

Trabalho sobre Métodos de busca (2017-2)

O propósito do trabalho é implementar (na linguagem de sua preferência) o algoritmo de busca adversária **MiniMax com podas α e β** . A implementação será testada através do jogo **5 em linha**, com tabuleiro tamanho 15x15.

Portanto, implemente o jogo **5 em linha** (<http://gomoku.yjyao.com/>) de forma a ser computador x humano (a interface pode ser bem simples, inclusive modo texto – deve permitir escolher quem inicia o jogo – quem quiser reutilizar alguma interface pronta [1] ou ainda reutilizar a parte de verificação de vitória de alguma implementação já construída ok, porém é **obrigatório** citar a fonte no relatório e no código-fonte, caso não for referenciado será considerado plágio). A cada jogada do computador, apresente a quantidade de iterações que foi necessária para a jogada.

Juntamente com a implementação deverá ser entregue um **mini-relatório** explicando o papel de cada classe e os métodos **principais**, qual a estrutura de dados adotada, como é feito o cálculo do utilidade e da heurística, demais otimizações (ex.: ordenação de estados), problemas encontrados (se houver) e limitações da implementação. A **avaliação da implementação** considera especialmente a forma de implementar a busca e o cálculo da heurística/utilidade. Para receber nota máxima na implementação é necessário utilizar uma estrutura de dados e de busca adequada, uma heurística/utilidade compatível com o algoritmo implementada matematicamente e o bom uso de alguma(s) otimizações (as mesmas não podem substituir o minimax).

A implementação enviada será executada e testada. Além da execução correta (fazer as jogadas o mais certas possíveis) será considerada a organização do código, explicação das estratégias, possibilidades, decisões e problemas encontrados (a avaliação considera todo o trabalho realizado, não apenas uma saída correta). Portanto **o relatório também será considerado durante a avaliação**.

Etapas do desenvolvimento

A avaliação do trabalho será realizada em duas etapas. Na primeira etapa será avaliada a implementação da estrutura de dados (modelo dos estados e do jogo – funcionamento do jogo humano x humano), definição matemática das funções de utilidade e heurística e a implementação parcial da função heurística. Quanto a definição matemática da utilidade e heurística, será avaliado um **estudo** das possibilidades que o grupo pretende adotar e o resultado do estudo em uma função matemática. Quanto a implementação parcial, espera-se que, ao menos, esteja pronta a detecção de fim de jogo e alguma detecção de sequencias de quatro peças.

→ o que entregar:

- mini-relatório com (no mínimo):
 - a definição matemática da utilidade, heurística(s)
 - estratégias e otimizações (quando aplicável - opcional),
 - descrição sobre como é feita a detecção de fim de jogo e quando ela é aplicada
 - descrição sobre como é feita a detecção de sequencias de quatro peças
- código-fonte

Na segunda etapa do trabalho será avaliada a implementação do algoritmo e o relatório final. Será considerada a organização do código, a descrição e a implementação das estratégias, possibilidades (o que poderia ser feito mas não deu tempo, como melhorar um problema que está ocorrendo, etc.), decisões e problemas encontrados (a avaliação considera todo o trabalho realizado, não apenas uma saída correta) e domínio do trabalho ao responder os questionamentos sobre o trabalho.

→ o que entregar: mini-relatório agora completo (explicação das decisões de projeto, limitações, descrições dos principais métodos, utilidade, heurística, etc.) e código-fonte

Prazos para entrega:

pt 1. 22/08/2017

pt 2. 01/09/2017

A 1a etapa do trabalho vale 30% da nota final do trabalho e a 2a etapa 70%.

Se for detectado plágio de qualquer forma, **todos** os envolvidos receberão nota 0 e o fato será aviso à coordenação de curso.

[1] <http://cs.gettysburg.edu/~tneller/cs111/gomoku/>