



Laboratorio 3

TP Final Integrador



Barrionuevo Damián
Barrionuevo Sebastián
Billa Carla Anahí
Peñalva Nahuel

Introducción

Para el siguiente trabajo integrador presentamos una simulación de consola con 4 juegos(hasta el momento), un nostálgico Mame nos permitirá logearnos como usuario común o administrador, el primero podrá elegir jugar cualquiera de los 4 juegos y acumular puntos, el tipo de usuario administrador además podrá buscar algún usuario y ver el total de los usuarios y sus características.

Los juegos desarrollados son el Snake, Laberinto Mario, Frog y PingPong.

Para tres de los cuatro juegos estaremos presentando la versión actualizada para Java, ya que tuvimos el placer de crearlos en C. Durante el proceso de aprendizaje fuimos utilizando herramientas vistas durante las clases y buscando nuevas, ya que al ser una propuesta de juegos la lógica es vital. El pasaje de los juegos al nuevo lenguaje fue lo más desafiante, la adaptación de las funciones y la utilización de los frameworks y paneles a través del código. Creamos clases para cada juego, a su vez sus propias carpetas de imágenes y paneles, contenidas en la clase MAME, con sus propias imágenes y métodos.

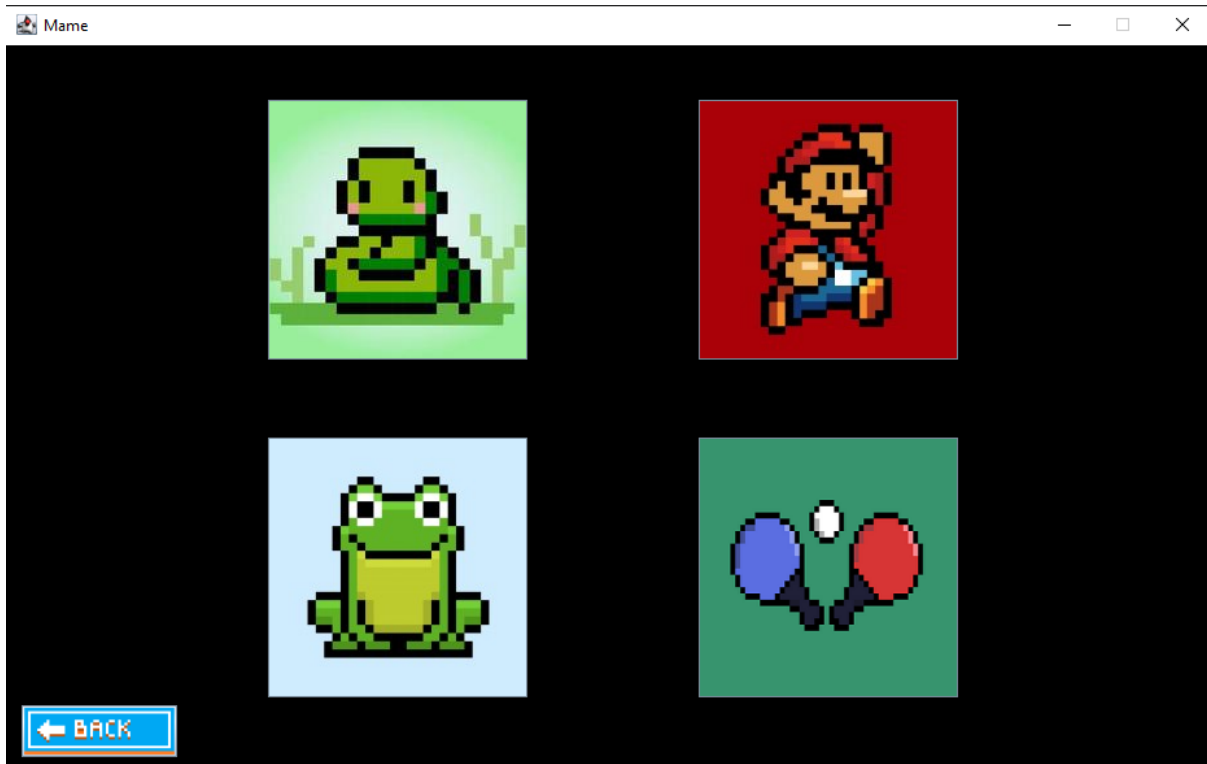
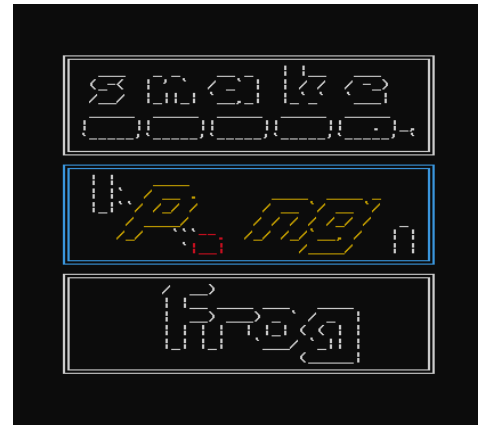
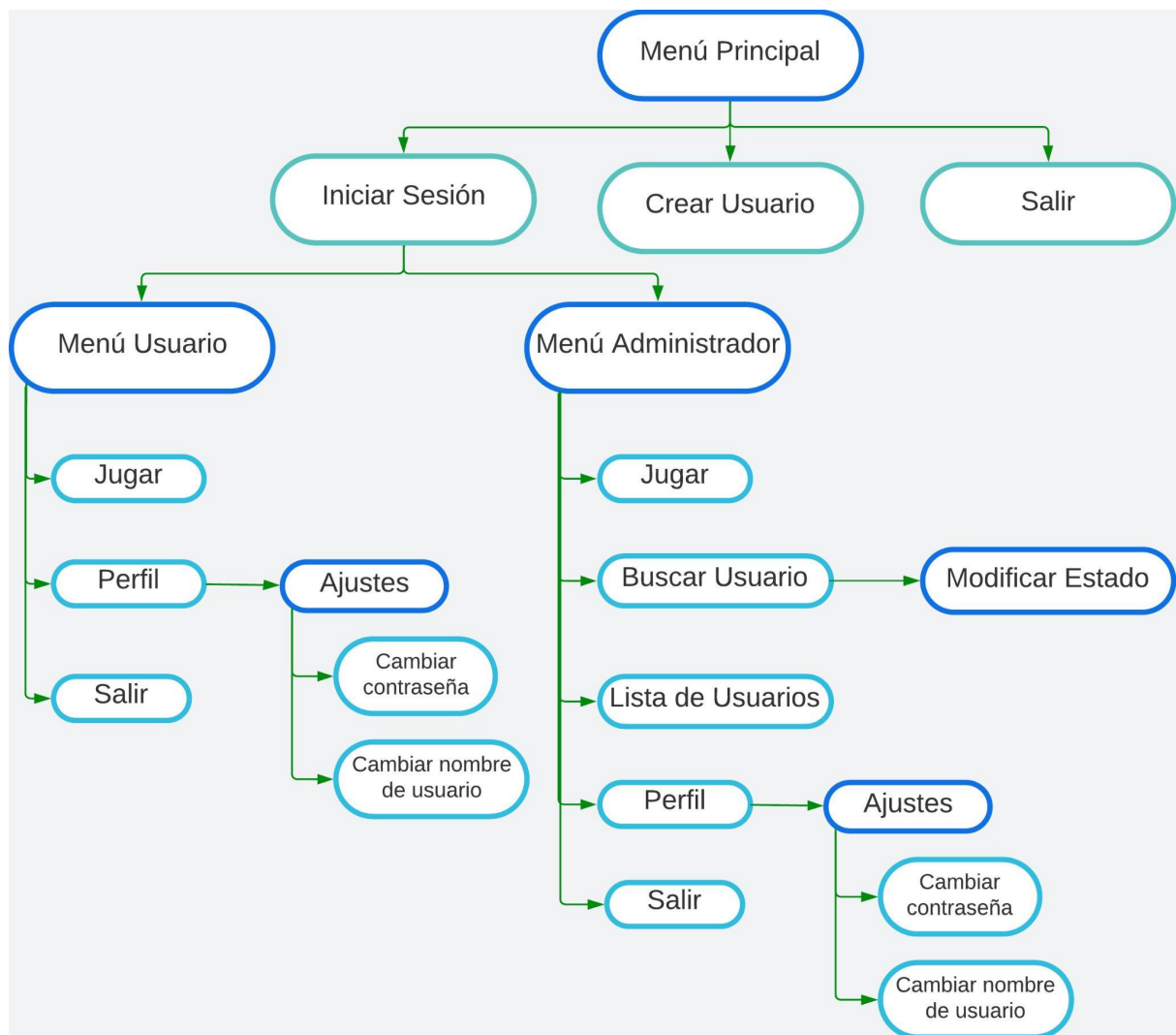
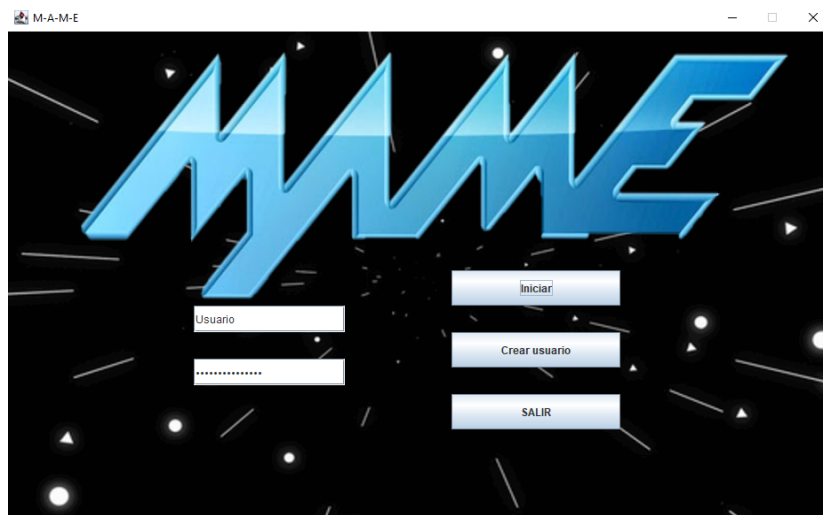


