

APELLIDOS y NOMBRE: _____

GRUPO: _____

TEST (1 punto):

- Marca con una cruz la respuesta correcta.
- Cada pregunta acertada vale 0.25 puntos.
- Cada pregunta incorrecta o no contestada vale 0 puntos (no restan).
- Contestad el test en estas mismas hojas.

1. ¿ Cuántos bits son necesarios, como mínimo, para representar la edad de una persona ?
(Supondremos que nadie vive más de 127 años).

- a) 6
- b) 8
- c) 7
- d) 5

2. ¿ Qué es un disco duro ?

- a) Un dispositivo de entrada.
- b) Un dispositivo de salida.
- c) Un dispositivo de almacenamiento secundario.
- d) Memoria RAM.

3. Señala la respuesta correcta si de lo que se trata de es mostrar 5 asteriscos por pantalla.

- a) `int i; for(i=1; i < 32; i *= 2) printf("*");`
- b) `printf("*****");`
- c) `int i; for(i=5; i > 0; i--) printf("*");`
- d) Todas las anteriores.

4. ¿ Qué definición de CONSTANTES es la correcta en lenguaje C ?

- a) `#define N=25;`
- b) `#define N 25`
- c) `int N=25;`
- d) `int N 25`

PROBLEMAS y CUESTIONES: (Se contestan en hojas aparte)

1. (1.5 puntos) Definid las estructuras necesarias para poder mantener en un vector el histórico de vuelos de un avión. Hasta una máximo de 100 vuelos.

Por cada vuelo se deben guardar los siguientes datos: identificador del avión, punto de partida, fecha y hora de salida, punto de llegada, y fecha y hora de llegada.

2. (1.5 puntos) Indicad la salida que muestra el siguiente programa por pantalla:

```
#include <stdio.h>

void segmenta( int a, int b )
{
    int      i;

    if ( a > b ) {
        for( i=a; i >=b; i-- ) printf( " %d ", i );
        printf( "\n" );
    } else {
        for( i=a; i <= b; i++ ) printf( " %d ", i );
        printf( "\n" );
    }
}

void main()
{
    segmenta( 5, 10 );
    segmenta( 15, 8 );
}
```

Sugerencia: Aunque sólo se pide lo que muestra por pantalla, el que no lo tenga muy claro que haga la traza.

3. (2 puntos)

Se pide la implementación de un programa en lenguaje **C** para rellenar un vector de **N** números enteros sin que se repita ningún valor. Donde **N** es una constante que podrá tomar cualquier valor positivo mayor que cero. El programa deberá controlar que cuando el usuario introduzca un nuevo valor éste no haya sido introducido previamente, si dicho valor ya existe en el vector el programa deberá volver a pedir otro hasta que el que se introduzca no exista.

Ejemplo:

```
V[0] ? 5
V[1] ? 4
V[2] ? 5
V[2] ? 7
V[3] ? _
```

4. (4 puntos) Realizad un programa en lenguaje **C** que nos permita disponer de las siguientes utilidades de conversión entre distintas monedas y el euro.

- Conversión de una cantidad en euros a la *divisa*.
- Conversión de una cantidad en la *divisa* a euros.
- Visualizar una tabla de conversión euros-*divisa*.
- Visualizar una tabla de conversión *divisa*-euros.

El programa deberá tener obligatoriamente un diseño modular (funciones), y sin la utilización de variables globales. Los datos sobre cada divisa se guardarán en un vector de tamaño **N**, donde **N** es el número de divisas que soportará el programa y tomará el valor que el programador decida. Se deberán implementar las siguientes funciones:

- **poner_divisas**: Función que debe solicitar por el teclado al usuario los datos necesarios de las divisas e ir guardándolos en el vector de divisas (que es un vector de estructuras). Por cada divisa será necesario su nombre (en plural) y el valor de 1 euro en dicha divisa. El cambio de cada divisa es un valor que deberá conocer el usuario, ya que los puede consultar en las páginas económicas de un periódico. Por ejemplo, para la peseta un 1 euro equivale a 166,386 ptas.
- **elegir_divisa**: Función que muestre las distintas divisas y permita elegir una por la posición que ocupan en el vector (de 0 en adelante). La divisa elegida será sobre la cual se realizarán el resto de las operaciones. Esta función devolverá un valor entero que servirá como índice para acceder al vector de divisas.
- **eur_a_divisa**: Función para convertir de euros a la divisa actual. Debe pedir una cantidad en euros y devolver su valor correspondiente en la divisa actual. A esta función, como a las 3 siguientes, será necesario que se les pase en la llamada el vector de divisas y el código de la divisa actual.
- **divisa_a_eur**: Función para convertir de la divisa actual a euros. Debe pedir una cantidad en la divisa actual y devolver su correspondiente en euros.
NOTA: La divisa actual se refiere a la divisa que el usuario haya elegido previamente mediante la función `elegir_divisa()`.
- **tabla_eur**: Función que visualiza por pantalla la tabla de conversión euros/divisa. Esta tabla consta de 2 columnas, la primera contiene cantidades en euros, y la segunda su conversión a la divisa actual. La función deberá pedir al usuario el incremento en euros, y mostrará una tabla con 20 líneas comenzando desde el valor asignado al incremento (ver ejemplo al final).
- **tabla_divisa**: Función que visualiza por pantalla la tabla de conversión divisa/euros. Esta tabla también consta de 2 columnas, pero al contrario que la de la función anterior, la primera columna mostrará las cantidades en la divisa actual, y la segunda en euros. También se visualizarán 20 líneas, comenzando por la cantidad *incremento* en la divisa actual e incrementando la cantidad en dicha divisa según el mismo *incremento*. Análogamente a la función anterior, ésta deberá pedir al usuario el incremento en la divisa actual.
- **menu**: Función que visualiza por pantalla un menú con las opciones disponibles, codificando cada una de ellas con un valor numérico. Ejemplo:

EURO-UTILIDADES

=====

- 1.- Introducir datos divisas
- 2.- Elegir divisa de trabajo
- 3.- Convertir euros a <divisa>
- 4.- Convertir <divisa> a euros
- 5.- Tabla de conversión de euros a <divisa>
- 6.- Tabla de conversión de <divisa> a euros
- 9.- Salir

Ya sea dentro de la función `menu()` o fuera de ella, una vez mostrado el menú se solicitará al usuario que elija una de las opciones y se leerá el valor que éste pulse.

La función principal o `main()` deberá visualizar el menú de opciones y realizar las operaciones pertinentes según la opción del menú que elija el usuario. Repetirá este proceso hasta que el usuario seleccione la opción de salida del programa. Además, en la función principal estará declarado el vector de divisas y una variable entera indicando la divisa actual. Tanto el vector como dicha variable entera deberán ser pasadas como argumentos a las funciones para que éstas sepan con qué información deben trabajar.

Ejemplo de lo que debe mostrar la función `tabla_eur()` si el incremento es de 3 euros, y la divisa actual la peseta.

EUROS	PESETAS
3.00	499.16
6.00	998.32
9.00	1497.47
12.00	1996.63
15.00	2495.79
18.00	2994.95
21.00	3494.11
24.00	3993.26
27.00	4492.42
30.00	4991.58
33.00	5490.74
36.00	5989.90
39.00	6489.05
42.00	6988.21
45.00	7487.37
48.00	7986.53
51.00	8485.69
54.00	8984.84
57.00	9484.00
60.00	9983.16