

Relatório Trabalho Inteligência Artificial AV1



Aluno: Diego Castelo (2214642)
Professor: Vasco Furtado

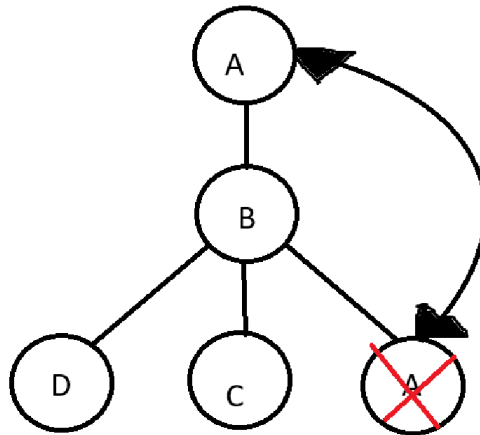
Descrição das heurísticas escolhidas

Heurística 1: Quantidade de peças na posição correta

O próximo estado escolhido será o que tem mais peças nas posições corretas de acordo com o estado final.

Heurística 2: O próximo estado não pode ser o mesmo que o anterior.

Essa heurística foi escolhida para que o algoritmo não entre em um loop facilmente quando o próximo melhor estado for igual ao anterior.



Heurística 3: Distância de Manhattan

É a soma das distâncias horizontais e verticais entre dois pontos se movendo somente na vertical e horizontal.

Comparativo dos resultados obtidos

Buscas Cegas:

8	1	3
2	b	4
7	6	5

1	2	3
8	b	4
7	6	5

Algoritmo	Memória	Tempo	Movimentos
Busca Largura	112.3125 MB	0.0022144317626953125 ms	3
Busca Profundidade	infinito	infinito	infinito
Busca Iterativa em Profundidade	112.54296875 MB	0.004470109939575195 ms	3

Buscas Heurísticas:

8	1	2
6	b	3
7	5	4

1	2	3
8	b	4
7	6	5

Algoritmo	Memória	Tempo	Movimentos
Guloso com Heurísticas	114.23828125	2.506218671798706	258
Hill Climb	113.8046875 MB	0.009624481201171875 s	11

Buscas Cegas:

1	2	3	4
12	15	13	5
10	11	14	7
b	9	6	8

1	2	3	4
12	13	14	5
11	b	15	6
10	9	8	7

Algoritmo	Memória	Tempo	Movimentos
Busca Largura	127.132415 MB	0.27145934104919434	10
Busca Profundidade	infinito	infinito	infinito
Busca Iterativa em Profundidade	127.171875 MB	0.276378870010376	10

Buscas Heurísticas:

1	2	3	4
12	15	13	5
10	11	14	7
b	9	6	8

1	2	3	4
12	13	14	5
11	b	15	6
10	9	8	7

Algoritmo	Memória	Tempo	Movimentos
Guloso com Heurísticas	tende ao infinito	tende ao infinito	tende ao infinito
Hill Climb	tende ao infinito	tende ao infinito	tende ao infinito

Conclusão:

As buscas cegas tendem a manter um padrão no consumo de memória e de tempo nos casos apresentados. Vimos que usar a busca por profundidade nesse caso é bem ruim porque tende ao infinito (testando várias vezes o código se tornava bem comum), por isso foi levantado que o uso desse tipo de busca para esse problema é equivocado.

Por conta de terem complexidades de tempo e espaço idênticas, pudemos observar que a busca por largura e a busca iterativa por profundidade desempenharam quase idênticas em ambos os casos das 8 peças e das 15 peças.

No caso das buscas por heurísticas pudemos ver uma disparidade grande entre os dois casos. No caso das 8 peças acredito que o resultado tenha sido satisfatório já que as heurísticas do algoritmo guloso não são tão elaboradas, já a do Hill Climbing é um pouco mais performática. Já nos casos de 15 peças o resultado foi falho, o principal motivo de que está tendendo ao infinito foi a má construção dessas heurísticas resultando na não solução de casos que deveriam ser “simples”, o resultado deveria ter sido parecido com o das 8 peças.

Então podemos concluir que o resultado da implementação dos algoritmos foi um sucesso para as buscas cegas, mas para a busca com heurística foi falha em casos mais complexos como o das 15 peças.