

Actividades propuestas

- 1) Escribir en lenguaje C un programa que: Pida por teclado dos números (datos enteros). Calcule la suma y multiplicación de los dos números introducidos. Muestre por pantalla los resultados (datos enteros).
- 2) Escribir en lenguaje C un programa que: 1º) Pida por teclado una letra (dato carácter) del abecedario. 2º) Muestre por pantalla: "ES UNA VOCAL", cuando la letra introducida sea una vocal minúscula o una vocal mayúscula. "NO ES UNA VOCAL", cuando la letra introducida no sea una vocal minúscula ni una vocal mayúscula.
- 3) Escribir en lenguaje C un programa que: Pida por teclado el número (dato entero) de un mes del año y muestre por pantalla el nombre (dato cadena) correspondiente a dicho mes. Nota: Si el número de mes introducido es menor que 1 o mayor que 12, se mostrará el mensaje: "ERROR: Mes incorrecto."
- 4) Escribir en lenguaje C un programa que: 1º) Pida por teclado la nota (dato real) de una asignatura. 2º) En el caso de que la nota sea incorrecta, muestre por pantalla el mensaje: "ERROR: Nota incorrecta, debe ser  $\geq 0$  y  $\leq 10$ ". 3º) Repita los pasos 1º y 2º, mientras que, la nota introducida sea incorrecta. 4º) Muestre por pantalla: "APROBADO", en el caso de que la nota sea mayor o igual que 5. "SUSPENDIDO", en el caso de que la nota sea menor que 5.  
Nota: Utilizar un bucle while.
- 5) Escribir en lenguaje C un programa que: 1º) Pida por teclado un número (dato entero). 2º) Pregunte al usuario si desea introducir otro número o no. 3º) Repita los pasos 1º y 2º, mientras que, el usuario no responda 'n' de (no). 4º) Muestre por pantalla cuántos números han sido introducidos por el usuario, así como, la suma de todos ellos.  
Nota: Utilizar un bucle do...while.
- 6) Escriba un programa que realice la lectura consecutiva, por teclado, de números enteros hasta que se introduzca el valor 0 (cero) o se hayan introducido 20 números, y los almacene en un vector, en orden inverso al de su introducción.
- 7) Rellene un array con los 100 primeros números pares y muestre su suma.
- 8) Lea 10 números por teclado, los almacene en un array y muestre la media.
- 9) Escriba un programa en C que pida al usuario los valores de una tabla de elementos enteros de dimensión 4 filas por 3 columnas, y muestre en pantalla la mayor de las sumas de sus columnas.
- 10) Escriba un programa que lea los valores de todos los elementos de dos vectores de enteros introducidos por el usuario, copie esos dos vectores en un tercero (uno a continuación del otro) y muestre sus valores por pantalla. Los tamaños de los dos vectores deben declararse como constantes.
- 11) Escriba un programa que declare un array llamado tabla, de dos dimensiones iguales, rellene todos sus elementos con el valor 1 y muestre dicho array. Luego el programa deberá poner a 0 todos los elementos de la diagonal principal y volver a mostrar el resultado.

- 12) Un negocio de paquetería con entregas a domicilio nos ha solicitado un programa para almacenar datos sobre las calles y las viviendas de su Ciudad. Cada calle se identificará por dos elementos: Nombre y viviendas que tiene. Además, cada vivienda estará identificada por el número que ocupa en la calle y la planta. Admitiendo que la ciudad tiene 5 calles y cada calle tiene cinco viviendas, desarrollar un programa que permita al usuario:
- a) Almacenar los datos sobre las calles y las viviendas de la ciudad.
  - b) Ver toda la información sobre una calle: número y viviendas que tiene.
- Nota: Por simplicidad se admitirá que los nombres de las calles no contienen espacios.
- 13) Escriba un programa que, tratando una fracción como una estructura de 2 componentes, permita realizar las operaciones suma, multiplicación, división, opuesto e inverso. El programa deberá preguntar la operación a realizar y después los datos para ejecutarla, y repetir estos pasos hasta que se seleccione entre las operaciones la opción terminar. Cada una de las operaciones se implementará como una función.
- 14) Escriba un programa en C que permita almacenar mediante el uso de estructuras los datos de 100 clientes de una empresa de automóviles. El programa permitirá ir almacenando los datos de los clientes, asignándole a cada uno un número según el orden en el que se van grabando. Los datos que se guardarán de cada cliente son el nombre, los apellidos, el teléfono y e-mail. Tras leer un cliente se dará la opción de terminar tecleando un 0.
- 15) Posteriormente modifique el programa para que evite que un cliente sea guardado dos veces. Para ello se comparará el teléfono con los ya guardados y en caso de que estuviera repetido se mostrará el nombre del cliente, el teléfono repetido y la posición en la que está grabado.

### **Material sugerido**

Introducción Debugger en CODEBLOCKS:

El Debug nos permite correr nuestro programa línea por línea observando, entre otras cosas cuánto valen nuestras variables a cada paso. Podríamos decir que es como mirar cada estado del programa.

Para iniciar el debug le damos al botón o en el menú Debug la opción Run. Para ver los valores de las variables ponemos en el menú de Debug y en la opción Debugging Windows marcar Watches. Esto nos abre una ventana donde podemos ver las variables y sus valores.

**Debug/Continue** Para comenzar/continuar el debug.

**Run to cursor** Corre el programa hasta donde está el cursor.

**Next line** Avanza una línea la ejecución. **Step into** En caso de estar en una línea que es una llamada a una función, con esta opción podemos hacer el seguimiento dentro de esta función (Ojo con las variables locales que la idea de "local" cambia).

**Step out** Termina la ejecución de la función actual y sigue en la siguiente línea desde donde fue llamado.

**Stop debugger** Termina el debug.

En los ejercicios de esta guía analice el uso del Debugger.