

TETRIS



Eduardo Cruz, nº 93088

Tiago Bastos, nº 97590

ORGANIZAÇÃO DO CÓDIGO

STUDENT.PY

AGENT.PY

- * Comunicação com o servidor
- * Inicialização dos níveis e do agente

- * Execução do agente
- * Calcular possiveis posições dada uma peça e um movimento.
- * Calcular heuristicas e escolher melhor movimento.

CALCULO HEURÍSTICA

HEURÍSTICA - FUNÇÃO

-0.510066*aggr + 0.760666*comp – 0.35663*holes– 0.184483*bump

TOTAL_HEIGHT (AGGR)

 Calcula a soma das alturas de todas as colunas consoante o estado do jogo.

CALCULATE_HOLES(HOLES)

 Calcula o numero de buracos existentes consoante o estado do jogo.

CALCULATE_COMPLETED_LINE S(COMP)

 Calcula o número de linhas completas possíveis consoante o estado do jogo.

CALCULATE_BUMPINESS (BUMP)

 Calcula a soma da variação da altura das colunas consoante o estado do jogo.

FUNÇÕES

Heuristic

Calcular heurística dado funções anteriores (slide ant.)

Possible positions

Calcular qual o melhor movimento possivel para uma dada peça

Next_key

Identificação da peça que entra no jogo

Calculate crust

Calcular a superficie (crosta) do jogo

Calculate total height

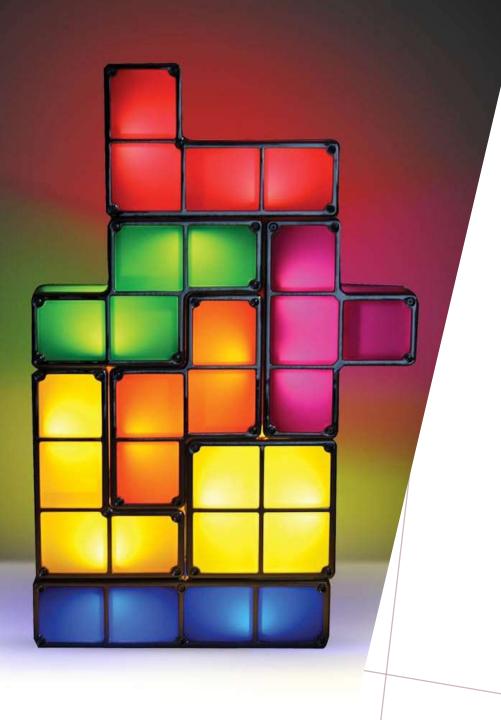
Calcular soma das alturas de cada coluna do tabuleiro

Possible moves

Todos os movimentos possiveis para cada peça

Final state piece

Calcula a posição final da peça dado uma lista de "movs"



CONCLUSÃO

O grupo considerou que existiam possíveis otimizações que poderiam ser implementadas:

- * Ver próximas peças no agente (mas diminuía tempo de resposta para o servidor)
- * Otimização de algumas funções para diminuir o tempo de resposta para o servidor.

Comentário final:

Foi um projeto/trabalho bastante interessante e contribuiu positivamente para o conhecimento na área da inteligência artificial

Fonte consultada para desenvolvimento do código:

https://codemyroad.wordpress.com/2013/04/14/t etris-ai-the-near-perfect-player/