

Universidad Tecnológica Centroamericana  
UNITEC

13 de septiembre del 2019

**Sesión #9**

Laboratorio de Programación III

Javier Cano

# Desarrollo de la práctica

Para el desarrollo satisfactorio del laboratorio, siga al pie de la letra cada instrucción que a continuación se le presenta.

# Objetivos del laboratorio

* Hacer uso de hilos en C++

# Ejercicio practico

Se le pide realizar un pequeño prototipo sobre un juego 1 contra 1 en línea. Las mecánicas específicas del juego por los momentos no son de gran importancia, pero si se desea que diseñe una arquitectura viable para el programa. La empresa para la que trabaja tiene algunas limitantes de presupuesto, entonces tendrá que simular los encuentros en línea por medio de hilos de control. El juego consistirá, inicialmente, sobre un simple encuentro de piedra, papel o tijera.

Se debe saber la siguiente información del jugador controlado por el usuario:

* Dinero (inicialmente 0)
* Vida (inicialmente 1)
* Vida máxima (inicialmente 3)
* Nivel (inicialmente 1)

El programa empieza con el menú principal donde tendremos la opción de ver los datos anteriormente mencionados, curar un punto de vida (cuesta 1 de dinero) o salir del programa.

Cada 10 segundos en el programa, hay un 80% de probabilidad de que se inicie una batalla con un oponente. Al momento de iniciar una batalla, debemos ganarle al oponente un numero de veces con respecto a nuestro nivel (ej: nivel 2, hay que ganarle dos veces) antes de perder toda nuestra vida. Si el jugador gana un round, el oponente pierde un punto, y si el oponente gana un round, nosotros perdemos un punto (de vida).

Si el jugador gana el encuentro, subimos de nivel en uno y aumentamos la vida máxima en uno. Si el jugador pierde el encuentro, debe mostrar todas las estadísticas y salir del programa (no olvide terminar todos los hilos de control). Si el jugador entra en batalla, avisarlo con un mensaje en pantalla y cambiar las opciones del menú a las opciones de batalla (Piedra, papel o tijera), ya que cuando el jugador se encuentra en batalla, no puede curarse puntos de vida. Cuando el jugador esta en batalla, mostrar la vida del jugador y del oponente en el menú.

Cada 7 segundos en el programa, el jugador gana +1 de dinero **siempre y cuando el jugador no se encuentre en batalla**. Si el jugador gana +1 de dinero, avisarlo con un mensaje en pantalla.

Cada 6 segundos en el programa, el jugador gana +1 de vida **siempre y cuando le falte vida (no esté en la vida máxima) y el jugador no se encuentre en batalla**. Si el jugador gana +1 de vida, avisarlo con un mensaje en pantalla.

# Ponderación

|  |  |
| --- | --- |
| Elemento | Puntaje |
| Uso correcto de hilos de control | 5 |
| Resolución del problema | 3 |
| Terminar los hilos de control y liberación de memoria (si se utiliza) | 2 |

# Comentarios adicionales

Subir el documento como un archivo de **Word (.docx),** incluyendo su nombre completo, número de cuenta y repositorio de GitHub.

**Asegúrense de probar su programa y de que el código sea legible.**

Se aplicarán penalizaciones por quiebres en el código, interrupciones o errores inesperados en el programa y por mala indentación del código. Por cada una de las penalizaciones encontradas se les restara un punto de la nota final de su laboratorio.

Si el código no es legible para otras personas o está mal indentado, entonces los instructores no nos molestaremos por tratar de entender que hace su programa en caso de que se presente un error en el mismo.