TIK TAK TOE 3D

DANIEL STEVEN FLORIDO SAEZ, JUAN SEBASTIAN GALEANO GONZALEZ

Resumen. En este documento se desarrollan distintos algoritmos con el fin de reemplazar un jugador en el juego 3D Tic Tac Toe, una versión en tres dimensiones del Tic Tac Toe convencional.

Parte 1. Análisis y Diseño del problema

1. Análisis

1.1. Tic Tac Toe. Tic-Tac-Toe, comúnmente conocido como Tres en Raya o Gato, es un juego de mesa tradicional diseñado para dos jugadores. El objetivo del juego es colocar estratégicamente sus respectivos símbolos en un tablero de 3x3, con el fin de crear una línea de tres símbolos en disposición horizontal, vertical o diagonal. El primer jugador generalmente adopta el símbolo "X", mientras que el segundo jugador utiliza el símbolo ."o". Si todas las casillas del tablero están ocupadas y ninguno de los jugadores ha logrado formar una línea ganadora, el juego termina en empate.

Debido a sus reglas sencillas y su baja complejidad, el Tic-Tac-Toe ha ganado popularidad como un juego casual o infantil, al tiempo que se utiliza como una herramienta educativa para enseñar conceptos fundamentales de estrategia y lógica.

1.2. Tic Tac Toe 3D. El Tic-Tac-Toe 3D es una variante tridimensional del juego clásico Tic-Tac-Toe, en la cual se juega en un tablero de 3x3x3 casillas. Al igual que en el juego convencional, el objetivo es que los jugadores coloquen sus símbolos (generalmente "Xτ ."o") en una línea de tres en cualquier dirección: horizontal, vertical, diagonal o incluso en las diagonales a través de las tres capas del tablero. Este desafío adicional de tener una dimensión extra agrega complejidad estratégica al juego y amplía las posibilidades de movimiento para los jugadores.

El Tic-Tac-Toe 3D ofrece una experiencia de juego más profunda y requiere un análisis estratégico más cuidadoso para lograr la victoria. La interacción entre las diferentes capas del tablero y las múltiples direcciones en las que se pueden formar líneas ganadoras brinda un desafío adicional y estimula el pensamiento tridimensional. Esta variante tridimensional del Tic-Tac-Toe puede proporcionar una experiencia más cautivadora y desafiante para aquellos que buscan un nivel adicional de complejidad en el juego.

Date: 2 de junio de 2023.

A continuación se definen formalmente los conjuntos para la solución del problema que plantea el tic tac toe tridimensional:

- $n \leftarrow$ tamaño de cada dimensión del tablero en el juego Tic-Tac-Toe 3D. Representa el número de casillas en cada dimensión del tablero.
- $T \leftarrow$ conjunto de casillas que conforman el tablero. El tamaño del tablero resulta en un total de n x n x n casillas en el tablero completo.

2. Diseño

A continuación se presentan las condiciones de entrada para el correcto funcionamiento de la solución y su respectiva salida:

Definición. Entradas:

1. $T \leftarrow$ Conjunto de casillas del tablero que conforman su estado.

Definición. Salidas:

1. $X \leftarrow$ coordenada en el eje x para hacer click en el tablero.
2. $Y \leftarrow$ coordenada en el eje y para hacer click en el tablero.
3. $Z \leftarrow$ coordenada en el eje z para hacer click en el tablero.

Parte 2. Algoritmos

Este algoritmo es una implementación de jugador que elige coordenadas en x, y y z de manera totalmente aleatoria. No obstante, este revisa el estado del tablero antes de dar click para no jugar en una coordenada que ya se encuentra ocupada.

Inicialmente se configuró para que generara 3 números aleatorios, pero esta idea fue descartada debido a que al ser pocas opciones por numero, el tiempo que demoraba eligiendo jugada el algoritmo aleatorio crecía. Por esta razón, se decidió dotarlo con un estado del tablero para que pudiera tomar la decisión de manera mas rápida, generando un solo numero aleatorio que se compara con una posición en el arreglo de casillas. A continuación se muestra la implementación del algoritmo:

Algorithm 1 RandomTiktaktoePlayer::play

procedure PLAY(x, y, z, boxes)
    randomNumber ← random number between 0 and size^3
    do
        randomNumber ← random number between 0 and size^3
    while boxes[randomNumber] $=' X'$ or boxes[randomNumber] $=' O'$
    convert randomNumber to (x, y, z) using _idx function
end procedure

Algorithm 2 Función _idx

function _IDX(x, y, z, number)
    $n \leftarrow$ size
    $z \leftarrow$ number $\div (n^2)$
    number ← number mód (n^2)
    $y \leftarrow$ number $\div n$
    number ← number mód n
    $x \leftarrow$ number
end function