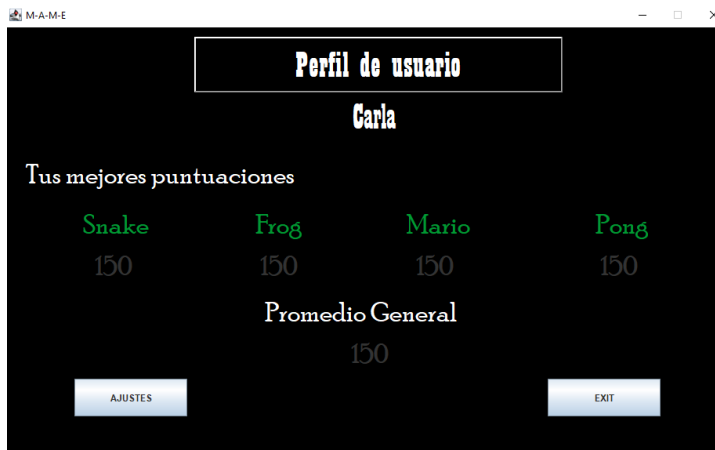
Diagrama de Pantallas



Pantalla de Inicio : Desde acá empieza todo, si ya tienes tu usuario ingresas, y si no tenes usuario lo podés crear.



Internamente el programa comprueba la validación del usuario ingresado, accediendo al archivo y buscando a través del Key un usuario existente, si se decide crear un usuario el programa comprueba que no haya usuarios repetidos, de no ser así te crea el usuario y te permite entrar con este mismo. Una vez ingresado el usuario habrá dos tipos de pantalla, la de un usuario común. Que tendrá las opciones de "Juego", "Perfil" o "Salir". Juego te mandará a una pantalla para selección el juego que elijas, Perfil te permitirá ver tus puntajes máximos obtenidos en cada juego, y además tendrás un botón de



"Ajustes" para cambiar tu usuario o contraseña.



O la de un Administrador, en esta pantalla se permitirá además de las opciones ya mencionadas, "Buscar Usuario" y "Lista de Usuarios". La de buscar usuario nos permite buscar un usuario por nombre, si este existe te lo muestra, y te permite cambiar el estado de activo a inactivo y viceversa. La lista de usuarios nos muestra todos los usuarios del programa registrados.

