**UNIVERSIDAD TÉCNICA NACIONAL**

**CARRERA: INGENIERÍA DEL SOFTWARE**

CURSO PROGRAMACIÓN I.

CÓDIGO: ISW-211

PROFESOR: EFRÉN JIMÉNEZ DELGADO.

PROYECTO: I

**I. OBJETIVO GENERAL**

Crear soluciones para resolver problemas simples mediante el modelaje, diseño detallado y programación, aplicando técnicas actuales de desarrollo de software orientado a objetos y considerando criterios de calidad apropiados, mediante la utilización de la notación UML.

**III. OBJETIVOS ESPECÍFICOS**

Resolver problemas mediante el uso de modelos abstractos, para la formación de un esquema lógico conceptual.

**Juego Aliados vs Enemigos**

**Enunciado**

Debemos crear un algoritmo que permita inicializar un juego. Para ello debe pedir los siguientes datos al usuario en que se indiquen la información necesaria.

1. Seleccionar el nombre del jugador número 1.
2. Seleccionar el nombre del jugador número 2.
3. Seleccionar la cantidad elementos a participar dentro la de la partida (mínimo 1 elementos por aliado y mínimo 1 elementos por enemigos).
4. Seleccionar la profundidad del área de juego (Largo y ancho)

**Ejemplo**

Cantidad de Aliados Destruidos: 0

Aliados

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | A |  |
|  | A |  |  | A |
| A |  |  |  |  |

Cantidad de Enemigos Destruidos: 0

Enemigos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | E |  |  |
| E |  | E |  | E |
|  |  |  | E |  |

Dentro de los métodos que se deberán implementar son:

1. AcertarBlancoAliado();
2. FallarBlancoAliado();
3. AcertarBlancoEnemigo();
4. FallarBlancoEnemigo();
5. SumarAliadosDestruidos();
6. SumarEnemigosDestruidos();
7. EscogerJugadorAliado();
8. EscogerJugadorEnemigo();
9. EscogerCeldaAliado();
10. EscogerCeldaEnemigo();
11. Ganador();
12. Perdedor();
13. Turno();
14. CantidadDePartidasJugadas();
15. CantidadDePartidasGanadasJugador1();
16. CantidadDePartidasGanadasJugador2();
17. CantidadDePartidasPerdidasJugador1();
18. CantidadDePartidasPerdidasJugador2();
19. Rendirse();
20. TotalDeAliadosDestruidos();
21. TotalDeEnemigosDestruidos();
22. Jugar();
23. IniciarNuevaPartida();
24. VerEstadisticas();

**Evaluación**

Todos los métodos del proyecto tienen un valor de 4pts excepto el método de jugar que tiene un valor de 8 puntos.

**Documentación Externa**

Portada.

Descripción del problema.

Solución del problema (la última solución, indique cuales son las estructuras utilizadas, lógica de cómo se trabajó para realizar el programa).

Análisis de Resultados (Resultados finales, indique que partes están completas, cuales defectuosos, y cuales no se realizaron y el porqué).

Conclusiones y recomendaciones al profesor (con respecto al proyecto, lenguaje, tiempo para la realización, la explicación en clase, horas de consulta etc.)

Literatura citada (mínimo de debe incluir dos con su respectivo resumen).

Documentación Interna

Fecha de inicio y Fecha última modificación.

Descripción para cada estructura (clase) y su uso en el programa,

Describir cada función e instrucciones dentro de estas.

**Aspectos Administrativos**

La tarea debe programarse en lenguaje Java. Debe enviar los programas fuentes.

El desarrollo de este trabajo se puede realizar en pareja como máximo.

Entrega de la tarea: 1 de Julio.

Se calificará con citas de revisión para la defensa de la tarea de ser necesario.

Si se encuentra copia la calificación será de cero para todos los implicados.

Si existe virus o si se encuentra mal identificando se rebajarán puntos por descuido del estudiante. Si no habré el proyecto no se calificara la parte programada.

Se recomienda que se comience a trabajar desde hoy.