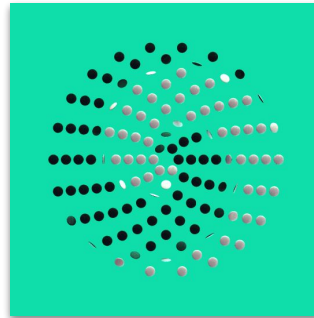


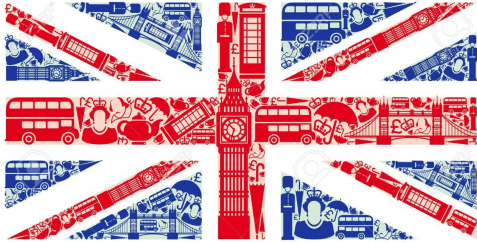
OTHELLO



MiniMax

Samantha Arbuola
Inteligencia Artificial
Semestre I 2018 - TEC

HISTORIA



Siglo XIX ~ 1880: Lewis Waterman y John W. Mollett



Goro Hasegawa :1971



1977: se juega un
Campeonato Mundial de
Othello

OTHELLO REVERSI YANG

Duración: 10-60 min

Dificultad: Baja



Ver Partida

— — —

ELEMENTOS PARA EL JUEGO

Jugadores

2

Ficha

64 fichas

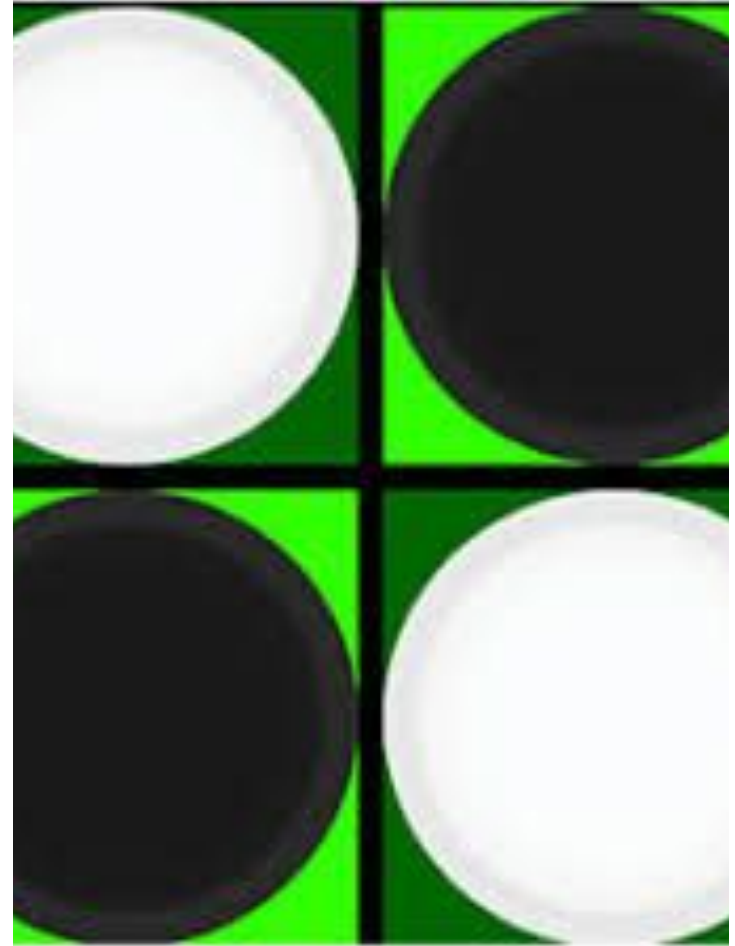
2 colores, 1 por cara
se comparten

Dimensión del Tablero

8x8

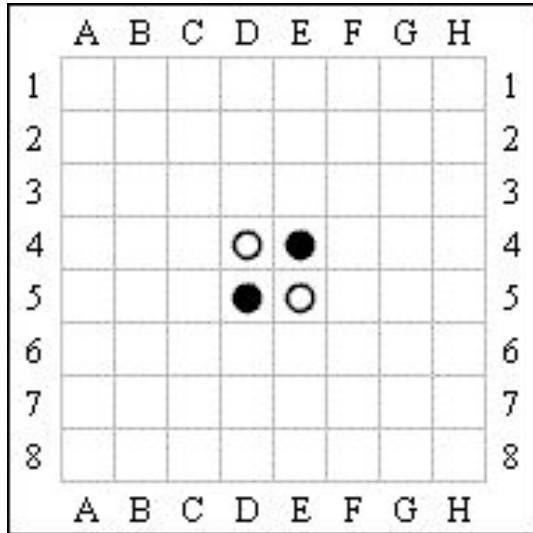
Verticales: Números 1 ~ 8

Horizontales: Letras A ~ H



¿CÓMO SE JUEGA?

Inicio



Fin

La partida finaliza cuando
ningún jugador puede mover

¿CÓMO SE JUEGA?

Reglas

- ★ Empieza quien tiene ficha negra
- ★ 1 movimiento por turno, a menos que no pueda hacerlo

Movimiento

El movimiento consiste en colocar una ficha de forma que flanquee una o varias fichas del color contrario y voltear esas fichas para que pasen a mostrar el propio color.

