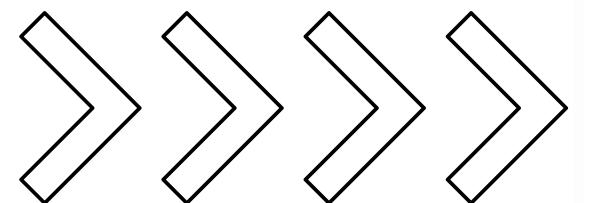


MÉTODOS DE BUSCA

INTELIGÊNCIA ARTIFICIAL



Alunos: Anthony e Vicenzo

Índice

03 Introdução

04 Para que serve?

05 Técnicas

06 Categorias

7/8 Problema do labirinto

9/10 Métodos implementados

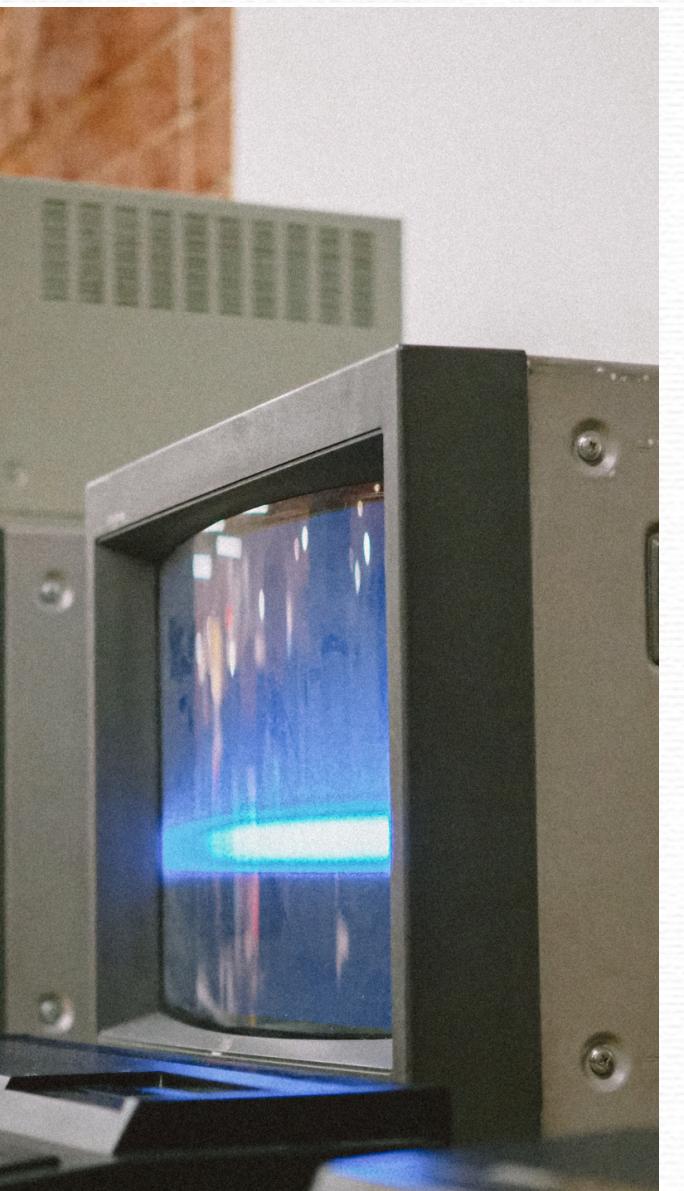
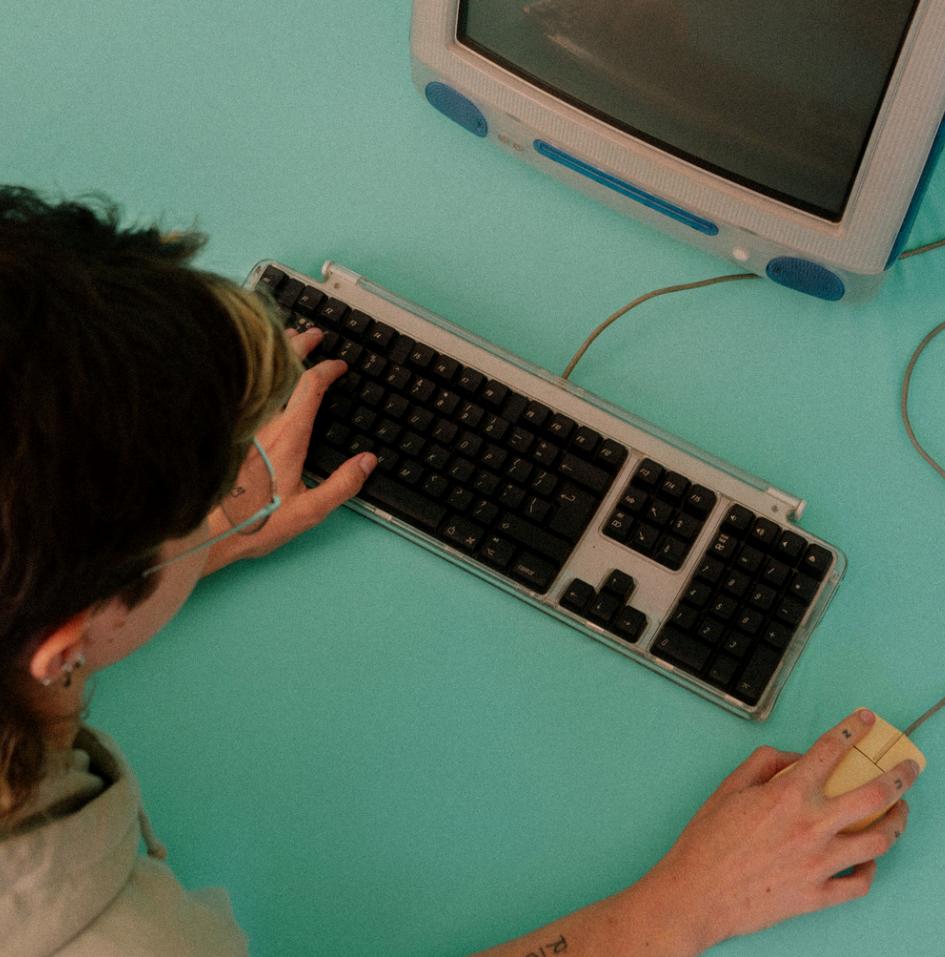
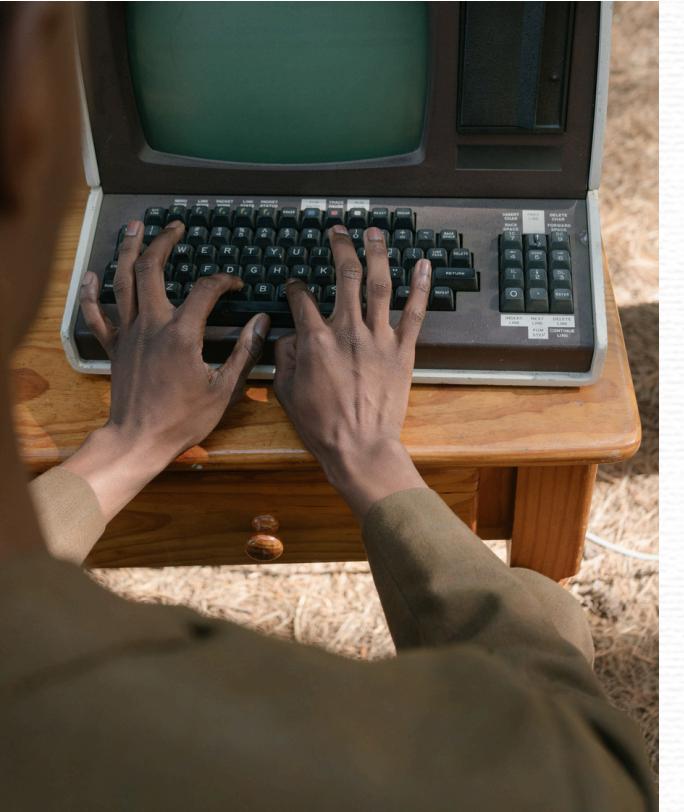
11 Análise

