

**UNIVERSIDAD PERUANA DE CIENCIAS APLICADAS**

Inteligencia Artificial

Sección: WV71

**Proyecto de TB1: “AUTO SNAKE”**

**INTEGRANTES**:

Diego Alonso Sanchez Parra - u201612784

David Obhed Porta Montes - u201812241

Miguel Antonio Rosales Silva - u201714279

**PROFESOR**:

CALDERON VILCA, HUGO DAVID

**Ciclo 2021-1**

**ÍNDICE**

**I. Planteamiento del juego a resolver**

**II. Fundamentación de cómo adapta o usa la técnica o el algoritmo de Inteligencia Artificial**

**III. Explicación de la heurística y de qué manera permite resolver el problema en cuestión**

**IV. Detalles del código fuente de la aplicación**

**V. Pruebas de uso, ejecución y descripción de las funcionalidades**

**VI. Conclusiones**

**VII. Referencias bibliográficas**

**I. Planteamiento del juego a resolver**

En los últimos años se han utilizado videojuegos clásicos para probar diferentes algoritmos de inteligencia artificial que ayudan a comprender a profundidad el proceso y solución que realizan cada uno. En muchos casos estos algoritmos también ayudan a testear estos juegos buscando el método en el que el jugador pueda ganar, esto con el fin de comprobar si es posible que el jugador pueda ganar. Estos algoritmos dan como resultado diferentes escenarios en el que el jugador puede ganar el juego, además en mostrar el tiempo o movimientos que demora en completar también son factores que se tienen que tomar en cuenta.

En este trabajo utilizaremos el juego de SNAKE un videojuego que ha sido lanzado en 1976 y que ha mantenido su popularidad desde ese entonces. El juego está basado en el que el jugador controla a una criatura similar a una serpiente que se desplaza alrededor de un mapa o plano delimitado, en donde el objetivo es recolectar los alimentos, evitando que el jugador choque con los bordes del plano y también evitando colisionar con su propia cola.

Utilizaremos algoritmos que hemos aprendido en el curso de Inteligencia Artificial que nos ayudará a buscar el camino correcto que tiene que tomar el jugador, en aplicaciones reales, son usadas para testear los caminos correctos y más cercanos entre dos puntos: como en las aplicaciones que ayudan a buscar la mejor ruta tenemos a la aplicación Waze y Google Maps, que podemos compararlo con el videojuego que hemos tomado, ya que el objetivo de ambos es buscar la mejor ruta de un punto a otro.

**II. Fundamentación de cómo adapta o usa la técnica o el algoritmo de Inteligencia Artificial**

Para este trabajo lo vamos a dividir en dos secciones, ya que hemos realizado dos juegos de SNAKE cada uno con un diferente algoritmo, entre estos dos algoritmos se va a escoger el más óptimo.

**Algoritmo Genético:** Es un algoritmo de optimización de búsqueda y aprendizaje que se basa en la evolución biológica y selección natural. Al ser un modelo de aprendizaje automático se le asigna entradas perceptivas y salidas de acción. Para este juego la entrada sería la distancia de la serpiente en las cuatro direcciones principales y la salida sería una acción como girar a la derecha o izquierda.

**Algoritmo A\*:** Es un algoritmo de búsqueda en grafos para buscar el camino con el costo más bajo, en este caso es tiene las entradas y salidas similares al del algoritmo genético, su entrada sería la distancia de la serpiente en las direcciones Derecha, Arriba, Abajo, Izquierda, y de salida y girar a las cuatro principales direcciones. Este algoritmo ayuda al usuario cuáles son los caminos de bajo costo para llegar al objetivo que en este juego que la serpiente coma las manzanas sin tener que colisionar entre sí o con un obstáculo.

**III. Explicación de la heurística y de qué manera permite resolver el problema en cuestión**

Los algoritmos de solución empleados en el juego de SNAKE son útiles para poder encontrar el mejor camino, ya sea por encontrar el camino de menor costo o encontrar el escenario, esto requiere que el algoritmo se entrene con las tomas de decisiones que escoja la IA para poder generar un mejor resultado.

**Algoritmo Genético:** En el algoritmo genético, se crea según el tamaño de población de las serpientes, en ese instante cada una se mueve aleatoriamente y se guardan sus puntajes se escoge a las 25 mejores para volver a crear otra generación y así hasta que se hagan todas las generaciones y al final te devuelve el mejor escenario de serpiente.