Búsqueda de usuario:

Ingrese nombre de usuario

Nahuel

Aceptar

Salir

M-A-M-E

Nombre: Nahuel
Tipo: 1
Estado: 1
Maximo puntaje en Frog: 150
Maximo puntaje en Snake: 150
Maximo puntaje en Pong: 150
Maximo puntaje en Mario: 150
Puntaje total: 150

Modificar Estado

Exit

Muestra de Usuarios Totales:

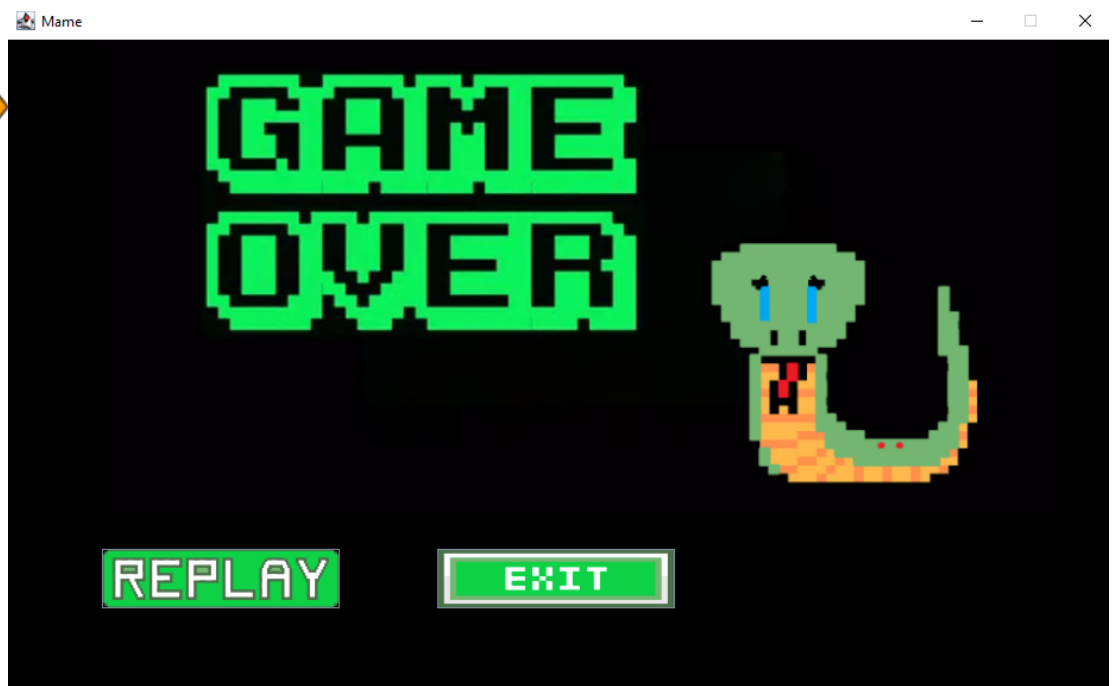
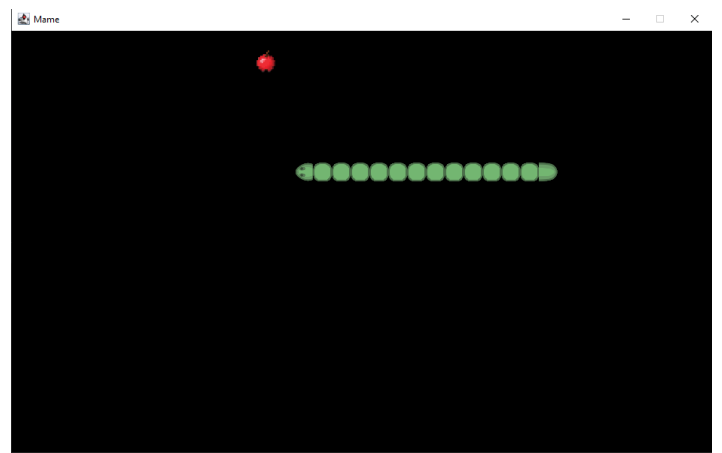
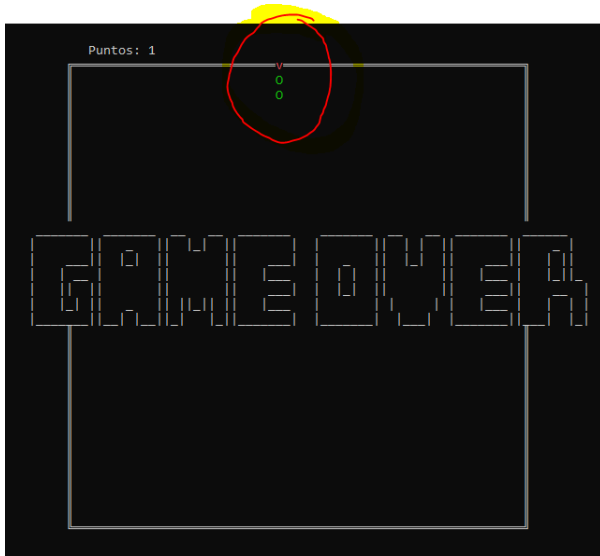
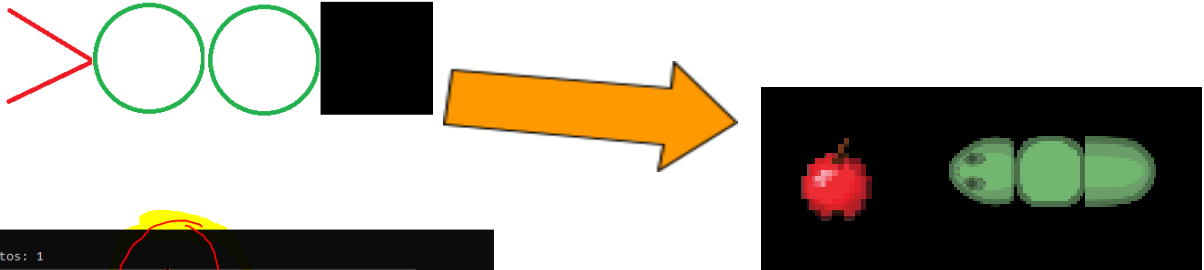
M-A-M-E

