



Introducción a la Programación

PRÁCTICA OBLIGATORIA GRUPAL

“Tienda Ordenadores URJC”

Una empresa llamada URJC (Universal Random Joint Commerce) tiene una tienda de venta de ordenadores. Los productos que vende la tienda son: Ordenadores completos (cuyos componentes hay que configurar) y componentes sueltos.

La información básica de un producto (ya sea un ordenador o un componente) es: identificador (se supone único), tipo, descripción y precio de venta.

El tipo de los componentes sueltos pueden ser: procesador, memoria o disco duro.

Además de la información anterior, un ordenador tiene un procesador, un módulo de memoria y un disco duro.

Se desea realizar un programa para poder gestionar la tienda. Las opciones que debe incluir el nuevo programa serán:

- A. **Dar alta un componente.** Esta opción permite dar de **alta** a un **nuevo componente suelto**. Para ello, si no está lleno el almacén de componentes, el sistema preguntará por todos los datos. El identificador del producto no puede estar repetido.
- B. **Configurar un ordenador.** Para configurar un ordenador (y darlo de alta), se deben solicitar los componentes que contendrá, que tienen que estar en el almacén de componentes sueltos. Una vez elegidos un procesador, una memoria y un disco duro, se solicitará el identificador, la descripción. El precio de venta se calculará como suma de todos sus componentes más 10 euros de mano de obra. Si todo es correcto (existe hueco en el almacén de ordenadores, se tienen todos los componentes y el identificador no está duplicado) se procederá a almacenarlo en el almacén de ordenadores y a eliminar los componentes sueltos del almacén de componentes.
- C. **Modificar un componente.** Para modificar cualquier campo de los datos de un componente se debe dar su identificador. El sistema mostrará los datos del componente y mediante un menú, preguntará el campo a modificar (se pueden modificar todos los datos excepto el identificador). El usuario introducirá el nuevo dato y volverá a mostrar los datos hasta que el usuario este conforme con la información. En ese momento se almacenarán los nuevos datos del componente en el almacén de componentes.
- D. **Vender un componente.** En esta opción se solicitará un identificador del componente a vender y si existe, se mostrarán los datos y se preguntará si está seguro de la venta. Si el usuario está de acuerdo, se procederá a la venta (y eliminación del almacén de componentes). Se debe actualizar el total de ventas.

- E. **Vender un ordenador.** En esta opción se solicitará un identificador del componente a vender y si existe, se mostrarán los datos y se preguntará si está seguro de la venta. Si el usuario está de acuerdo, se procederá a la venta (y eliminación del almacén de componentes). Se debe actualizar el total de ventas.
- F. **Mostrar las ventas actuales.** Se debe mostrar el valor del precio total de las ventas producidas
- G. **Mostrar todos los ordenadores, ordenados por precio de menor a mayor.** Al elegir esta opción, el programa deberá mostrar todos los ordenadores del almacén ordenados, mostrando primero el ordenador con menor precio.
- H. **Mostrar todos los componentes sueltos.** Con esta opción el programa deberá mostrar todos los componentes sueltos del almacén de componentes.
- I. **Guardar datos en ficheros binarios.** En esta opción se escribirá en dos ficheros binarios, denominados *componentes.dat* y *ordenadores.dat*, cada componente y cada ordenador de los almacenes respectivos. Se deben almacenar únicamente los productos que existen en los almacenes.
- J. **Guardar datos en ficheros de texto.** Esta opción permite almacenar la información de todos los componentes y de todos los ordenadores de los almacenes en dos ficheros de texto, denominado necesariamente *componentes.txt* y *ordenadores.txt*. En ambos ficheros, cada línea contendrá un dato de un producto. Así cada componente suelto se almacenará en el fichero *componentes.txt* en 4 líneas (una para el identificador, otra para el tipo, otra para la descripción y la última para el precio de venta). Cada ordenador se almacenará en 16 líneas (una para el identificador, otra para el tipo, otra para la descripción, una para el precio de venta, 4 para la memoria, 4 para el procesador y 4 para el disco duro). Se debe almacenar únicamente los datos de los productos que están en los respectivos almacenes.
- K. **Cargar datos de ficheros binarios.** Esta opción permite cargar los productos (tanto los componentes sueltos como los de los ordenadores) de los ficheros binarios. Los productos pre-existentes en el sistema se eliminarán y el total de ventas se pondrá a cero. Si el fichero no existe, se mostrará un mensaje por pantalla.
- L. **Cargar datos de ficheros de texto.** Esta opción permite cargar los productos (tanto los componentes sueltos como los de los ordenadores) de los ficheros de texto. Los productos pre-existentes en el sistema se eliminarán y el total de ventas se pondrá a cero. Si el fichero no existe, se mostrará un mensaje por pantalla.

