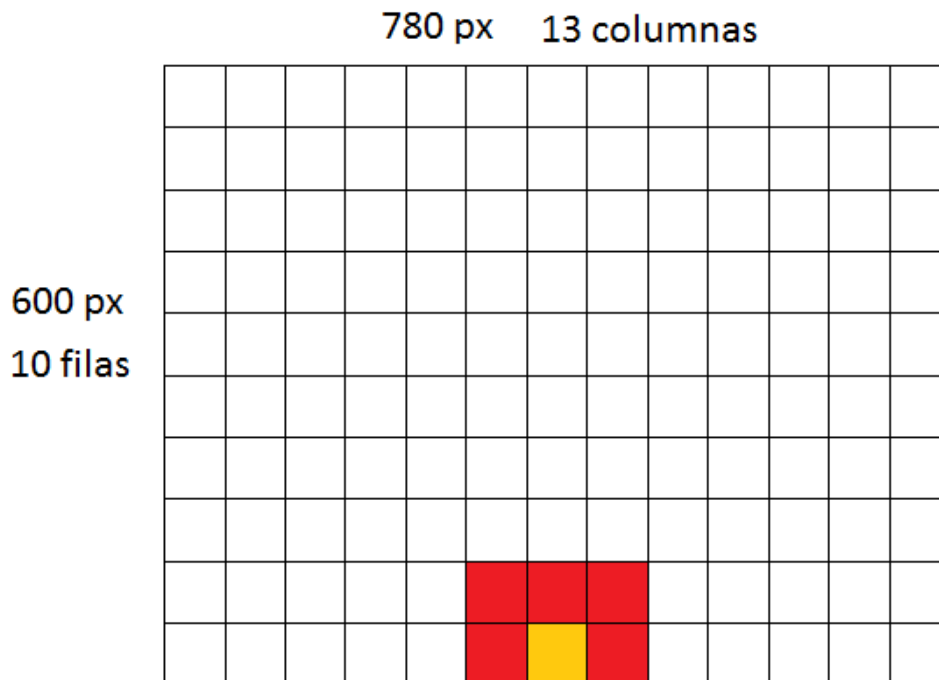


## Pseudocódigo de generación del mapa

Inicialmente existirá una clase que creara el panel base que contendrá dos paneles, uno correspondiente al terreno de juego, y otro encargado de mostrar datos del juego tales como las vidas, el puntaje. Etc.

La dimensión del panel correspondiente al terreno será de 780x600 pixeles, donde cada celda tendrá un tamaño de 60x60 pixeles dando como resultado un mapa de 13x10 celdas.

### Modelo Básico de un mapa



Para la carga de las celdas del mapa se utilizara una clase llamada “GeneradorDeMapas” la cual proveerá métodos “creadores de mapas” que leerán un archivo de texto con información acerca de que tipo de obstáculo corresponde en cada celda y retornara una matriz con dicha información.

La clase Generador de mapas se encargara de detectar cualquier inconsistencia en el archivo que contiene la información codificada del mapa, como por ejemplo si se intenta crear el águila en un lugar que no es el indicado.

El archivo con la información codificada del mapa contendrá filas de números separadas por guiones (por ejemplo: 1-1-1-1-1-2-3-2-1-1-1-1-1) donde cada número corresponde con un obstáculo:

- 0      Vacío
- 1      Pared de ladrillos
- 2      Pared de hierro
- 3      Arboles
- 4      Agua

El archivo tendrá información acerca de las celdas de la fila 2 hasta la fila 13 ya que la fila 1 se generara automáticamente con celdas vacías y será el espacio en donde aparecerán los tanques enemigos.

#### **Pseudocódigo de un método generador de un mapa**

- Crear matriz de 13x10
- Para i=0 hasta 12  
matriz[0][i]="0" //inicializa la primer fila con celdas vacías
- Cargar el archivo de texto para lectura
- Para i=1 hasta 10  
Leer línea del archivo  
Copiar los caracteres delimitados por "-" en la fila correspondiente del arreglo
- Retornar la matriz

Finalmente la clase encargada de inicializar el mapa recorrerá la matriz y de acuerdo al carácter leído ("0", "1", "2", "3" o "4") creara dinámicamente el tipo de obstáculo y lo posicionara en el mapa.

#### **Pseudocódigo del movimiento de los jugadores y enemigos**

El tanque jugador se posicionara de acuerdo a coordenadas "X" e "Y" las cuales serán usadas para moverse dentro del mapa, de acuerdo a los eventos de teclado que se detecten el jugador podrá moverse por el mapa por medio de sus métodos arriba (), derecha (), etc. Cada uno de esos métodos moverá a la representación gráfica del tanque una cierta cantidad de pixeles en la dirección que corresponda.

Los enemigos se mueven de forma similar al tanque jugador con la diferencia de que el movimiento de los enemigos es aleatorio y no depende de eventos de teclado.

Los casos en donde las entidades entran en contacto entre si son explicados más adelante en una sección titulada como "interacciones dentro del juego"

## **Interacciones dentro del juego**

La única forma en que las entidades del juego interactúen es por medio de un contacto, un contacto ocurre cuando la representación gráfica de dos entidades intenten compartir algún pixel dentro del mapa, por ejemplo si un tanque que está pegado a una pared intenta moverse una cierta cantidad de pixeles en dirección a esa pared, el mapa detectara que esas entidades entran en contacto. Las únicas entidades que pueden provocar un contacto son las entidades que se mueven, como los tanques y las balas.

### **Interacción de los proyectiles con el resto de las entidades**

Cuando una bala es disparada, se posiciona en la pantalla y comienza a avanzar en la dirección que fue lanzada, en caso de que se detecte el contacto entre una bala y otra entidad, se llama al método "impactar" definido en la clase "proyectil" el cual estará implementado para cada tipo de bala. El método "impactar" está definido para cada entidad (tanque\_jugador, enemigo, obstáculo y águila), por lo tanto el método que se ejecute dependerá del tipo de bala y del tipo del objeto que se envía por parámetro. Cada método posee un comportamiento adecuado para el tipo de parámetro que recibe.

### **Interacciones entre jugador y enemigos con los obstáculos**

Cuando se detecta un contacto entre un tanque (enemigo o jugador) y un obstáculo, se invocara al método "Contacto (e: EntidadMovil)" definido en la clase "obstáculo" el cual tendrá un comportamiento definido para cada obstáculo concreto, es decir cada obstáculo se encargara de decidir si lo puede atravesar o no.

### **Interacción entre jugador y enemigos**

Si se detecta un contacto entre el jugador y algún tanque enemigo, el jugador perderá una vida y en el caso de tener más vidas disponibles volverá a nacer en el nivel 1.

### **Interacción entre tanque jugador y powerup**

Cuando se detecta un contacto entre un tanque jugador y un powerup se invoca al método "contacto (ta:Tanque\_Jugador)" definido en la clase "powerup" el cual tendrá un comportamiento definido para cada uno de los powerup concretos, algunos modificaran el estado del tanque pasado por parámetro (el tanque que los toco) y otros realizaran ciertas acciones sobre el mapa o sobre los enemigos.