

**Fundamentos de la programación II**

Curso 2019-20

**Convocatoria ordinaria**

Tiempo disponible: 1 hora

(6,5 puntos) Una clínica quiere desarrollar una aplicación para gestionar la asignación de citas médicas entre los pacientes y los médicos. Para representar la información de cada médico se define un tipo **tMedico** que se compone de:

- ✓ Número de colegiado (entero que describe de manera unívoca a un médico).
- ✓ El número de citas libres o disponibles.
- ✓ El importe o tarifa de la consulta de dicho médico.

La aplicación debe disponer de una lista de médicos. Para ello se define un tipo **tListaMedicos** que representa una lista de hasta 100 punteros a variables de tipo **tMedico**. El catálogo estará ordenado de manera decreciente por el número de colegiado. A modo de ejemplo, aquí tenemos un catálogo de 7 médicos:

9999	8888	7777	6666	5555	4444	3333
10	12	10	8	1	15	20
35€	70€	40€	50€	80€	35€	60€

También disponemos de una lista de las citas solicitadas por los pacientes. Definimos un tipo **tListaCitas** que representa la lista de citas solicitadas por los pacientes: una lista implementada con un array dinámico, donde cada elemento es del tipo **tCita** compuesto por el código del médico y el código del paciente que solicita la cita (cadena de 4 caracteres). La lista de citas no tiene ningún orden establecido.

A modo de ejemplo, aquí tenemos el contenido de una lista de 4 peticiones o solicitudes de citas. En la primera solicitud se interpreta que el paciente K122 solicita una cita con el médico 7777:

7777	7654	8888	5555
K122	U345	M355	L244

En estos momentos tenemos definidos los tipos, hemos implementado un subprograma para cargar desde archivo los datos de los médicos, otro para cargar las peticiones de citas, y sendos subprogramas para mostrar la lista de médicos por pantalla. También tenemos un programa principal de prueba que deberá acabar haciendo lo siguiente: (a) cargar la lista de médicos y la lista de peticiones de citas, (b) visualizarlos por pantalla, (c) llamar a un subprograma **procesarPeticiones** que permite gestionar las peticiones de citas: (d) visualizar por pantalla el coste de las citas solicitadas y actualizar la información de los médicos tras las citas y (e) liberar la memoria.

Tu misión es completar el primer prototipo de esta aplicación:

1. Desarrolla el subprograma **procesarPeticiones**, que recibe una lista de médicos y una lista de peticiones de citas y, con las citas que son posibles, calcula el coste total de las citas y actualiza las citas disponibles en la lista de médicos . Ten en cuenta que
  - La lista de peticiones de citas puede incluir médicos que no existan en la lista de médicos, no pudiendo contabilizarse.
  - El procesamiento de las solicitudes de citas debe actualizar convenientemente el número de citas disponible de cada médico en la lista de médicos, teniendo en cuenta que si un médico se queda sin citas , éste desaparecerá de la lista.

Tomando como ejemplo la lista de médicos y la lista de citas anteriores, el coste total a mostrar sería 190€ (resultado de 40 + 70 + 80). Se mostrará esto:

**El importe total es: 190 euros**

Ojo!: fíjate que no existe el médico 7654 en la lista y dicha lista quedaría actualizada así (puedes ver que ha desaparecido el médico 5555, pues se ha quedado sin citas):

9999	8888	7777	6666	4444	3333
10	11	9	8	15	20
35€	70€	40€	50€	35€	60€

2. Para finalizar, se debe liberar toda la memoria que se ha reservado. Para ello, añade las funciones que consideres necesarias y llámalas desde el programa principal.

A continuación se muestra el ejemplo de ejecución con los datos anteriores:

```
***** LISTA DE MEDICOS *****

=====
9999 10 35
8888 12 70
7777 10 40
6666 80 50
5555 1 80
4444 50 35
3333 20 60
=====

***** LISTA DE PETICIONES DE CITAS PENDIENTES DE PROCESAR *****

7777 - K122
7654 - U345
8888 - M355
5555 - L244

El importe total es :190 euros

***** LISTA DE MEDICOS *****

=====
9999 10 35
8888 11 70
7777 9 40
6666 80 50
4444 50 35
3333 20 60
=====

Presione una tecla para continuar . . .
```

Debes enviar por el campus todos los archivos modificados (o un archivo comprimido con todos ellos)