**Algoritmo A\*:** En el caso de A\* es muy diferente cada movimiento que hace la serpiente tiene que tomar la decisión de qué dirección tomar para llegar al objetivo sin colisionar y que sea de menor costo posible, además de que cada movimiento guarda los posibles escenarios en el que la serpiente llegue al objetivo.

Gracias a estos dos algoritmos tendremos una perspectiva de cómo se utiliza la IA como implementación en pruebas o simulación a tiempo real, en lo que sería encontrar la ruta más óptima, pero también el programa necesita entrenar en diferentes mapas o entornos, mapas con más obstáculos que puedan aumentar la complejidad de generar resultados.

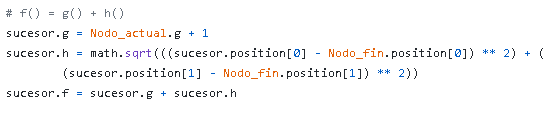
**IV. Detalles del código fuente de la aplicación: si ha utilizado algún framework, especificar librerías si fuera el caso que está usando, API etc., todo lo que fuera necesario para poner en marcha su aplicación.**

**Librerías usadas en el Algoritmo A\*:**

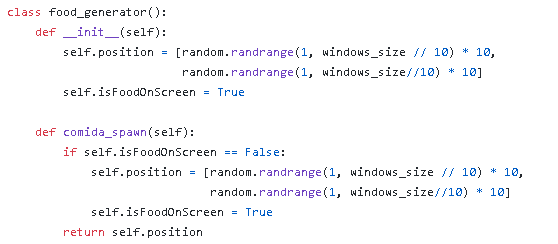
Pygame: Que es una librería que permite la creación de videojuegos en 2D de manera sencilla. Esto lo utilizamos para la GUÍA del juego.



Librería math: Esta librería nos proporciona funciones matemáticas como por ejemplo la raíz cuadrada y potenciación. En este trabajo lo utilizamos para resolver operaciones matemáticas.

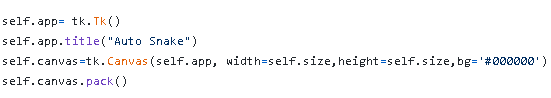


Librería random: Una librería que nos genera valores aleatorios según el intervalo que nosotros definimos. Lo utilizamos para generar la posición de la manzana o comida de la serpiente.

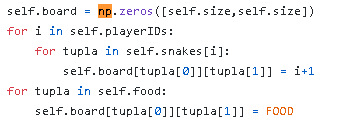
****

**Para el juego de Snake Algoritmo Genético utilizamos las siguientes librerías:**

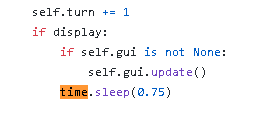
Tkinter: Que es una librería para generar una interfaz de gráfico, lo utilizamos para poder generar la interfaz de usuario para que se visualice la serpiente y los obstáculos.



Numpy: Es una librería que utilizamos para crear vectores y matrices.



Librería time: Lo utilizamos para controlar el tiempo de la animación.



