

Proyecto#2: Torneo Ta-Te-Ti 3D

Introducción a la Inteligencia Artificial

TRABAJO INDIVIDUAL

Fecha Entrega: 29 de Octubre 23:59hs

Objetivo

Este proyecto tiene como objetivo escribir un algoritmo Minimax con podado Alfa-Beta y heurísticas necesarias para hacer un juego ganador.

Instrucciones

Para este proyecto se utilizará Python 3.6 para corregir el problema

Primero familiarízate con las funciones (de Python) que hemos incluido en los archivos. El archivo más importante que NO HACE FALTA modificar es **competencia.py** que hace que 2 o más jugadores jueguen Ta Te Ti 3D en un torneo que determina quién es el ganador.

Fijate en **jugador_libre.py** y **jugador_aleatorio.py**. Estos dos archivos (y sus copias de demostración) usan una estrategia muy simple para jugar. Tu versión llamado **jugador_tunombre.py** (donde reemplazas "tunombre" por tu nombre) competirá con estos para ver quién gana.

Tu misión es simple. Implementa:

1. Minimax con Podado Alfa-beta
2. Limitación de profundidad en el algoritmo (Nota: tu algoritmo no debe exceder el tiempo permitido o será descalificado)
3. Una heurística de evaluación para determinar si la posición favorece o desfavorece al jugador
- 4.

Nota: toda implementación debe ser tuya. No bajar código.

IMPORTANTE: no cambiar la firma de método **jugar()** en el archivo `jugar_*.py` porque este proyecto se corregirá de manera automatizada. Si no se ejecuta.. no tendrás puntos.

La Competencia / Evaluación

La calificación será en base a cuántos de los puntos anteriores se lograron. Según lo siguiente:

20pts Minimax correcto y Alfa-beta correcto
20pts Limitación de profundidad y tiempo
10pts Optimizaciones
30pts Heurística efectiva para el juego
20pts Según competencia

La alumna o alumno que pueda resolver los problemas dados con el MENOR total número de pasos ganará puntaje extra según la siguiente tabla:

Primer lugar: 20 pts
Segundo lugar: 18 pes
Tercer lugar: 15 pts

Solución válida no ganadora: 10 pts

Haber entregado el proyecto: 5 pts

No entregado: 0 pts