

Universidad de San Carlos de Guatemala
Facultad de Ingenieria
Escuela de Sistemas
Sistemas Operativos 1

Manual Tecnico

Natthaliee Maria Molina Cruz

201212501

Contenido

Introducción.....	3
-------------------	---

Introducción

Objetivos

1. Implementar el algoritmo de Dekker para hacer uso de memoria compartida.
2. Implementar semáforos del sistema operativos GNU/Linux.
3. Poner en práctica los conocimientos adquiridos en el curso y laboratorio de sistemas operativos 1.

Descripción

El proyecto consiste en la creación de un juego similar a "Space Invaders" en lenguajes C. El juego será desarrollado para dos jugadores en simultaneo (modo versus). Para ello se deben utilizar dos instancias diferentes del mismo programa. El cual se podrá jugar utilizando conexión SSH y usuario local. El juego consiste en bando defensor y bando invasor y contara con un sistema de puntuación para definir el bando ganador.

La comunicación entre ambas instancias del programa se realizará por medio de memoria compartida, deben implementar la versión 5 del algoritmo de Dekker. También se debe hacer uso de semáforos en ambas instancias del juego.

Características de la aplicación

El juego se debe desarrollar en lenguajes C y debe implementar graficas basadas en texto que hagan del juego más llamativo.

La aplicación debe funcionar con memoria compartida ya que no debe mostrarse diferencias visuales en ambas instancias (punteo, posición, tiempo, etc.).

El juego debe iniciar cuando los dos jugadores estén listos. El juego terminará cuando uno de los dos jugadores se quede sin vidas o el score del defensor sea suficiente.

Lógica del Juego

El juego cuenta con dos bandos: defensor e invasor.

Defensor:

Es el encargado de derribar la línea de naves invasoras y comandante.

Invasor:

Es el comandante de las líneas de naves invasoras y es el encargado de indicar que nave debe disparar.

Inicio

El juego inicia cuando ambos jugadores se encuentran listos. Las líneas invasoras inician con un total de 20 naves. Estas naves se mueven de forma síncrona (al mismo tiempo, distancia y dirección). El defensor inicia con 0 puntos de score. Ambos jugadores inician con 5 vidas.

Punteo

Cada vez que el defensor logre darle a una nave invasora se le sumaran 10 o 15 puntos, según el tipo de nave que será definido más adelante). Si el invasor logra disparar y darle al defensor se le restará una vida al defensor.

Tiempo

El tiempo se debe mostrar en ambos jugadores, y debe ser continuo, no se debe reiniciar durante la ejecución del juego.

Invasores

Las naves invasoras se estarán moviendo de forma horizontal, y en ambas instancias del juego deben moverse igual. El movimiento de los invasores es automático, es decir no es controlado por el jugador invasor.

Las naves invasoras desaparecen al momento de hacer contacto con una bala. Existen dos tipos de invasores comunes y fuertes.

Invasores comunes:

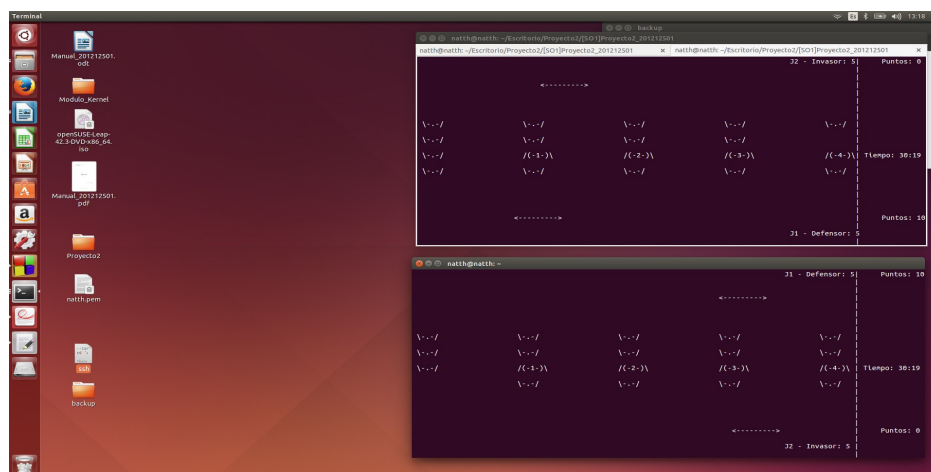
Al derribarlos suman 10 puntos al score del defensor. Deben existir 16 naves de este tipo. Forma: \-.-/

Invasores fuertes:

Al derribarlos suman 15 puntos al score del defensor. Estos son los únicos que pueden disparar en contra del defensor. Deben existir 4 naves de este tipo y estarán identificadas por un numero el cual será utilizado por el comandante invasor para indicar que nave será la que debe disparar. Forma: (/ -1 -\) (el número de nave debe ir al centro) .

Para el desarrollo de esta aplicacion se utilizaron muchas librerias, de las cuales se pueden mencionar las librerias para el desarrollo visual llamada ncurses la cual permite el desarrollo de las imagenes y el movimiento por medio de limpiar la pantalla y repintar. Ademas de ncurses tambien podemos mencionar hthread la cual permite la utilización de los hilos que son indispensables para toda la aplicacion.

Para el desarrollo visual del proyecto la libreria ncurses permite la utilizacion del comando "mvprintw" el cual nos pide un valor en 'x' en 'y' y uno o un conjunto de caracteres para mostrar. Ademas de esto nos brinda el comando 'refresh' el cual refresca las ultimas graficas agregadas en el screen con el comando antes mencionado sin el comando refresh es factible la posibilidad de imagenes congeladas o el de dibujar caracteres encima de otros, para el desarrollo de las naves se utilizo el conjunto de caracteres "<----->" la balas son "*".



Para los threads se utilizo su debida libreria pero ademas en conjunto con ellos se utilizaron semaforos y el algoritmo de dekker ambos para llevar un mejor control entre procesos y permitir sincronización entre ambos, esto nos permite un mejor control entre disparos, daños, y puntuacion de cada jugador mientras cada uno siga jugando, ademas se incluye otro thread el cual lleva control estrictamente del tiempo este es controlado por un semaforo ya que puede causar gran descontrol entre el proceso de repintado y los procesos de los tiros.

Requerimientos:

Para esta aplicacion solo es necesario un compilador de c como gcc o codebloks que ya incluye el compilador, ahora en linux el compilador de c ya viene incluido en la terminal desde el modo no grafico.

El juego se ejecuta desde cualquier terminal en linux, conectandose a la instancia Ec2 con el comando:

```
ssh -i "natth.pem" ubuntu@ec2-18-218-113-193.us-east-2.compute.amazonaws.com
```

Ejecutamos las dos instancias del juego con los comandos:

```
./spaceInvaders
```

```
./spaceInvaders2
```

Es necesario tener la llave de acceso ".pem". El juego inicia con un mensaje de bienvenida, presione enter para continuar.

El jugador inicial deberá esperar al jugador invasor. Es necesario abrir 2 interfaces de terminal lo que permite el uso para los 2 jugadores y dar inicio al juego.

Para Jugar las teclas son 'w', 'a', 'd' desde cualquiera de las 2 interfaces con la tecla w se dispara y con 'a' y 'd' es el movimiento.