¿CÓMO SE JUEGA?

Flanquear Ficha

- ★ Deben formar una línea continua recta (diagonal u ortogonal) de fichas del mismo color entre dos fichas del color contrario

Objetivo

Dejar al oponente sin movimientos

¿CÓMO SE JUEGA?

Ganador

Quien tenga más fichas en el



SOLUCIÓN DE SOFTWARE



¿CÓMO SE PROGRAMA?

Aplicar un movimiento al tablero y hacer una copia del tablero en cada llamada recursiva a minimax.

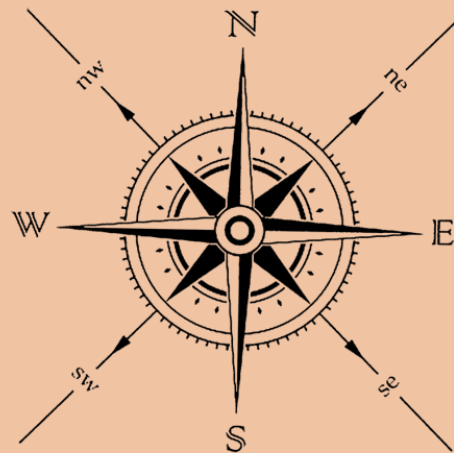
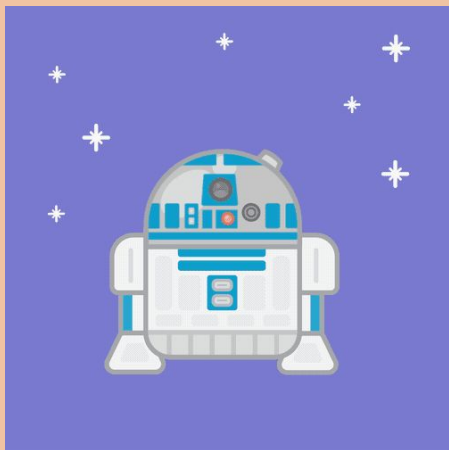
`copy.deepcopy(board)`

Esta es una forma eficiente de copiar una matriz 2D a otra, siempre que la matriz de datos esté almacenada contiguamente en la memoria. Esto es mucho más rápido que los bucles anidados para copiar cada elemento

```
(boardTemp, totctr) = MakeMove(copy.deepcopy(pBoard), x, y, pPlayer)
```

¿CÓMO SE PROGRAMA?

La función MiniMax para llamar al agente AI



Evalúa las 8 direcciones para tomar la mejor posición; tomando en cuenta las fichas que le benefician, para hacer un movimiento

¿CÓMO SE PROGRAMA?

Necesitamos el giro original

Por lo que sabemos si se trata de un
movimiento de minimización o
maximización

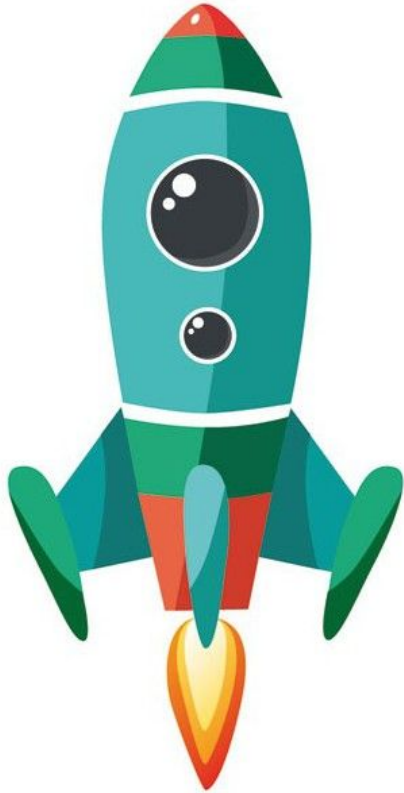
Con un booleano definimos:



True si maximizamos



False si minimizamos



POSIBILIDADES DE MEJORA

Cambios en la
profundidad de la
búsqueda

Ponderar las
esquinas

Implementación
de Búsqueda
Alfa-Beta

REFERENCIAS

- Code.activestate.com. (2018). *Reversi Othello « Python recipes « ActiveState Code*. [online] Available at: <http://code.activestate.com/recipes/580698-reversi-othello/> [Accessed 2 Apr. 2018].
- Cse.uaa.alaska.edu. (2018). [online] Available at: http://www.cse.uaa.alaska.edu/~afkjm/csce211/handouts/ot_hello-minimax.pdf [Accessed 2 Apr. 2018].
- Inventwithpython.com. (2018). [online] Available at: <http://inventwithpython.com/chapter15.html> [Accessed 2 Apr. 2018].
- JK Webdesign, +. (2018). *World Othello Federation*. [online] Worldothello.org. Available at: <http://www.worldothello.org/> [Accessed 2 Apr. 2018].
- Web.archive.org. (2018). *Wayback Machine*. [online] Available at: <https://web.archive.org/web/20110512210955/http://othellogateway.com/rose/book.pdf> [Accessed 2 Apr. 2018].
- YouTube. (2018). *Othello - NES Gameplay*. [online] Available at: https://www.youtube.com/watch?v=DG2xFqIBZ_0 [Accessed 2 Apr. 2018].