

Principios de Programación

Obligatorio 2

Para lograr la aprobación del curso además de llegar a un puntaje mínimo de 25% en los parciales el estudiante deberá realizar dos tareas obligatorias cuya no aprobación de alguna de ellas implicara la pérdida del curso.

El estudiante deberá resolver e implementar las tareas propuestas utilizando el lenguaje de Programación C, aplicando todos los conocimientos impartidos en el curso.

Para completar la parte obligatoria del curso, se pide una nueva entrega, en esta se debe extender el *TA-TE-TI* de la primer entrega.

En esta nueva versión se jugara un número determinado de veces definido por una constante por ejemplo:

```
#define MAX_JUEGOS 10
```

Por cada vez se recordará el nombre del ganador, y la cantidad de fichas que puso en el tablero. En el caso de empate se anotara “*empate*” en el nombre, y “-1” en la cantidad de fichas.

Una vez terminado el juego se deberá imprimir la estructura de datos almacenada, en una primera instancia ordenada alfabéticamente por nombre, y luego ordenada por cantidad de fichas.

Se deberá implementar una estructura que permita almacenar estos datos.

La salida será idéntica a la de la primer entrega, salvo que en el momento de anunciar quien gano, se preguntara el nombre al ganador, y se almacenara el nombre y la cantidad de fichas en la estructura.

En caso de salir del juego antes del final con el comando “*salir*”, se deberá imprimir la tabla igual que en el caso que el juego culmina normalmente, pero solo ordenando e imprimiendo hasta el lugar en que se anoto el último juego.

Ejemplos:

Ejemplo de cómo pedir al usuario el nombre:

```
***Partido en juego***
*Jugador 1 X
*Jugador 2 0

_X_|_O_|_X_

_O_|_X_|_O_

_X_|_O_|_X_

GANO Jugador 1!!!!
Nombre: Virginia
```

Ejemplo de salida, impresión de la tabla para MAX_JUEGOS 10:

(Alfabéticamente)

Fichas	Nombre
Empate	-1
Empate	-1
Juan	3
Lucas	4
Mauricio	3
Miguel	5
Rodrigo	3
Rosario	4
Victoria	3
Virginia	4

(Cantidad de Fichas)

Fichas	Nombre
Miguel	5
Virginia	4
Lucas	4
Rosario	4
Victoria	3
Juan	3
Rodrigo	3
Mauricio	3
Empate	-1
Empate	-1

Otros

El estudiante para realizar la tarea, deberá estudiar el uso de las estructuras de control *for*, *while*, *do while*, *if then else*, *case*, etc. Deberá definir tipos de datos para almacenar la información requerida, deberá también definir procedimientos y funciones.

Se deberá hacer uso de la biblioteca `string.h` para manejo de cadenas de caracteres.

Esta terminantemente prohibido y se castigará con dureza el uso de sentencias que no respeten la programación estructurada, como break goto, etc, y el uso de variables globales.

Compilador

Todos los módulos entregados deben compilar y linkeditar con el compilador del curso mingw32 3.4.2. Versión que viene con el entorno de desarrollo:

Dev-C++ 5.0 beta 9.2 (4.9.9.2) (9.0 MB) with Mingw/GCC 3.4.2

Que se puede obtener del sitio : <http://bloodshed.net/dev/devcpp.html>

De no ser así el trabajo será eliminado, con la consiguiente pérdida del curso.

Formas y Plazos de Entrega

El plazo de entrega será hasta el domingo 21 de junio del 2009 a las 23:30 horas.

Se deberá entregar un archivo con nombre ***tateti2.cpp*** con el código de programa Cuya primera línea debe contener la cedula del integrante y en la segunda su nombre completo.

Los estudiantes podrán discutir la resolución de la tarea entre ellos, pero de ninguna manera se permitirá compartir código ni copiar otros trabajos, la tarea será realizada de forma individual, de no ser así, o no se entrega o la entrega es insuficiente, implicará la pérdida del curso. La copia será castigada severamente.

La forma de entrega será comunicada por el docente del práctico.