Ejercicio de Laboratorio 6. Minimax y Poda Alfa-Beta

# Ejercicios

1. Implementa tu propia versión del juego del gato (tic tac toe) empleando minimax:
   1. A partir de la implementación anterior, crea una versión que implemente la poda alfa beta.
2. Prueba tu juego con otros usuarios humanos.
3. Prueba tu juego con otro agente inteligente.
4. Diseña y de ser posible implementa una versión de tu juego con conectividad a red. \*

\* no es obligatorio

# Código

## Función minimax (tablero, profundidad, es\_maximizando)



Este código implementa el algoritmo Minimax, una técnica de búsqueda en árboles utilizada para tomar decisiones en juegos de adversarios. Funciona considerando todas las posibles jugadas a partir del estado actual del juego y determinando la mejor jugada posible para el jugador maximizador (generalmente el jugador "X" en el juego del Tic Tac Toe) y la peor jugada posible para el jugador minimizador (generalmente el jugador "O" en el Tic Tac Toe).

* La función minimax evalúa todas las posibles secuencias de movimientos recursivamente.
* Si uno de los jugadores gana, asigna un valor de evaluación alto si es el jugador "X" y un valor bajo si es el jugador "O".
* Si el juego termina en empate, devuelve un valor neutro.
* Para cada posible movimiento, se simula el movimiento y se llama recursivamente a minimax con un nivel de profundidad incrementado.
* Dependiendo de si es el turno del jugador maximizador o minimizador, se maximiza o minimiza el valor de evaluación resultante.
* Finalmente, devuelve el mejor valor de evaluación encontrado para el jugador maximizador o el peor valor para el jugador minimizador.

## Función movimientoComputadora ()



Este código implementa una función movimientoComputadora que utiliza el algoritmo Minimax para determinar el mejor movimiento para la computadora en el juego del Tic Tac Toe.

* Inicializa mejor\_valor con un valor muy bajo (negativo infinito) y mejor\_movimiento como None.
* Itera sobre todas las posiciones del tablero.
* Para cada posición vacía, simula que la computadora coloca una ficha "X" en esa posición y llama al algoritmo minimax para evaluar el tablero resultante.
* Luego, se deshace del movimiento simulado.
* Si el valor evaluado es mayor que el mejor valor encontrado hasta ahora (mejor\_valor), actualiza mejor\_valor con el nuevo valor y guarda las coordenadas de esa posición como mejor\_movimiento.
* Finalmente, devuelve las coordenadas del mejor movimiento encontrado.

# Pruebas

## Humano vs Computadora

Primero ingresamos el nombre del jugador

Texto

Descripción generada automáticamente

Se muestran las instrucciones

Texto

Descripción generada automáticamente

La computadora comenzó, entonces nos toca elegir una casilla que será la 5

Diagrama

Descripción generada automáticamente con confianza media

En el turno de la computadora, intenta ganar ya que no hay suficientes casillas del jugador X para bloquear la jugada, entonces la bloqueamos nosotros

Diagrama

Descripción generada automáticamente

Como ahora si pudiéramos ganar la computadora nos bloquea la jugada y ahora nos vuelve a tocar bloquear su jugada, pero esta nos intentara bloquear eligiendo la casilla 6.

Imagen que contiene Diagrama

Descripción generada automáticamente

Este gato ya se ahogó, será un empate.

Imagen que contiene Gráfico

Descripción generada automáticamente

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

## Computadora vs Computadora

Seleccionar el modo

Texto

Descripción generada automáticamente

Gano la computadora 1, aquí se llena por ticks el tablero

Texto

Descripción generada automáticamente con confianza media

También hay empates

Interfaz de usuario gráfica

Descripción generada automáticamente