Primer entrega del Trabajo Práctico

Clash of UNLA

Algoritmos y Estructuras de Datos

Ing. Damián Santos – Lic. Romina Mansilla – Agustín Di Stefano

UNLA

2019

|  |  |  |
| --- | --- | --- |
| DNI | Nombre | Evaluación Individual |
| 35639606 | Oscar Ruina |  |
| 39274740 | Micaela Jaldin Montero |  |
| 401336661 | Nicolás Romero Ortiz |  |
| 35243071 | Rodrigo Aguirre |  |
| 33691227 | Ariel Nicolas Heredia |  |
| 39450435 | Esteban Emanuel Rodriguez | |

**Primer Entrega:** “Estrategia de Resolución”.

Para esta entrega deberá presentarse un documento impreso (en un folio o carpeta) conteniendo, como mínimo:

* Carátula de presentación con los datos de los integrantes del equipo (primer hoja de este documento).
* Índice de contenidos (en caso de ser necesario).
* Desarrollo de la estrategia de resolución detallando:
  + Estructuras a utilizar (arrays, pilas, listas, colas).
  + TDAs y sus relaciones (diagrama de interacción de todos los componentes).
  + Estrategia de resolución de operaciones.
* División del trabajo y cronograma: se deberá presentar la división de actividades dentro del grupo y un cronograma de trabajo a alto nivel.

Todas las hojas deben estar numeradas.

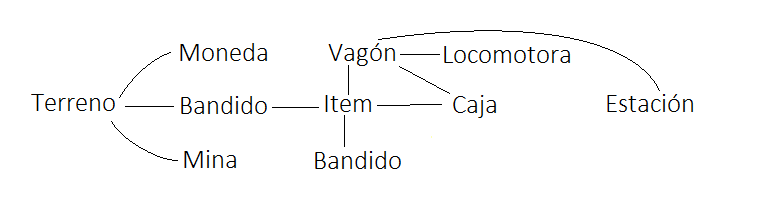
* Estructuras a utilizar:

Para desarrollar la aplicación en c++ usaremos las siguientes estructuras al día 10/05/2019:   
-Listas.

* TDAs que utilizaremos:

-TDA que maneje la creación de la ventana del juego llamado ManejoVentana  
 -TDA que se encargue de manejar la creación del terreno dentro de la ventana llamado Terreno.  
 -TDA Estación que se encargara de generar los vagones para el tren.  
 -TDA Moneda se genera aleatoriamente por el terreno y se usa para compra vagones en la estación.  
-TDA Locomotora.  
-TDA Vagón.  
-TDA Caja.  
-TDA Ítem.  
-TDA Bandido.

* Relación de los TDAs:



* Composición de los TDAs y sus operaciones:

**Terreno**

Composición estructura:

Int Width,height;

Array[][] SDL\_Rect;

Textura textura;

Operaciones:

cargarArray();

cargarTerreno();

dibujarTerreno();

**EstacionTren** //esto va a ser un trigger entre el rectángulo que forma x,y w,h. Lo único que va a hacer es llamar a crearVagon en la clase tren.

Composición estructura:

Int posX,posY;

Int width,height;

Textura textura;

Operaciones:

crearEstacion();

eliminarEstacion();

dibujarEstacion();

updateEstacion();

**Locomotora**

Composición estructura:

Int posX , posY;

Int width,height;

Textura textura;

Enum dirección; //para ver donde está yendo el tren;

Lista listaVagones;

Operaciones:

crearLocomotora();

eliminarLocomotora();

destruirLocomotora();

dibujarLocomotora();//para render en background

updateLocomotora();//para cambiar la posición en donde va a estar el tren;

**Vagón**

Composición estructura:

int posX,posY;

int width,height;

Textura textura;

Enum tipoVagon; //para que tipo de mineral;

Int tamaño; //el tamaño que sale de la compra del vagón dependiendo la cantidad de monedas

Lista listaCajas;

Operaciones:

crearVagon();

destruirVagon();

eliminarVagon();

destruirVagon(); // para cuando el bandido destruye el ultimo vagón;

dibujarVagon();

updateVagon();

buscarVagon(); //para buscar el tipo de vagon

**Cajas** //el tipo de caja sale del vagón.

Composición estructura:

Int Tamaño;

Int utilizado;

Operaciones:

agregarCaja();

eliminarCaja();//para eliminar al final

destruirCaja();//para cuando el ladron destruye una caja

modificarCaja();//para cuando el ladron roba de una caja

**Mina**

Composición estructura:

Int posX,posY;

Int width,height;

Textura textura;

tipoMina; //el tipo de mineral de esa mina

listaMenasMineral;

Operaciones:

crearMina();

eliminarMina();

destruirMina(); //para cuando el tren pasa sin el espacio necesario.

dibujarMina();

updateMina(); //esto es por si queremos agregar una animación al crear o destruir la mina.

**Bandido**

Composición estructura:

int posX,posY;

int width,height;

Textura textura;

Enum tipoLadron;

Int cantidadARobar;

Int vidaPrincipio;

Int vidaUtil;

Array areaDeAtaque // de la fila y columna que va a estar el bandido, desde fila-1 a fila+1 y columna-1 a columna+1;

Operaciones:

crearLadron();

eliminarLadron();

destruirLadron();

setearAreaDeAtaque();

atacar();

dibujarLadron();

updatearLadron(); //para actualizar los movimientos del bandido.

**Moneda**  
Composición estructura:

int posX, posY;  
int width,height;  
Operaciones:  
crearMonedas();  
updateMonedas();  
setMonedas();

**MenaMineral** //el tipo sale de la mina

Composición estructura:

(cola ) filaCajas: // cantidad X que se van a ir cargando en el tren cada vez que pasa por la mina

int vidaPrincipio;

Operaciones:

eliminarMena()

crearMena();

destruirMena();

* Cronograma

-Semana 11/05 al 17/05 Archivos generados: parámetros.txt,comanda.txt, minas.txt; TDA hechos: ManejoVentana, Terreno y Estación.  
-Semana 18/05 al 24/05 TDA hechos: Moneda, Locomotora, Vagon.  
-Semana 25/05 al 31/05 TDA hechos: Caja, Item, Bandido.  
-Semana 1/06 al 14/06 prueba y corrección de todo el proyecto.