

## Programación 2

### Examen de teoría (2021 - C4)

8 de julio de 2021



### Ejercicio 1 (5 puntos)

#### Instrucciones

- Duración: **1 hora y 25 minutos**. La hora tope para la entrega de este primer ejercicio será las **13:25**
- Después de esto, habrá un **descanso de 5 minutos** y a continuación se podrá descargar el enunciado del siguiente ejercicio
- **No se puede salir del aula durante el descanso**
- En este primer ejercicio debes entregar un único fichero llamado **ej1.cc**, que contendrá el código fuente del ejercicio. **Pon tu DNI y tu nombre en un comentario al principio del fichero**
- La entrega se realizará como en las prácticas, a través del servidor del DLSI (<http://pracdlsi.dlsi.ua.es>), en el enlace **Programación 2 - Ingeniería Informática**. Puedes realizar varias entregas, aunque sólo se corregirá la última
- Está permitido durante el examen consultar los materiales de la asignatura, libros, apuntes y/o referencias en Internet. Si el código **no compila**, se restará un punto de la nota final del ejercicio
- Para la resolución de los ejercicios, **solo está permitido el uso de las librerías y funciones vistas en clase de teoría y prácticas**. No es válido aportar soluciones copiadas y pegadas de Internet

#### Código de conducta

- El examen es un trabajo individual. **Cualquier indicio de copia, comunicación con otros alumnos (p.ej. mediante grupos de WhatsApp) o intervención de terceras personas en su realización será sancionada según la legislación vigente**, con medidas que pueden llevar a la **expulsión** del alumno/a de la titulación

#### Enunciado

La empresa *Tefío S.A.* se dedica a conceder micropréstamos rápidos a sus clientes. Por cada cliente se dispone de la siguiente información:

- Código de cliente
- Nombre y apellidos
- Importe total de préstamos concedidos hasta el momento
- Clasificación del cliente: T si es de riesgo o F en caso contrario

Deberás crear un registro (**struct**) para almacenar la información de cada cliente. Debes decidir tú los nombres y tipos de datos que tendrán los campos del **struct**, teniendo en cuenta estas consideraciones:

- El código de cliente será siempre un valor entero positivo
- El nombre y apellidos se almacenarán juntos en un único campo

La información de los clientes está guardada en un fichero de texto llamado **endeudados.txt**. Este fichero tiene las siguientes características:

- El número de clientes almacenado no es fijo (lo mismo pueden haber 10 que 10.000)
- Para cada cliente, aparecerá cada uno de sus datos en una línea, uno a continuación de otro, siguiendo el orden que se ha mencionado arriba (primero código cliente, luego nombre, etc.)
- La información de todos los clientes es completa y correcta, es decir, siempre habrá cuatro líneas por cada cliente con la información correspondiente en el orden indicado

Un ejemplo de fichero sería el siguiente:

```
1234
Paco Avutarda
750
F
2345
Emilio Pandorgo
4900.89
T
3456
Teresa Gobanilla
3000.5
T
```

Al iniciar el programa se deberá cargar en memoria el contenido del fichero `endeudados.txt` o el contenido del fichero que se indique mediante argumentos por línea de comando, tal y como se describe más abajo. Si en cualquiera de los dos casos el fichero no se puede abrir, se mostrará un error por pantalla y terminará el programa sin llegar a mostrar el menú de opciones.

Tras cargar esta información, se mostrará un menú por pantalla con tres opciones:

- La primera opción permitirá ampliar el préstamo de un cliente. Para ello se pedirá el código del cliente y, una vez introducido por el usuario, se deberá verificar que el cliente existe antes de continuar. Si no existe se mostrará un mensaje de error y se volverá al menú principal. Si el cliente existe y no es de riesgo (su clasificación como cliente es F) se pedirá al usuario que introduzca una cantidad y se ampliará el importe total que tiene concedido sumándole el importe introducido, sólo si el importe resultante no excede los 5.000€. Además, se debe comprobar que el importe introducido sea mayor que 0. Esta información se actualizará en memoria y se volverá al menú principal. Si el préstamo no se ha ampliado por alguno de los motivos anteriores (es cliente de riesgo, el importe es menor o igual a cero, o el resultado supera los 5.000€) se mostrará el un mensaje de error y se volverá al menú principal sin modificar los datos del cliente.
- La segunda opción mostrará todos los datos de los clientes de riesgo, indicando al final de la lista el importe total de todos los préstamos concedidos a estos clientes (es decir, la suma de todos estos préstamos).
- La opción tercera finalizará el programa, pero antes deberá sobrescribir el fichero de texto `endeudados.txt`, o el fichero que se hubiera pasado como argumento al programa, con la información de todos los clientes (incluyendo las modificaciones que se hayan podido hacer durante la ejecución del programa).

Si el usuario introduce una opción incorrecta en el menú, se mostrará un mensaje de error y se volverá a mostrar el menú.

El programa admite argumentos por línea de comando. Estos argumentos son:

- `-i <nombre fichero>`: iniciará el programa cargando en memoria los datos del fichero de texto pasado como argumento y mostrando el menú principal a continuación.
- `-s <nombre fichero>`: al igual que el anterior, iniciará el programa cargando en memoria los datos del fichero de texto pasado como argumento, pero además mostrará a continuación por pantalla toda la lista de clientes que se han leído (pero no se mostrará la cantidad total de los préstamos). Finalmente se mostrará el menú principal.

Solo se puede dar uno de estos argumentos al mismo tiempo. Si los argumentos son incorrectos, se deberá mostrar un error por pantalla y finalizar el programa sin llegar a mostrar el menú inicial. Ejemplo de llamadas válidas al programa:

```
$ ej1  
$ ej1 -i morosos.txt  
$ ej1 -s deudores.txt
```

Introduce los comentarios que creas necesarios en el código para que el profesor que te corrija entienda la información que estás almacenando y, en general, para aclarar cualquier función que desarrolles.