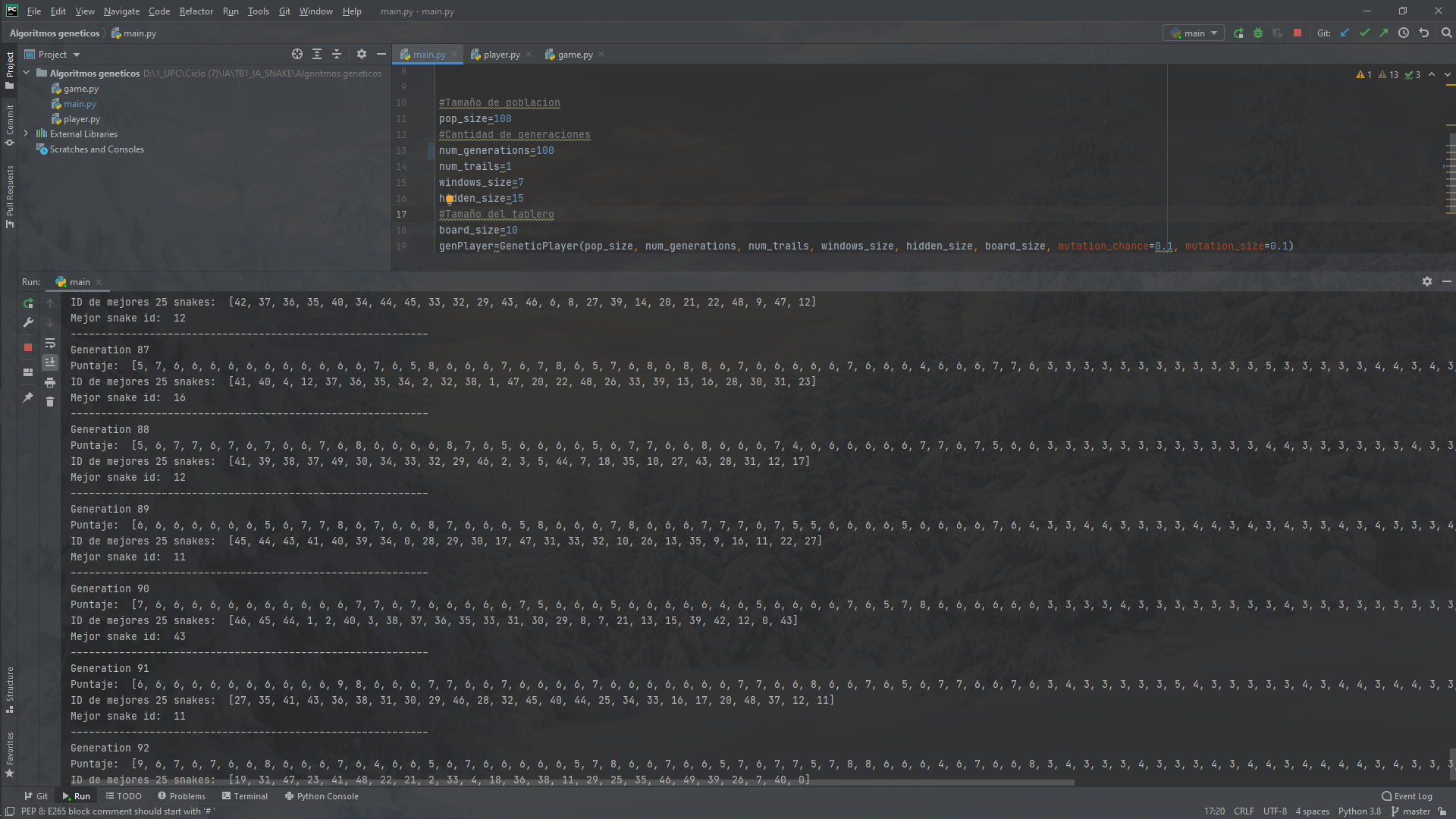
Eso hace que cada paso que da el snake demore 0.75 segundos.