Usuarios

Nombre	Tipo	Activo	Frog	Snake	Pong	Mario	Total
BarackOrcBa...	0	1	150	150	150	150	150
Carla	1	1	150	150	150	150	150
Damian	1	1	150	150	150	150	150
DarthBaston	0	1	150	150	150	150	150
Dinius	0	1	150	150	150	150	150
ElPunza	0	0	150	150	150	150	150
Nahuel	1	1	150	150	150	150	150
TopDPS	0	0	150	150	150	150	150
Trahear	1	1	150	150	150	150	150
TutiFruti	0	1	150	150	150	150	150

EXIT

Snake (de C a Java):

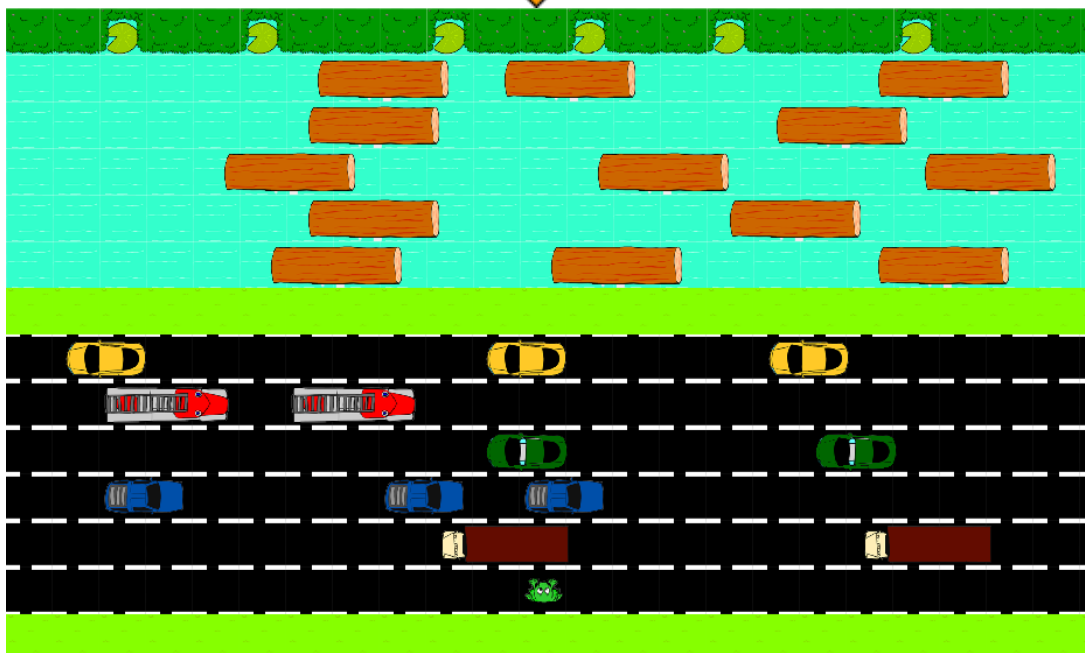


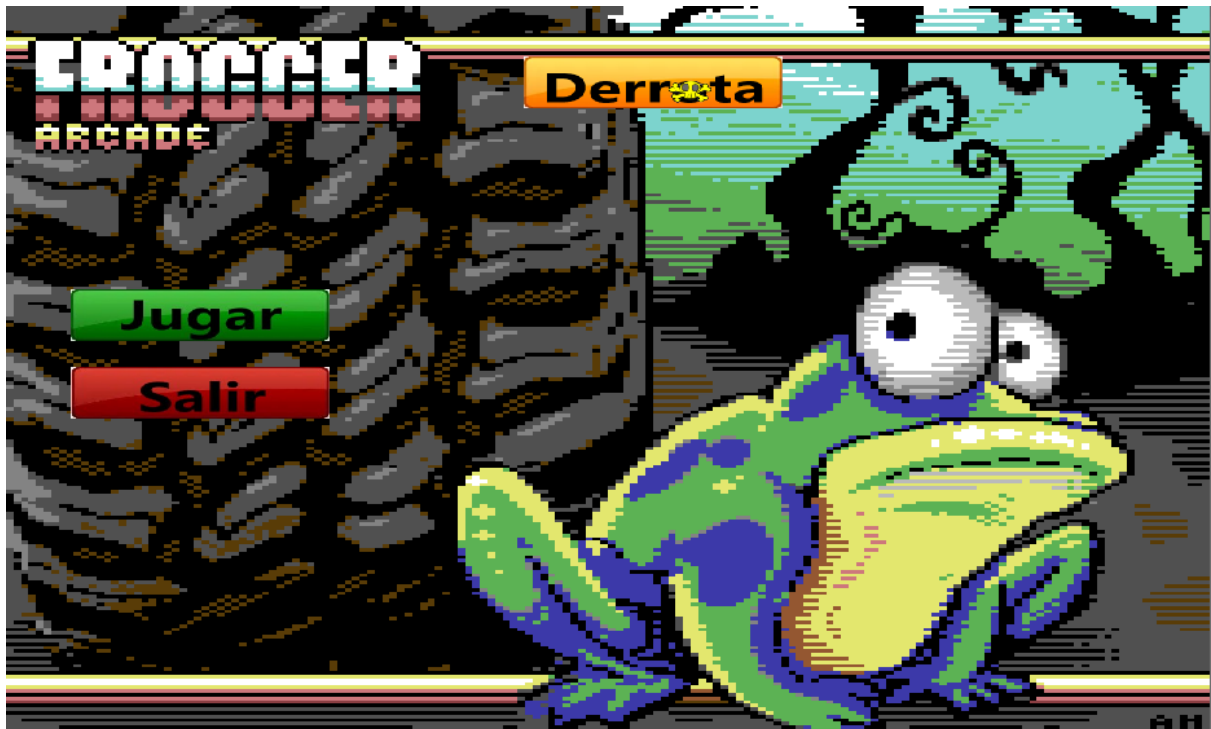
Función Destacada:

```
private void movimientoCabeza(int x, int y, List<Snake> s) {  
    int i = 0;  
    int posActualX, posActualY, posProxX, posProxY;  
    posProxX = s.get(0).getX() + x;  
    posProxY = s.get(0).getY() + y;  
    while (i < this.snake.size()) {  
        posActualX = s.get(i).getX();  
        posActualY = s.get(i).getY();  
        s.get(i).setX(posProxX);  
        s.get(i).setY(posProxY);  
        posProxX = posActualX;  
        posProxY = posActualY; i++;  
    }  
}
```

