00	0,00	DENTRO
5	0,00	-5,00	FUERA
6	6,00	0,00	DENTRO

10. La ACB nos pide que desarrollemos un programa en lenguaje **C** que gestione las faltas personales cometidas en un partido de baloncesto, entendiendo que cada equipo está compuesto por 10 jugadores. El programa debe constar de un menú que lleve a cabo estas operaciones:
1. Gestionar las faltas personales: Tras cada falta cometida, se debe preguntar el equipo y el número de jugador, incrementando en uno su número de faltas.
  2. Mostrar faltas por equipo: Se mostrará por pantalla el número de faltas cometidas por cada equipo en el instante de tiempo actual, sumando para ello el número de faltas de los jugadores de cada uno de ellos.
  3. Listar el número de faltas cometidas por los jugadores de cada equipo.
  4. Mostrar los jugadores que han sido expulsados: Mostrará por pantalla todos aquellos jugadores con 5 faltas personales cometidas.
  5. Mostrar los jugadores que no han cometido ninguna falta: Mostrará por pantalla todos aquellos jugadores que no han cometido ninguna falta personal.