

Instituto Tecnológico de Costa Rica
Ingeniería en computadores
Antonio González Torres
Harold Espinoza Matarrita (2019185140)
Fabricio Mena Mejia (2019042722)
Grupo #4

Proyecto #1: Space invaders

Tabla de contenido

| | |
|--|---|
| Introducción: | 2 |
| Descripción del Problema: | 2 |
| Diagrama UML:..... | 2 |
| Análisis de resultados:..... | 2 |
| Bitácora: | 3 |
| Estadística de tiempos: | 5 |
| Conclusión personal: | 5 |

Introducción:

Este proyecto tiene como objetivo aprender de las interfaces graficas de Python (GUI) tales como tkinter o pygame mediante la programación del juego "Space Invaders" creado por Toshihiro Nishikado. Nosotros decidimos utilizar tkinter y la versión de Python 3.7.2. El juego consiste en derrotar las naves enemigas sin recibir ningún disparo este contiene 3 niveles los cuales aumentan su dificultad a medida que se va avanzando de nivel.

Durante la ejecución del programa se despliegan los "invaders" que pueden ser de 3 tipos: Los normales que no atacan al jugador, los atacantes que disparan al jugador y los defensores que aguantan más de un disparo del jugador. Cada vez que se elimina una nave enemiga se otorga un punto al jugador.

El juego finalizara si el jugador es eliminado por los "invaders", si el jugador consigue pasar los 3 niveles o en la pestaña de "file" seleccionar la opción "quit" que preguntara al jugador si desea volver al menú principal o cerrar el juego directamente

Descripción del Problema:

Mediante Python y una (GUI) se debe realizar el juego denominado "space invaders", el cual debe contener: un jugador, naves alienígenas, meteoritos, disparos del jugador y enemigos y debe contener 3 niveles que serán accedidos cada vez que se pase el nivel anterior e iran aumentando de dificultad conforme se vaya subiendo de nivel.

Además, debe poder insertar un usuario con el nombre del jugador y guardarlo en un archivo csv, si el usuario ya se encuentra en el csv no deberá ser guardado. El usuario también se utilizará para el almacenamiento de los 5 puntajes más altos los cuales deberán ser guardados en un archivo json y estos se tienen que mostrar cuando el jugador pierda o supere los 3 niveles.

Diagrama UML:

Este se encuentra en un pdf aparte debido a su extencion, tiene el nombre De "Doc UML" y se encuentra en la misma carpeta

Análisis de resultados:

Decidimos utilizar tkinter ya que nos pareció más cómodo para trabajar.

Usamos un código de referencia para observar el comportamiento de los objetos y poder darnos una idea de cómo empezar el código.

Primero logramos mover la nave en todas las direcciones en una simple ventana, a partir de ahí insertamos los 4 tipos de aliens y que hacía cada uno de ellos. Introducimos las balas y los asteroides. Posteriormente elaboramos un menú para la introducción del usuario y aprendimos a utilizar los documentos csv.

Logramos mostrar los scores en tiempo real así como el nivel en el que se encuentra el jugador, también se muestran los 5 scores mas altos en la pantalla final del juego y aprendimos a guardarlos y leerlos de un archivo json.

Bitácora:

11/04/2019

19:00

A partir de un código base intentamos comprender el funcionamiento de las clases, como movilizar la nave, y añadir sorites

12/4/2019

20:00

Decidimos buscar información sobre cómo funciona tkinter, para así facilitar nuestro entendimiento sobre este y aprender a programar en él.

13/04/2019

12:00

Se investigo acerca del uso de los archivos. Json y como utilizarlos en python y creamos un programa para comprender su funcionamiento

14:00

Se investigaron los archivos csv y se creó un programa para la escritura y lectura de los usuarios

14/04/2019

12:00

Hicimos un programa con la nave defensora y logramos que se moviera en todas las direcciones y la preparamos para luego añadir un menú antes de iniciar el juego

15/04/2019

1:00

Investigamos sobre como añadir pistas de audio al programa, y como ejecutarlas al mismo tiempo que la animación, para así evitar que haya cortes en estas.

13:00

Se investigan los UML (diagrama de clases) y se inicia la parte escrita del proyecto

15:00

Logramos añadir pistas de audio al juego. Aprendimos el control de los "invaders"

17:00

Se añade una barra de menú con distintas opciones para el jugador. Añadimos un texto que muestra el nivel en el que se encuentra el jugador y el score en tiempo real

16/04/2019

13:00

Creamos una opción para reiniciar el juego

17:00

Añadimos un menú que permite introducir el nombre y botones para jugar y establecer el nombre que se guarda en un csv. Se añadieron comentarios para que el código pueda ser entendido con mayor facilidad

17/04/2019

14:00

se investigan más a fondo los .json

17:00

Se logran mostrar los scores en la pantalla final pero aún no se actualizan

18/04/2019

12:00

Se logra actualizar los scores en la pantalla final

17:00

Se arregla una línea del score. Se logra comprobar si el usuario ya estaba guardado y si no se guarda

21:00

Se agregaron asteroides al videojuego. Se investigó sobre threads y se lograron implementar en el programa. También se añadieron spawns de las balas enemigas y asteroides.

19/04/2019

12:00

Se introducen los últimos comentarios al programa fuente, se hace el diagrama UML y se termina la elaboración del documento.

Fin de bitácora

Estadística de tiempos:

| | |
|--|-----------------|
| Análisis de requerimientos | 04 horas |
| Diseño de la aplicación y diagrama de clases | 04 horas |
| Investigación de funciones | 12 horas |
| Programación | 15 horas |
| Documentación interna | 03 horas |
| Pruebas | 15 horas |
| Elaboración documento | 05 horas |
| Total | 58 horas |

Conclusión personal:

Aprendimos a utilizar la interfaz grafica (GUI) tkinter de python, así como el comportamiento de distintos objetos, widgets como entry, crear un texto, distintos messagebox o ventanas emergentes, barra de menú y submenús que realizan distintas opciones. También aprendimos el manejo de archivos csv y json en python además de introducir pistas de audio