

## Examen parcial – 30 de enero de 2013

Tiempo disponible: 2,5 horas – Entrega: febrero.cpp

Debes construir un programa (**febrero.cpp**) que muestre repetidas veces el siguiente menú de opciones y realice la opción seleccionada hasta que el usuario escoja la opción 0 para terminar:

- 1 - Generar una secuencia
- 2 - Procesar la secuencia
- 3 - Buscar en la secuencia
- 0 - Salir

Opción 1: 2 puntos    Opción 2: 3 puntos  
Opción 3: 2 puntos    General: 3 puntos

El programa informará al usuario si su opción no es válida, volviendo a leer opción.

1. Si elige la opción 1, se le pedirá que indique un valor máximo para los números de la secuencia y a continuación se llamará a un procedimiento **generar()**, al que se le pasará ese valor máximo y que generará una secuencia de números enteros en el archivo **datos.txt** (que terminará con un **-1** como centinela). Se le pedirán números al usuario hasta que introduzca un **0**, escribiendo en el archivo **sólo** los valores que estén entre 1 y el valor máximo indicado por el usuario. Cada número irá en una línea del archivo.
2. Esta opción sólo se ejecutará si previamente se ha generado ya una secuencia. Si el usuario elige la opción 2, se llamará a un procedimiento **recorrer()** que realizará un recorrido de la secuencia del archivo **datos.txt**, indicando, para cada uno, si se trata de un *número feliz* o no. Para saber si un número (por ej., **49**) es feliz, sumamos los cuadrados de sus dígitos ( $4^2 + 9^2 = 16 + 81 = 97$ ) y volvemos a sumar los cuadrados de los dígitos del resultado hasta llegar a un solo dígito ( $9^2 + 7^2 = 81 + 49 = 130$ ;  $1^2 + 3^2 + 0^2 = 1 + 9 + 0 = 10$ ;  $1^2 + 0^2 = 1 + 0 = 1$ ). Si el resultado final es 1, el número es feliz.

Se usará una función **esFeliz()** que aceptará un entero y devolverá **true** o **false** indicando si ese número es o no feliz.

3. Esta opción sólo se ejecutará si previamente se ha generado ya una secuencia. Si el usuario elige la opción 3, se buscará en la secuencia del archivo **datos.txt** el primer número que tenga una **diferencia con el anterior** en la secuencia menor o igual que un determinado error que será proporcionado por el usuario.

Esta opción hará uso de un procedimiento **buscar()** que recibirá el error admisible proporcionado por el usuario y devolverá una variable booleana que indique si se ha encontrado un número que cumpla tal condición, así como (si se ha encontrado) la diferencia real del número con el anterior y el número de línea en que se encuentra ese número en el archivo. Véase el ejemplo de ejecución.

Estructura adecuadamente el código del programa, creando los subprogramas oportunos y estableciendo los canales de comunicación necesarios entre ellos. Se valorará la estructura, la legibilidad y el adecuado uso de los esquemas de recorrido y búsqueda de secuencias vistos en clase.

Cuida el estilo y depura a medida que vayas construyendo el programa (no esperes a terminarlo todo; prueba cada parte que vayas completando). No olvides poner al principio tu nombre y DNI. Deberás entregar (sólo) el archivo **febrero.cpp** con el código del programa a través del Campus Virtual (*Tareas*).

Ejemplo de ejecución (en negrita las entradas del usuario):

```
1 - Generar la secuencia
2 - Procesar la secuencia
3 - Buscar en la secuencia
0 - Salir
Opción: 2
Ninguna secuencia generada.
```

```
1 - Generar la secuencia
2 - Procesar la secuencia
3 - Buscar en la secuencia
0 - Salir
Opción: 7
Opción no válida!
```

```
1 - Generar la secuencia
2 - Procesar la secuencia
3 - Buscar en la secuencia
0 - Salir
Opción: 1
Límite para los valores: 1000
Entero positivo (0 termina): -2
Entero positivo (0 termina): 1200
Entero positivo (0 termina): 7
Entero positivo (0 termina): 97
Entero positivo (0 termina): 137
Entero positivo (0 termina): 250
Entero positivo (0 termina): 0
```

```
1 - Generar la secuencia
2 - Procesar la secuencia
3 - Buscar en la secuencia
4 - Dibujo
0 - Salir
Opción: 2
7 sí es feliz
97 sí es feliz
137 no es feliz
250 no es feliz
```

```
1 - Generar la secuencia
2 - Procesar la secuencia
3 - Buscar en la secuencia
0 - Salir
Opción: 3
Límite de diferencia con anterior: 50
Encontrado en línea 3 (diferencia: 40)
```

```
1 - Generar la secuencia
2 - Procesar la secuencia
3 - Buscar en la secuencia
0 - Salir
Opción: 0
```