12 Conclusão

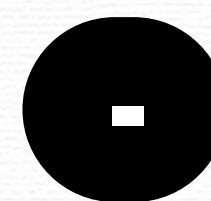
Introdução

IA/Métodos de busca

- Os métodos de busca são algoritmos utilizados para encontrar um estado ou caminho em diversas possibilidades.
- São comumente usados para resolver problemas como navegação, jogos e inteligência artificial.



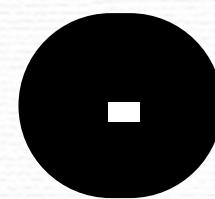
Para que servem?



São essenciais para diversas aplicações como:

- Inteligência Artificial (ex: agentes autônomos como assistentes virtuais e em reconhecimento de padrões)
- Jogos (ex: busca de movimentos em Xadrez)
- Algoritmos de roteamento (ex: GPS e redes de computadores)
- Sistemas de recomendação (ex: buscas de recomendação para o usuário, em grandes bases de dados)

IA/Técnicas para resolução de problemas



Que tipos de problemas?

- (A) diagnóstico ou reconhecimento de padrões
 - tomada de decisões
 - área medica
 - processamento de imagens
- (B) definição de passos até o estado final (empacotamento)
 - área de jogos
 - linha de montagem

Categorias

Busca Cega (não usa heurística)

- Busca em Largura (BFS): Explora todos os nós do mesmo nível antes de avançar.
- Busca em Profundidade (DFS): Explora um caminho até o final antes de retroceder.

