

MANUAL TECNICO

Este documento proporciona una descripción detallada del código y la arquitectura del juego de Totito (Tic-Tac-Toe) con una interfaz gráfica creada con tkinter. También se implementa un árbol AVL para almacenar y visualizar el historial de movimientos del juego.

PASO 1

Requisitos:

- Python 3.x
- Biblioteca tkinter (incluida en la instalación estándar de Python)
- Biblioteca pydot y graphviz para la visualización del árbol AVL (pueden instalarse usando pip install pydot graphviz)

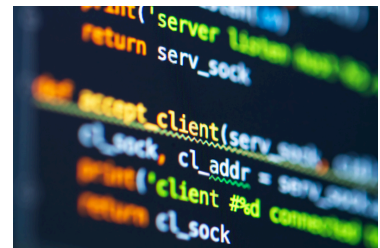


PASO 2

Arquitectura del Código:

Estructura del Código:

- **AVLNode**: Clase que representa un nodo en el árbol AVL.
- **AVLTree**: Clase que implementa un árbol AVL con métodos para insertar y balancear nodos.
- **TicTacToe**: Clase que implementa la lógica del juego y la interfaz de usuario.



PASO 3

Descripción de Funcionalidades:

1. Interfaz de Usuario:

- La interfaz gráfica permite a los usuarios jugar al Totito (Tic-Tac-Toe).
- Los botones en la cuadrícula representan las celdas del tablero del juego.
- Hay etiquetas para mostrar el puntaje, el historial de movimientos y el estado del árbol AVL.

2. Manejo de Eventos:

- Cuando se hace clic en un botón de celda, se actualiza el estado del tablero, se inserta el movimiento en el árbol AVL y se verifica si hay un ganador o un empate.

3. Juego de la Máquina:

- La máquina intenta primero ganar, luego bloquear al jugador, y finalmente hacer un movimiento aleatorio si no puede ganar o bloquear.

4. Árbol AVL:

- El árbol AVL se utiliza para almacenar el historial de movimientos del juego.
- Se muestra en una nueva ventana al finalizar el juego y también se guarda como una imagen.



PASO 4

Reinicio del Juego

- Se puede reiniciar el juego usando el botón "Reiniciar".



PASO 5

Diálogo para Jugar de Nuevo:

- Al finalizar un juego, se pregunta al usuario si quiere jugar de nuevo.