Es obligatorio utilizar estos tipos de datos en el programa:

CONST

```
NCTIPO = 15; {número caracteres tipo}
```

```

NCIDENTIFICADOR = 4; {número caracteres del identificador del componente}
MAXPC = 25; {número de ordenadores (PC) máximos en la tienda}
MAXCOMPONENTES = 100; {número de componentes sueltos máximo}
MIN = 1;

```

TYPE

```

tTipo = string[NCTIPO]; {Tipo para almacenar el tipo del componente}
tIdentificador = string[NCIDENTIFICADOR]; {Para almacenar el identificador}
tNumComponentes = MIN..MAXCOMPONENTES; {Para almacenar el índice de componentes}
tNumPc = MIN..MAXPC; {Tipo para almacenar el índice de ordenadores}

tComponente = RECORD {Tipo para almacenar un producto}
  tipo: tTipo;
  id: tIdentificador;
  descripcion: String;
  precio : real;
END;

tPc = RECORD {Tipo para almacenar un ordenador}
  datos, memoria, procesador, discoDuro: tComponente;
END;

tListaComponentes= ARRAY [tNumComponentes] OF tComponente;

tListaPcs = ARRAY [tNumPc] OF tPc;

tAlmacenComponentes = RECORD {Almacén de componentes}
  listaComponentes : tListaComponentes;
  tope: integer;
END;

tAlmacenPcs = RECORD {Almacén de Pcs}
  listaPcs : tListaPcs;
  tope: integer;
END;

tTienda = RECORD {Tienda}
  almacenPcs : tAlmacenPcs;
  almacenComponentes: tAlmacenComponentes;
  ventasTotales: real; {Almacena el total de la ventas}
END;

tFicheroPcs = FILE OF tPc;
tFicheroComponentes = FILE OF tComponente;

```

Normas de entrega.

La práctica se entregará siguiendo la siguiente normativa, en caso de no cumplir cualquiera de los puntos aquí expresados la práctica se calificará como **NO APTA**.

- La práctica se tendrá que realizar en grupos de tres o cuatro personas, no se admitirán entregas por grupos de alumnos de número superior o inferior al indicado (a no ser que haya una aceptación expresa por parte del profesor) quedando, si se diese el caso, calificados todos los implicados como NO APTOS.
- Los alumnos deberán aportar, además del código fuente (tienda.pas), dos archivos binarios *-componentes.dat* y *ordenadores.dat*- que contengan información sobre al

menos 5 componentes sueltos y 3 ordenadores, que permitan realizar pruebas **y una memoria explicativa**. Esta memoria contendrá:

- los nombres de los alumnos que entregan la práctica, la titulación de cada uno,
 - las cabeceras de los subprogramas con un breve comentario de su utilidad y el nombre del programador de ese subprograma.
 - En este archivo se podrán introducir las observaciones que consideren convenientes sobre lo entregado.
-
- La entrega, se realizará a través del **aula virtual** en un **único archivo** comprimido con Winzip o Winrar, que llamaremos **tienda.zip**.
 - El programa deberá compilar en Eclipse Gavab. La ausencia del archivo fuente (tienda.pas), de los ficheros binarios o si tiene problemas de compilación supondrá que la práctica no sea corregida y por lo tanto calificada como NO APTA.
 - La utilización de variables globales dentro de los subprogramas conllevará la evaluación con un 0 para todos los componentes del grupo.
 - Fecha límite de entrega el **11 de enero de 2016 a las 23:00**. Bajo ninguna circunstancia **se recogerán prácticas pasado ese día/hora** o mandadas por otro medio de comunicación.
 - El archivo **tienda.zip** será enviado a través del campus virtual **por un único integrante del grupo, el portavoz**. Este archivo se compondrá del fichero **tienda.pas** y los dos **ficheros binarios** que contengan los datos de pruebas. También **se debe incluir en el zip/rar la memoria**.

NOTA IMPORTANTE 1: El uso de variables globales en los subprogramas supondrá un suspenso en la práctica.

NOTA IMPORTANTE 2: La copia de la práctica será motivo de suspenso en TODA la asignatura.

NOTA IMPORTANTE 3: No se permitirá el envío de la práctica por ningún otro medio que no sea a través de la sección habilitada a tal efecto en el Aula Virtual.

FECHA DE ENTREGA:

El último día de entrega es el **11 de enero de 2016 a las 23:00**. A partir de ese momento no está permitido ningún envío.