

Algoritmos y Programación III – 75.07

Cursos 1, 2 y 3

Primer cuatrimestre 2009

Trabajo Práctico

Tower Defense 09

Enunciado General

Introducción

El presente trabajo consiste en el desarrollo de un juego del tipo “Tower Defense”.

Descripción básica del juego: El jugador debe defender la puerta de salida del terreno, evitando que los enemigos que entran al terreno puedan llegar a dicha puerta. Para poder eliminar a los enemigos, el jugador debe colocar obstáculos y torres de defensa en el terreno. Los obstáculos dificultan el paso de los enemigos y las torres de defensa disparan misiles contra ellos.

Pueden encontrar varios ejemplos de este juego online (<http://towerdefencegames.com/>).

El juego consta de los siguientes elementos:

- **Escenario**

El escenario del juego es un terreno cuadrado ó rectangular. En la parte superior del mismo habrá una puerta de entrada, y en la parte inferior una puerta de salida.

Desde la puerta de entrada y hasta la puerta de salida habrá una ruta que conecta las 2 puertas. Esta ruta no será una línea recta, sino que deberá tener varias curvas en su trayecto.

- **Enemigos**

Los enemigos ingresan al terreno por la puerta de entrada, y circulan a través de la ruta definida en el mismo hacia la puerta de salida.

Hay distintos tipos de enemigos, definidos por su velocidad, y su resistencia a los misiles. Se pide, como mínimo, los siguientes enemigos:

Nombre enemigo	Velocidad	Resistencia
Hormiga	1	1
Araña	3	2
Cucaracha	5	4
Mosca (vuela)	4	3

- **Obstáculos**

El jugador puede colocar diferentes tipos de obstáculos sobre el camino para dificultar el pasaje de los enemigos.

Hay distintos tipos de obstáculos, definidos por su costo, y el efecto que producen sobre los enemigos. Se pide, como mínimo, los siguientes obstáculos:

Obstáculo	Costo	Efecto
Arena	10	Reduce la velocidad del enemigo en un 50%
Pegote	20	El enemigo queda frenado por un período de tiempo, luego sigue avanzando a velocidad normal.

Estos obstáculos no afectan a los enemigos voladores.

- **Torres**

El jugador puede colocar diferentes tipos de torres al costado del camino (nunca sobre el camino). Las torres disparan misiles a los enemigos.

Hay distintos tipos de torres, definidas por su costo, su alcance (hasta qué distancia llegan sus misiles) y el daño que producen sus misiles en el enemigo (por ejemplo, un enemigo con resistencia 4 debe recibir 2 misiles con daño 2 para ser eliminado). Se pide, como mínimo, las siguientes torres:

Torre	Costo	Alcance	Daño
Blanca	10	3	1
Plateada	20	5	2
Dorada	50	7	4

Dinámica del juego:

El juego tendrá varios niveles, definidos por: la forma del camino y la cantidad y tipo de los enemigos que salen de la puerta de entrada.

Al comienzo de cada nivel, el jugador tendrá una cierta cantidad de “vida” y una cierta cantidad de “dinero”.

Antes de comenzar el nivel, el jugador podrá usar su dinero para colocar obstáculos y torres en el terreno.

A su vez, durante el desarrollo del nivel, el jugador también podrá colocar nuevas torres y obstáculos utilizando el dinero que gana por eliminar enemigos.

Cada enemigo que consiga llegar a la puerta de salida disminuye en 1 la cantidad de “vida” del jugador (cuando llega a 0, el jugador pierde el juego).

Cada enemigo eliminado suma 1 unidad de dinero al jugador.

Normas generales de entrega:

- La estructura de carpetas que se entrega deberá tener como raíz una carpeta o paquete con el siguiente nombre: algo3c[1|2|3]g[número de grupo]. Donde 1, 2 o 3 es el curso en el cual está inscripto el grupo. Ej: algo3c1g8 es la carpeta que deberá usar el grupo 8 del curso 1 (turno tarde).
- La documentación deberá estar en 1 sólo archivo (texto y diagramas) preferiblemente en formato .pdf o en su defecto como .doc.
- Deberá coordinar con su ayudante asignado si se entregará el código fuente impreso y su formato.

Entrega 1

(La fecha de entrega del trabajo: semana del 11 de mayo de 2009)

Elementos a entregar:

- Una aplicación Java o C# que responda al enunciado del trabajo práctico, con todas las estructuras de datos, atributos y comportamiento necesario para la resolución del enunciado. (NO incluye visualización e interacción con el usuario).
- Pruebas unitarias automatizadas (utilizando JUnit para Java y NUnit para C#) que contemplen las siguientes condiciones:
 - Instanciación y mensajes de todos los elementos desarrollados
 - Creación de autos completos y mensajes los mismos
 - Interacción entre enemigos, obstáculos y torres, con combinaciones varias de los mismos
- Diagrama de clases de las clases principales de la aplicación.
- Diagrama de secuencia de los siguientes casos:
 - Envío de mensajes a la torre, donde se muestre claramente como interactúan los diferentes elementos de la misma.
 - Interacción entre el enemigo y obstáculo.
 - Interacción entre el enemigo y la torre.

Entrega 2

(La fecha de entrega del trabajo: semana del 29 de junio de 2009)

Elementos a entregar:

- Programa Java o C# que utilice el modelo desarrollado en la entrega 1 y agregue interfaz gráfica e interacción con el usuario. Se deberá desarrollar utilizando el patrón MVC que se verá en clase, de forma tal de minimizar el acoplamiento entre la aplicación visual y el modelo.
- Se deberá incorporar la funcionalidad de persistir el juego en un estado dado, para retomar el mismo en otro momento.
- Pruebas unitarias de las funcionalidades nuevas desarrolladas, y de las modificaciones a funcionalidades ya desarrolladas.
- Diagramas actualizados del modelo, incorporando los cambios y correcciones realizadas.
- Diagramas de clases de las estructuras nuevas surgidas de la implementación del modelo de visualización e interacción con el usuario.
- Diagramas de secuencias acordados con el ayudante.
- Diagrama de estados general de la aplicación.