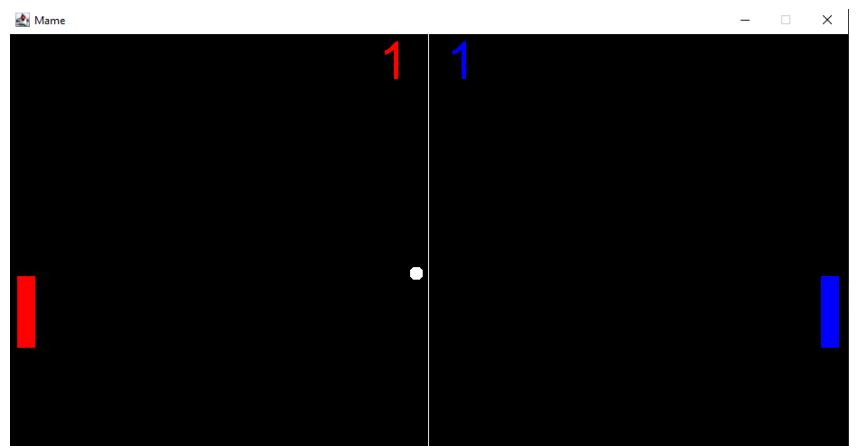
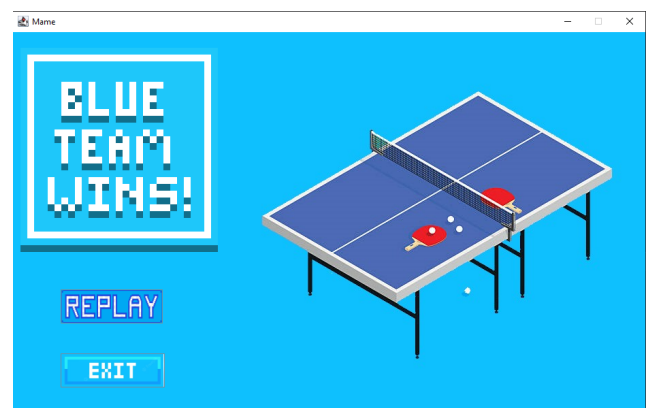
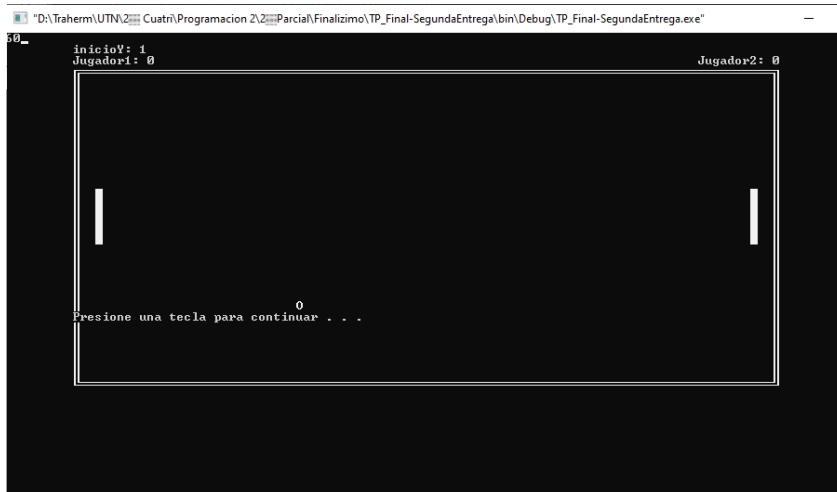
La función “movimientoCabeza” recibe la nueva posición en X e Y + la lista de estructuras “cabeza, cuerpo y cola”. Genera variables auxiliares, ya que solo vamos a trabajar con la X e Y de las estructuras, sin alterar lo demás. Asignamos la posición actual a dos variables llamadas posProxX y posProxY, luego mientras que “i” sea menor a válidos, copiamos la posición actual de la cabeza para no perderla, luego asignamos la posición nueva a la cabeza y colocamos la posición vieja de la cabeza en posprox, la cual será asignada al cuerpo y luego a cola, y así consecutivamente. Y mientras se produce el movimiento se cambian las imágenes de la cabeza y la cola dependiendo de la dirección.

Frog (de C a Java):





Ping Pong(De C a Java):



Laberinto Mario:

