



Departamento de Lenguajes  
y Ciencias de la Computación  
UNIVERSIDAD DE MÁLAGA

# LABORATORIO DE PROGRAMACIÓN II

E.T.S.I.Informática. 1º C de GESTIÓN. 03-07-00

Duración del Examen: 3.5 horas

APELLIDOS \_\_\_\_\_ NOMBRE \_\_\_\_\_

ORDENADOR \_\_\_\_\_

La empresa de electrónica *Componentes Descompuestos S.A.* con motivo del lanzamiento de su nueva campaña *Ponga un Condensador en su Vida* ha decidido informatizar la gestión de sus clientes. Para ello nos solicita el desarrollo de una aplicación de gestión llamada **COMPO.MOD** que tenga las siguientes opciones de Menú:

Elaborado por: <Nombre> <Apellidos> <Curso>  
<Fecha> <Ordenador>

M E N U P R I N C I P A L  
=====

- A.- Introducir un cliente nuevo desde Teclado.
- B.- Mostrar la Información de un Cliente.
- C.- Mostrar por pantalla la Información de todos los clientes ordenada por su número e cliente.
- D.- Salvar los Datos en un fichero de Texto.
- E.- Cargar Datos desde un Fichero de Texto.
- F.- Borrar un Cliente.
- G.- Generar Etiquetas y Carta Personalizada.
- X.- Salir del Programa.

Introduzca Opción: \_

## Implementación:

Por cuestiones de eficiencia vamos a almacenar nuestros clientes en un árbol binario de búsqueda cuya **DEFINICIÓN** se adjunta y que **NO PODRÁ SER MODIFICADA**.

- **Introducir un cliente nuevo desde Teclado.**  
Se pedirá al usuario todos los datos del cliente y se insertará en el árbol.
- **Mostrar la información de un Cliente.**  
Se pedirá al usuario el número de cliente y se mostrará por pantalla toda su información. Si no existiera dicho cliente se informará de su no existencia.
- **Mostrar por pantalla la información de todos los clientes ordenada por el número de cliente.**  
Se mostrará por pantalla toda la información de todos los clientes ordenada por el número de cliente.
- **Salvar los datos en un fichero de Texto.**  
Se pedirá al usuario el nombre de un fichero y se grabará todo el contenido del árbol con el formato que se adjunta. Si existiera previamente un fichero con dicho nombre se borrará su contenido y si no existiera se creará.  
El formato del fichero será: 1 línea por cliente con la siguiente información:  
<numCli> <nom> <ap1> <ap2> <dir> <cp> <pob> <pro>

donde :

- <numCli> número de cliente (numérico)
- <nom> nombre del cliente (cadena de caracteres que no tiene espacios)
- <ap1> 1er Apellido del cliente (cadena de caracteres que no tiene espacios)
- <ap2> 2º Apellido del cliente (cadena de caracteres que no tiene espacios)
- <dir> Dirección del cliente (cadena de caracteres que no tiene espacios)
- <cp> Código Postal del cliente (numérico)
- <pob> Población del cliente (cadena de caracteres que no tiene espacios)
- <pro> Provincia del cliente (cadena de caracteres que no tiene espacios)

- **Cargar Datos desde un Fichero de Texto.**

Se pedirá al usuario el nombre de un fichero que tiene el mismo formato descrito para su escritura y se insertará su contenido en el árbol (el árbol puede o no estar vacío). Si el fichero no existiera no se inserta nada.

- **Borrar un Cliente.**

Se pedirá al usuario el número de cliente y se borrará toda su información del árbol. Si no existiera dicho cliente no se hará nada.

- **Generar Etiquetas y Carta Personalizada.**

Se solicitará al usuario el número de cliente así como el nombre de un fichero de texto donde se encuentra una carta modelo. Deberemos generar como salida dos ficheros de texto con la siguiente información:

\* ETIQUETA.XXX (donde XXX es el número de cliente) y que contendrá una etiqueta para el sobre con el siguiente formato:

```
<numCLi> <nom> <ap1> <ap2>
<dir>
<pob>
<cp> <pro>
```

\* CARTA.XXX (donde XXX es el número de cliente) y que contendrá la misma información que haya en el fichero que nos han pasado pero sustituyendo todas las apariciones de \$NOM por el nombre del cliente, \$AP por los dos apellidos del cliente y \$NUMC por el número de cliente.

- **Salir del Programa.**

Pedirá confirmación de salida, liberará todos los recursos que hayan sido reservados y saldrá al sistema.

Definición del Módulo que **NO PODRÁ SER MODIFICADA**:

```
DEFINITION MODULE MABB;
```

```
CONST
```

```
MAXCAD=80;
FINCAD=CHR(0);
```

```
TYPE
```

```
TCadena    = ARRAY [1..MAXCAD] OF CHAR;
TNumCli     = [0..999];
TElemento = RECORD
                numCli:TNumCli; (* campo clave del registro *)
                nom,ap1,ap2,dir,pob,pro:TCadena;
                cp:CARDINAL;
            END;
```

```
TABB;
```

```

PROCEDURE CreaABB():TABB;
(* Retorna un árbol binario de búsqueda vacío *)

PROCEDURE ABB_Vacio(a:TABB):BOOLEAN;
(* Nos dice si un árbol binario de búsqueda está vacío *)

PROCEDURE ABB_Lleno(a:TABB):BOOLEAN;
(* Nos dice si un árbol binario de búsqueda está lleno *)

PROCEDURE Info(a:TABB):TElemento;
(* Retorna la información de la raíz de un árbol binario de
búsqueda. Requiere para su uso que el árbol que se le pase como
parámetro NO esté vacío *)

PROCEDURE Izq(a:TABB):TABB;
(* Retorna el hijo izquierdo de un árbol binario de búsqueda no
vacío. En caso de que esté vacío devolverá el árbol vacío *)

PROCEDURE Dch(a:TABB):TABB;
(* Retorna el hijo derecho de un árbol binario de búsqueda no vacío.
En caso de que esté vacío devolverá el árbol vacío *)

PROCEDURE InsABB(VAR a:TABB;e:TElemento);
(* Inserta el elemento e en el árbol binario de búsqueda a. SI ya
existiera un nodo con la misma clave (número de cliente) se
entenderá que estamos actualizando los datos del cliente y se
copiará la información suministrada en dicho nodo sin generar un
nodo nuevo.*)

PROCEDURE EliminaABB(VAR a:TABB;clave: TNumCli);
(* Elimina del árbol el nodo cuyo número de cliente coincida con la
clave. Si no hay ningún nodo que cumpla dicha condición no se hará
nada *)

PROCEDURE BuscaABB(a:TABB; clave: TNumCli;
VAR existe:BOOLEAN;VAR e:TElemento);
(* Busca en el árbol el nodo cuyo número de cliente coincida con la
clave. Si lo encuentra se colocará la variable existe a TRUE y la
variable e contendrá una copia de los datos del cliente. Si no hay
ningún nodo que cumpla dicha condición se colocará la variable
existe a FALSE y el valor de e quedará indeterminado *)

PROCEDURE DestruirABB(VAR a:TABB);
(* Destruye todos los nodos del árbol *)
END MABB.

```

## Ejemplos de Ficheros:

### CLIENTES.TXT:

021 Pepe Pérez García AV.Principal 29091 Málaga Málaga  
201 María Sánchez Sánchez C.Pozo 29281 Torremolinos Málaga  
023 Antonio Ruiz Sánchez C.Cuesta 28432 Alcorcón Madrid

### CARTA.TXT:

A la Atención de D./ña \$NOM \$AP

Estimado \$NOM,

siendo usted uno de nuestros clientes más fieles, queremos comunicarle el lanzamiento de nuestra campaña "Ponga un Condensador en su Vida" por la que sólo por decir a la hora de hacer su pedido que su número de cliente es el \$NUMC por cada condensador que compre le regalaremos un resistencia.

Saludos,

EL Departamento Comercial.

### ETIQUETAS.021:

021 Pepe Pérez García  
AV.Principal  
Málaga  
29091 Málaga

### CARTA.021:

A la Atención de D./ña Pepe Pérez García

Estimado Pepe,

siendo usted uno de nuestros clientes más fieles, queremos comunicarle el lanzamiento de nuestra campaña "Ponga un Condensador en su Vida" por la que sólo por decir a la hora de hacer su pedido que su número de cliente es el 021 por cada condensador que compre le regalaremos un resistencia.

Saludos,

EL Departamento Comercial.

## Se valorará:

1. La corrección del programa.
2. Una buena modularización (uso de procedimientos/funciones)
3. Uso de tipos adecuados.
4. La ausencia de efectos laterales.
5. La legibilidad del código.
6. La presencia y claridad de mensajes de salida resultado de cada operación.

## Notas:

1. Se puede utilizar cualquier subrutina de los módulos estándares de TOPSPEED **EXCEPTO LA LIBRERÍA STR**
2. Es obligatorio trabajar en el directorio **C:\LP2\_1C**. Si no existe se creará. El programa principal se llamará COMPO.MOD
3. **Para Aprobar** deberá ser correcta la definición de tipos, la modularización y funcionar **CORRECTAMENTE** las **Opciones A,B, C, D y X** del menú.
4. Aquel **alumno que NO tenga superada la asistencia a prácticas** deberá realizar la **opción E** con obligatoriedad para aprobar.
5. Las opciones F y G del menú servirán para obtener más de aprobado.