Instituto de Tecnología ORT

Carrera: Analista de Sistemas

Materia: Fundamentos de Programación

EXAMEN FINAL 1 **Tema 1** **02/12/2021**

😃 **Leé por lo menos dos veces el enunciado antes de resolverlo.**

Enunciado

“Luz verde, luz roja”

Una organización clandestina lleva a cabo una competencia donde los jugadores arriesgan su vida a cambio de una importante suma de dinero para el único ganador/superviviente.

Nos piden simular el juego conocido como “Luz verde, luz roja”.

Este juego consiste en cruzar la meta, la cual se encuentra a **150 metros** de donde se encuentran los jugadores al empezar. Todos los jugadores están en un extremo (el metro cero). En el otro extremo de la pista (adonde los jugadores deben llegar) se encuentra una muñeca gigante. Si la muñeca les da la espalda los jugadores tienen “luz verde” y pueden correr hacia la meta y ponerse a salvo. Pero cuando la muñeca se dé vuelta para enfrentarlos se decretará la “luz roja” y todos los jugadores deberán detenerse y quedarse “congelados”. Todo aquel jugador que se detecte en movimiento será eliminado.
La muñeca completará un total de **10 rondas** cambiando de estado (una luz verde y una roja por ronda): primero “luz verde”, cuando los jugadores serán libres de moverse y correr todos los metros que puedan. Cuando la muñeca gire y se prenda la luz roja, la muñeca los "observará", revisando uno a uno a cada jugador para detectar quién está en movimiento. Quien se esté moviendo será eliminado.
En cambio, cuando un jugador alcance o supere la meta estará a salvo y deberá moverse junto a los otros supervivientes. También se moverán (a otro lado) los que sean eliminados. Al cumplirse la décima luz verde y prenderse la luz roja por última vez todos aquellos jugadores que no hayan conseguido llegar a la meta serán eliminados sin importar los metros recorridos.

Cada jugador es identificado de forma unívoca con un número entero. Si hiciese falta se pueden agregar los atributos que considere necesarios para resolver el comportamiento descrito antes.

Métodos provistos:

Para saber cuántos metros avanzó un jugador en una ronda se provee el método:
`private double obtenerDistanciaRecorrida()` de la clase Jugador, el cual no recibe parámetros y siempre devuelve un número que representa la cantidad de metros que el jugador se desplazó en una ronda.

Para saber si un jugador está quieto durante una luz roja se provee el método:
`public boolean estaEnMovimiento()` de la clase Jugador, el cual no recibe parámetros y devuelve un valor booleano indicando si el jugador está moviéndose o no.

Estos dos métodos provistos NO deben diagramarse en NS+, solamente deben agregarse al diseño e invocarse en donde se considere necesario al desarrollar los métodos requeridos en NS+.

Instituto de Tecnología ORT

Carrera: Analista de Sistemas

Materia: Fundamentos de Programación

EXAMEN FINAL 1 **Tema 1** **02/12/2021**

Se pide:

- Confeccionar el diagrama UML que describe el escenario del enunciado incluyendo los atributos de cada clase y los métodos a desarrollar y aquellos que creas conveniente.
- Desarrollar el método **`jugar()`** de la clase **Juego** que no recibe parámetros y devuelve uno de los siguientes valores:
    - UNICO_GANADOR: cuando sobrevive exactamente un solo jugador.
    - MENOS_DE_LA_MITAD: cuando la cantidad de jugadores que sobrevivieron al juego sea menor o igual a la mitad de los jugadores que arrancaron a jugar.
    - MAS_DE_LA_MITAD: cuando la cantidad de jugadores que sobrevivieron al juego sea mayor a la mitad de los jugadores que arrancaron a jugar.
    - SIN_SOBREVIVIENTES: cuando ningún jugador haya sobrevivido.

  Nota: *En el desarrollo de este método se dará especial énfasis a la correcta modularización y distribución de tareas y/o responsabilidades.*

  Importante: El juego posee una colección con todos los jugadores que participarán del mismo y dos colecciones más donde quedarán distribuidos los jugadores al terminar el juego:
    - En una colección deben quedar cargados aquellos jugadores que sobrevivieron.
    - En otra colección deben quedar cargados los jugadores que fueron eliminados.
    - La colección original debe quedar vacía.

Criterios

Para considerar aprobado el examen, el mismo debe resolver lo pedido y aplicar los siguientes conceptos de la programación orientada a objetos:

- Detección de clases, atributos, métodos y relaciones (asociativas y de uso).
- Modularización reutilizable y mantenible usando métodos con correcta parametrización.
- Asignación de responsabilidades a cada clase y correcto encapsulamiento.
- Manejo del concepto de instancia y de la interacción entre objetos.
- Manipulación de listas de objetos (ArrayList) y su uso en ciclos condicionales y for-each, eligiendo siempre el más adecuado en cada situación.
- Manejo de diagramas Nassi-Schneiderman y UML de clases.

% Correcto	0 a 20	25 a 45	50 a 55	60	65 a 70	75	80	85 a 90	95	100
Nota	1	2	¿4?	4	5	6	7	8	9	10