

Examen de junio– 18 de junio de 2012

Tiempo disponible: 3 horas

Una conocida empresa nos ha encargado un programa que realice el control horario de su personal, de tal forma que sepan el tiempo que ha trabajado cada empleado, sus entradas y salidas, etcétera. Queremos mantener una lista con la información de los empleados y de cada empleado tenemos inicialmente la siguiente información:

- Id de empleado (entero)
- Nombre completo (nombre y apellidos)
- Departamento al que pertenece
- Lista de los Id de los empleados que trabajan en el mismo departamento. Se implementará como un array dinámico.
- Tiempo trabajado en la jornada actual (horas y minutos, por separado)

Para controlar las entradas y salidas de los empleados se han instalado en todas las puertas de la empresa unos controles que recogen información sobre las entradas y salidas de los empleados; la información se almacena en el archivo de texto `accesos.txt` que nuestro sistema deberá procesar.

El archivo corresponde al día actual de forma que nuestro programa se ejecutará al final del día. En cada una de las líneas del archivo viene un movimiento en el que aparece el Id del empleado que lo realizó, si es una entrada o una salida ('E' o 'S'), y la hora exacta del movimiento (en formato 24 horas, HH MM). Los datos vienen separados por un espacio en blanco. Y el archivo termina con una última línea con un `0` como centinela. Ejemplo de archivo `accesos.txt`:

```
123 E 12 34
345 E 12 35
123 S 20 34
245 E 08 00
345 S 13 00
245 S 13 05
0
```

Se pide construir un programa en C++ (**junio.cpp**) que muestre repetidas veces el menú de opciones que se muestra en la siguiente página y ejecute la opción seleccionada hasta que el usuario escoja la `0`.

- 1 - Alta de empleado
- 2 - Procesar accesos
- 3 - Eliminar dpto.
- 4 - Listado
- 0 - Salir

- Declaraciones y estructuras de datos: 1 punto
- main() y menús: 0,5 puntos
- Opciones 1, 2 y 3: 2 puntos cada una
- Opción 4: 1,75 puntos
- Copia de seguridad: 0,75 puntos

El programa deberá encargarse de validar la opción introducida por el usuario, de forma que reiteradamente solicite una opción mientras que la introducida no sea válida. Una vez que el usuario haya elegido una opción válida, si es la 0 terminará la ejecución del programa, y si es alguna de las otras realizará el correspondiente procesamiento, atendiendo a las siguientes indicaciones:

1. Si el usuario elige la **opción 1**, se leerán del teclado los datos de un empleado: id, nombre completo, departamento, cuántos otros empleados trabajan en el mismo departamento y los id de esos otros empleados (en la misma línea y separados por espacios). Se dará de alta el empleado en la lista si el Id de empleado no existe todavía.
La lista se mantendrá ordenada por Id de empleado.
2. Si el usuario elige la **opción 2**, se procesará el archivo de movimientos `accesos.txt`, actualizando las horas trabajadas por cada empleado. (Pista: resta las horas, resta los minutos; si la resta de minutos sale negativa, suma 60 al total de minutos y resta 1 al total de las horas).
3. Si el usuario elige la **opción 3**, se pedirá un departamento y se eliminarán todos los empleados de dicho departamento.
4. Si el usuario elige la **opción 4**, se mostrará un listado de los empleados. Por cada empleado: Id, tiempo trabajado, nombre completo y los Id de los compañeros de departamento que han trabajado menos tiempo que el empleado en cuestión.

Ejemplo de listado:

```
123 08:00 Pedro López (245, 345)
245 05:05 Luis Ramos (245)
345 00:25 Juan Rivas ()
...
```

Al salir del menú con la opción 0, el programa hará una copia de seguridad del día en un archivo `backup.txt` (para cada uno de los empleados se grabará su Id de empleado y el tiempo trabajado ese día).

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 un comentario al principio con tu nombre. Deberás entregar el archivo `junio.cpp` con el código del programa a través del Campus Virtual.