

Facultad de Informática – Universidad Complutense
1º curso de los Grados
Fundamentos de la programación – Grupos D, E y D.G.
Curso 2015–2016

Examen de febrero de 2016

Tiempo disponible: 3 horas

Se realizará una aplicación para clasificar el consumo de los clientes de una compañía de gas en función de su consumo. Se quiere saber el número de clientes por intervalos de consumo en euros (por ejemplo, cuántos tienen un consumo entre 30 euros y 100).

La información sobre cada cliente consta del identificador del contrato (cadena de caracteres), el consumo en euros (nº real) y el tipo de cliente (enumerado: particular, PYME, industria).

Los intervalos de consumo vienen definidos por unos enteros positivos crecientes. Por ejemplo, 30, 100, 500, 1000, 2000, denotan estos intervalos: consumo de 0 a 30, (30,100], (100, 500], (500, 1000] y (1000, 2000]. Por ejemplo, (30,100] es el intervalo de valores de consumo mayores que 30 y menores o iguales que 100 (no incluye el 30, sí el 100).

La aplicación manejará una tabla de frecuencias de exactamente NUM_INTERVALOS, siendo NUM_INTERVALOS > 0. Ejemplo con NUM_INTERVALOS = 5, la tabla podría ser:

30	100	500	1000	2000
80	100	99	50	3

Esta tabla nos indicaría que hay 80 clientes con un consumo mayor que 0 y menor o igual que 30 euros, 100 clientes con un consumo mayor que 30 y menor o igual que 100, ..., y 3 clientes con un consumo mayor que 1000 y menor o igual que 2000.

La información de los clientes se guarda en un archivo de texto `consumos.txt` organizado de forma que los datos de cada cliente aparecen en una línea separados por un espacio: el consumo (nº real), el tipo de cliente (entero asociado al enumerado) y el identificador del contrato (cadena de caracteres con posibles espacios). El archivo termina con una línea correspondiente al cliente centinela: consumo -1. **Ejemplo** de parte del archivo:

```
45 0 234AC 23Z    // consumo 45, tipo particular, identificador de contrato
634 1 45A32 12S   // consumo 634, tipo PYME, identificador de contrato
....
-1 0 XXX          // centinela: consumo -1, tipo, identificador de contrato
```

(El fichero `consumos.txt` no incluirá los textos “// ...” mostrados arriba, los mostramos aquí sólo para aclarar lo que significa cada renglón del fichero.)

La información sobre los intervalos de consumo a utilizar se encuentra guardada en un archivo `intervalos.txt`, que contiene enteros y termina con `-1`. Se espera que el número de enteros sea `NUM_INTERVALOS` y que estén ordenados de menor a mayor (estrictamente creciente). En caso de no ser así el programa dará un mensaje y parará. Por ejemplo, si `NUM_INTERVALOS` es 5, el fichero podría contener:

```
30 100 500 1000 2000 -1
```

Así que `intervalos.txt` sólo indica los intervalos a utilizar, no el número de clientes dentro de cada uno.

Declara las constantes y tipos apropiados para representar la información descrita.

Programa principal (Declaración de constantes, tipos y programa principal [1 pto.])

El programa comienza cargando la información del archivo `intervalos.txt` en la tabla de frecuencias. Si la carga tiene éxito, se procede a contar cuántos clientes del archivo `consumos.txt` y **del tipo PYME** hay en cada intervalo. Se termina visualizando la tabla de frecuencias de consumo de clientes de tipo PYME.

Implementa, al menos, los siguientes subprogramas:

- ✓ [3 ptos.] `cargarIntervalos()`: Genera la tabla de frecuencias con los datos del archivo `intervalos.txt`, inicializando todas las frecuencias a 0, y devuelve un booleano que indica si la carga ha tenido éxito o no. Durante el proceso, se comprobará que haya exactamente `NUM_INTERVALOS` enteros en orden creciente (sin contar el centinela).
- ✓ [2,5 ptos.] `buscarIntervalo()`: Dada la tabla de frecuencias y un **cliente**, devuelve la posición que corresponde al intervalo de consumo de ese cliente, y si existe o no.
- ✓ [2,5 ptos.] `frecuenciasConsumo()`: Dada la tabla de frecuencias ya inicializada con los números de los intervalos, y el tipo de cliente a clasificar (particular, PYME o industria), escribe en la tabla las frecuencias de los clientes de ese tipo presentes en el archivo `consumos.txt`. Los clientes cuyo consumo no se corresponda con ningún intervalo de la tabla se ignoran.
- ✓ [1 pto.] `mostrarTabla()`: Dada una tabla de frecuencias de consumo, muestra en la pantalla los intervalos de consumo y el número de clientes en cada uno. Cada elemento de la tabla se mostrará en una línea según el ejemplo:

Tabla de frecuencias por consumo:

De 0 a 30:	80
De 30 a 100:	100
De 100 a 500:	99
De 500 a 1000:	50
De 1000 a 2000:	3

Ejemplo de ejecución:

Si el fichero consumos.txt contiene:

```
45 0 234AC 23Z
634.3 1 45A32 125
1000 1 55A31 1F
2542.2 2 57F30 45S
2500 2 55Z21 23A
100.1 1 21Q23 33W
-1 0 XXX
```

Y el fichero intervalos.txt contiene:

```
30 100 500 1000 2000 -1
```

Entonces el programa mostrará:

```
Calculamos las frecuencias de consumo de los clientes de tipo PYME.
Tabla de frecuencias por consumo:
De 0 a 30: 0
De 30 a 100: 0
De 100 a 500: 1
De 500 a 1000: 2
De 1000 a 2000: 0
```

Y terminará.

Instrucciones de entrega

- Pon al inicio de tu archivo de código .cpp un comentario con tus datos:
Nombre completo y DNI
Laboratorio y puesto
- Cuando termines el ejercicio sube el archivo con el código fuente (ninguno más) por el sistema de envío de ftp.
- Se valorará la estructura, legibilidad y uso de los esquemas de recorrido y búsqueda vistos en clase. Recuerda que no se permite el uso de variables globales, ni de instrucciones de salto, salvo un return como última instrucción de las funciones y break en los casos del switch.