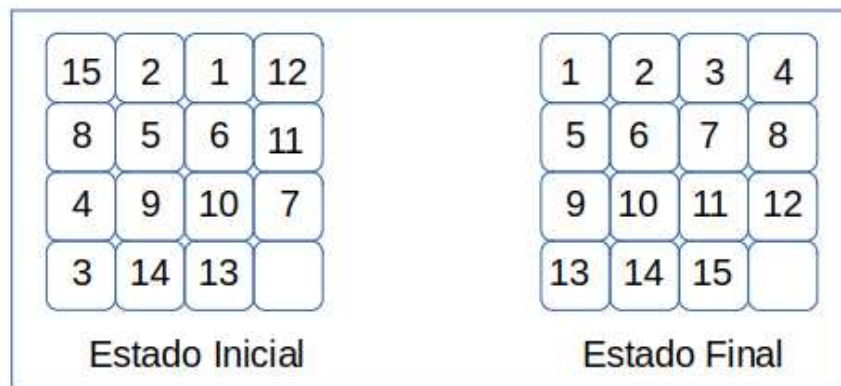


Inteligência Artificial

Tarefa - Busca Informada

15-puzzle

1. Nesta tarefa você deve implementar em Prolog o algoritmo A^* para resolver o Jogo do 15 (https://pt.wikipedia.org/wiki/Jogo_do_15)



Além do arquivo com o programa, você deve entregar um relatório com as seguintes informações:

Modelagem do Problema

- Descreva como o tabuleiro inicial deve ser representado no seu programa e apresente o tabuleiro da figura acima na representação proposta. Considere que para testar o programa com outras configurações iniciais, bastará substituir no seu programa a cláusula (ou cláusulas) Prolog que definem o tabuleiro inicial. O tabuleiro inicial **não deve** ser passado como parâmetro da consulta.
- Apresente e explique cada uma das regras do programa que definem os movimentos válidos no tabuleiro.
- Apresente e explique o teste que é feito na configuração inicial e que permite identificar se existe ou não um caminho até o estado final. Mostre como isso é feito no seu programa.

Heurísticas

- Utilize as heurísticas:
 - h_1 : número de peças fora do lugar
 - h_2 : distância Manhattan
- A escolha da heurística a ser utilizada deve ser feita através de um dos parâmetros da consulta.

-
- Apresente e explique como é feita a avaliação de uma configuração do tabuleiro.

Apresentação do resultado

- A saída do seu programa deve apresentar:
 - No caso de haver um caminho do estado inicial ao estado final
 - * A sequência de movimentos que permite sair do estado inicial e chegar no estado final.
 - * O número total de nós **gerados** durante a busca.
 - No caso de não haver um caminho do estado inicial ao estado final, uma mensagem informando que não existe o caminho.

Análise dos Resultados

- Você deve avaliar o seu programa, aplicando as duas heurísticas definidas em 3 configurações iniciais distintas. O tamanho do caminho da solução ótima encontrada deve ser diferente em cada um dos casos.
- compare os resultados obtidos em cada caso com relação a quantidade de configurações geradas durante a busca.

IMPORTANTE: o programa deve ser feito de maneira que cada tarefa realizada (por exemplo, determinação se existe ou não um caminho ligando o estado inicial ao final; geração de estados sucessores do estado corrente; avaliação de uma dada configuração) possa ser testado de forma independente.