**V. Pruebas de uso, ejecución y descripción de las funcionalidades**

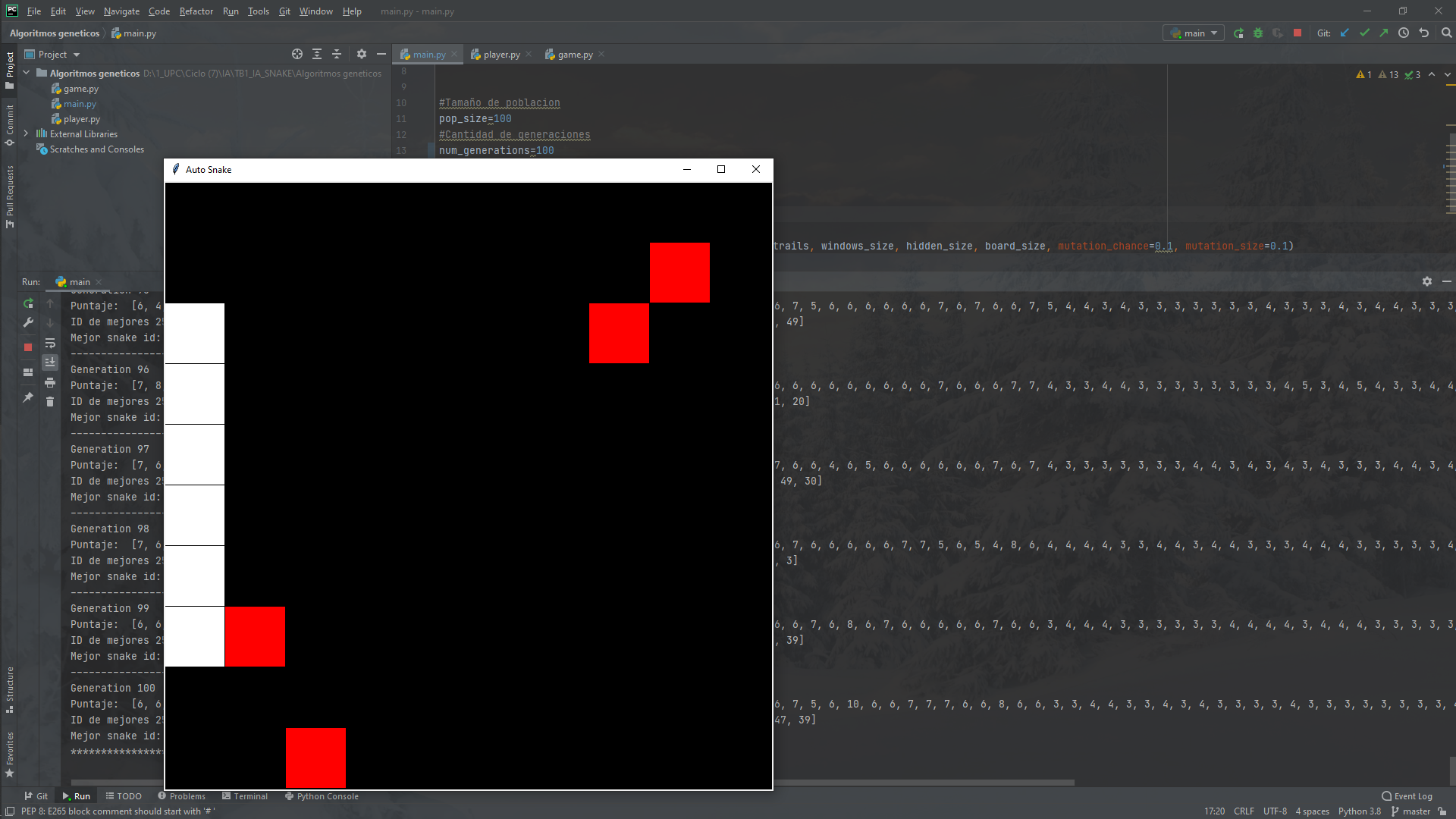
**Algoritmo Genético:**

Antes de comenzar, es importante resaltar que las pruebas con el algoritmo genético tienen por limitantes a la cantidad de generaciones, población y la aleatoriedad al momento de crear nuevas poblaciones, por lo que no todos los casos tendrán este mismo resultado.

Primero comencemos en un tablero con dimensiones de 10x10, 100 individuos por población y 100 generaciones:

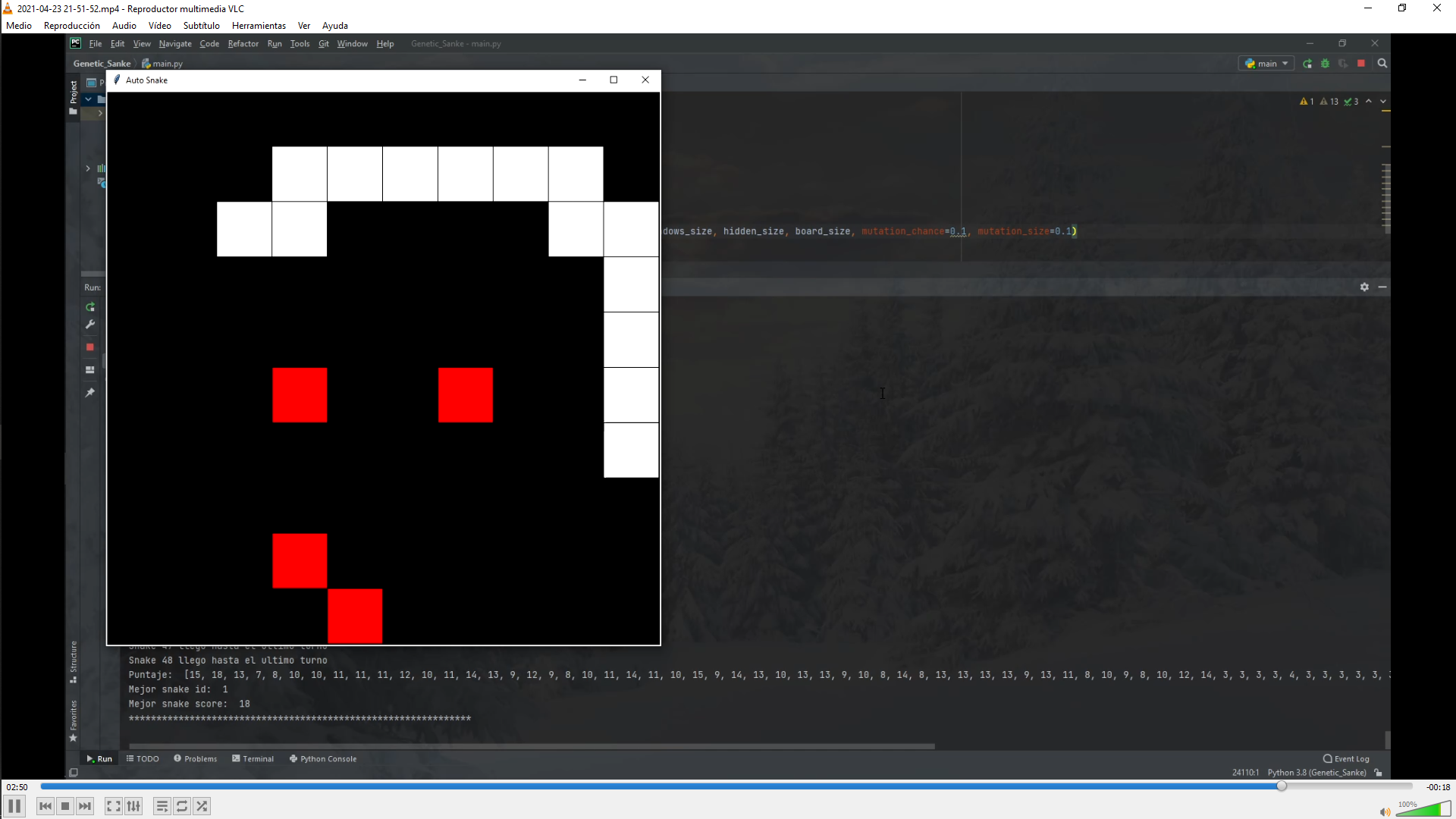
****

Podemos ver que cada generación crea una lista de serpientes, y para la siguiente reproduce a los 25 mejores. Los resultados obtenidos son:



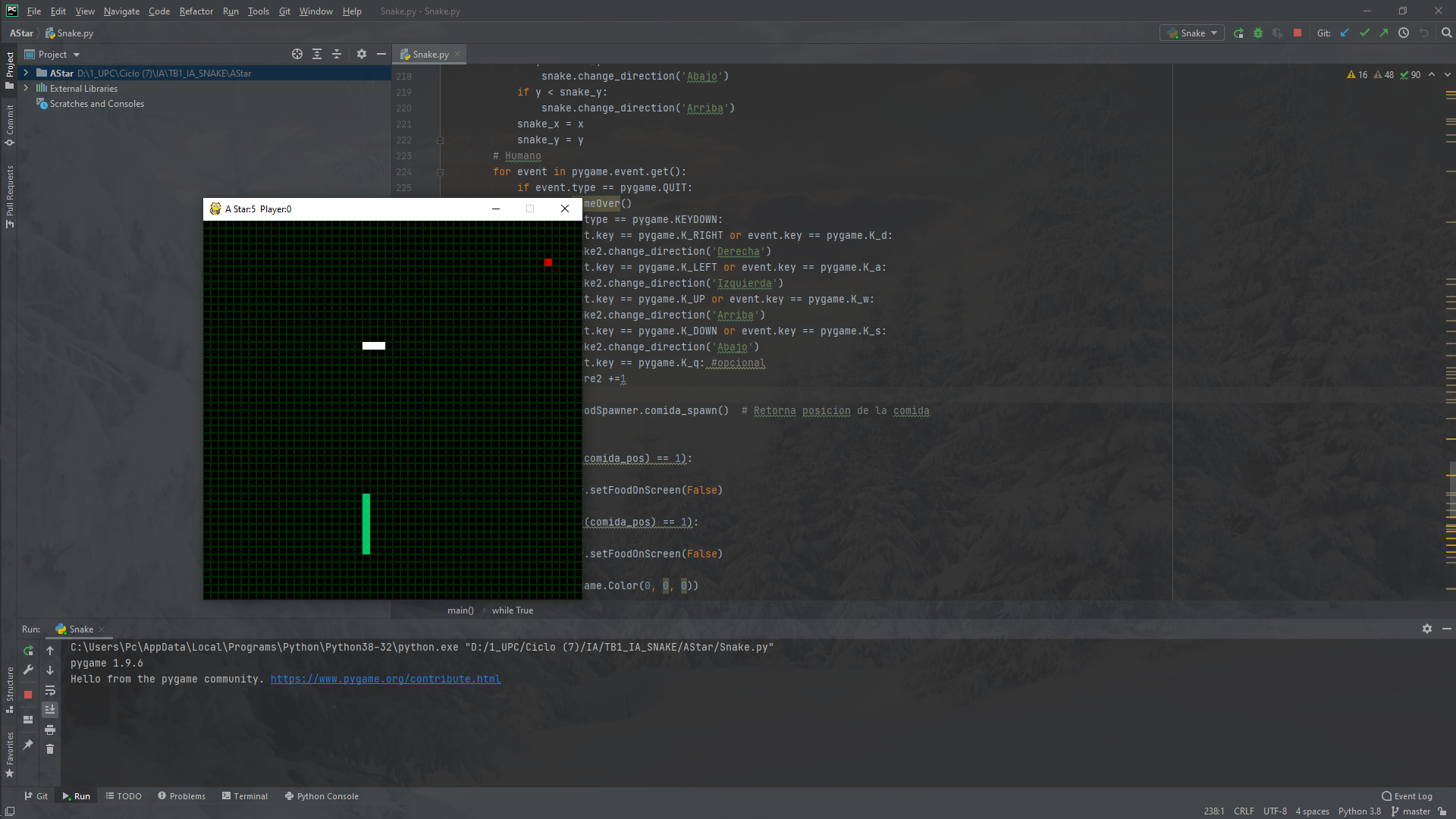
Después de 100 generaciones el rango de la serpiente va desde 6 a 8. Esta no es una buena solución dado que el algoritmo de A\* pudo conseguir este mismo resultado en menos tiempo y en un tablero más grande.

Para continuar hicimos las pruebas con 5000 generaciones, para esta prueba el programa estuvo compilando alrededor de 1 hora y los resultados obtenidos son los siguientes:



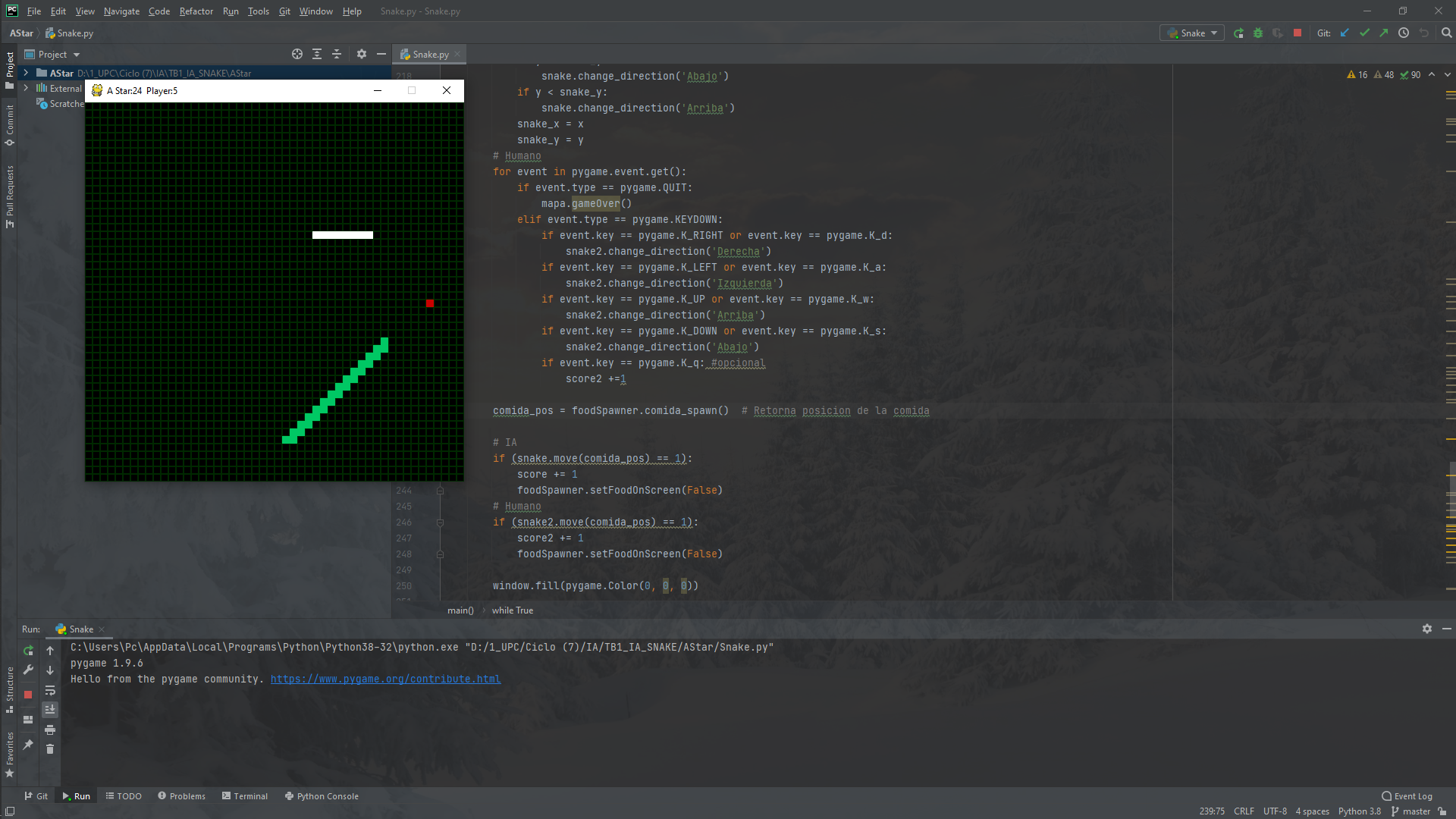
El snake tuvo una mejora con respecto a la anterior prueba, pero tampoco fueron tan buenos resultados dado que el snake solamente llegó a una longitud de 14.

**Algoritmo A\*:**



Para esta implementación contamos con dos snakes. La primera serpiente es la verde y está controlada por la IA, esta serpiente calculará la ruta más corta desde su posición hasta donde encuentre su comida.

La segunda serpiente de color blanco, es controlada por el usuario con las teclas direccionales como si del juego original se tratase. A medida que va pasando el tiempo se nota una gran diferencia entre la IA y el jugador, la más notoria es que la IA nunca falla en encontrar el objetivo y siempre llega con la ruta más corta, además, se siente como si siempre supiese donde aparecerá la siguiente comida dado que nunca está dando vueltas sin hacer nada.

****

Como se puede apreciar, la snake controlada por la IA, casi siempre llegará al objetivo antes que el controlado por el usuario.

**VI. Conclusiones**

Con Algoritmos Genéticos podemos llegar a obtener resultados muy buenos y perfectos, sin embargo, estos resultados tardan mucho tiempo en ser encontrados, por otra parte, el algoritmo A\*, para esta prueba, tuvo resultados más favorables y en menos tiempo. Esto no quiere decir que un algoritmo sea mejor que otro, sino que ambos tienen ventajas y desventajas las cuales no serán igual de buenas en las mismas áreas.

**VII. Referencias bibliográficas**

* Kuri Morales, A., Galaviz Casas, J., & e-libro, Corp. (2002). Algoritmos genéticos. Instituto Politécnico Nacional.
* Marín Morales, Roque Luis, & Palma Méndez, José Tomás. (2008). Inteligencia artificial. McGraw-Hill.