Busca Heurística (usa heurística)

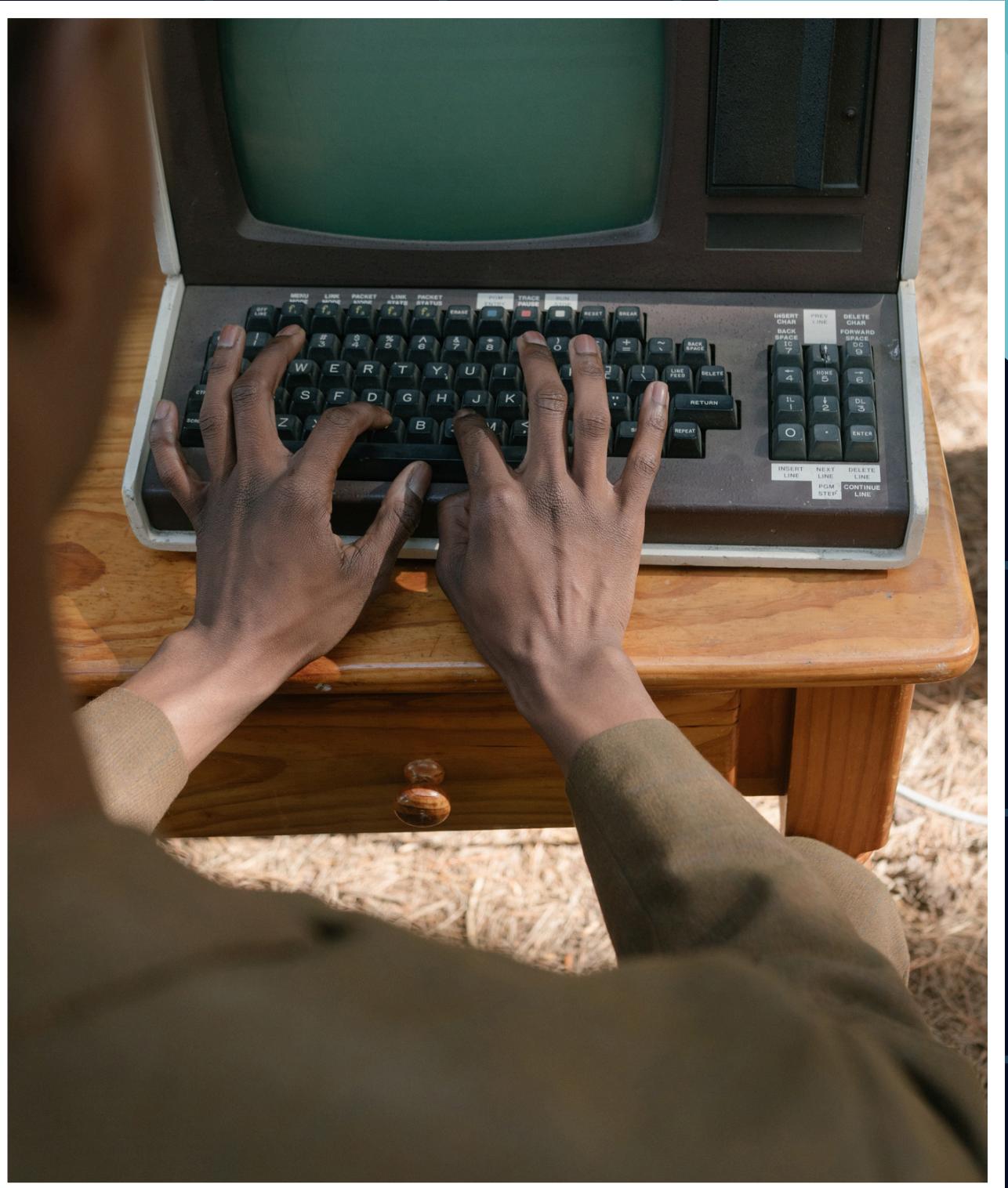
- Busca Gulosa: Escolhe sempre o próximo passo que parece mais promissor.
- Algoritmo A*: Usa uma combinação de custo e heurística para encontrar o caminho mais eficiente.

Explicação do Problema do Labirinto

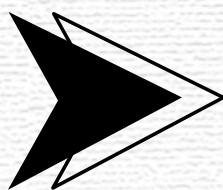
O problema consiste em um labirinto de tamanho NxN, gerado pelo usuário, contendo X obstáculos distribuídos aleatoriamente.

Há uma saída e duas entradas, onde cada entrada utilizará um método de busca diferente para encontrar um caminho até a saída.

A ideia é analisar o “desempenho” dos métodos escolhidos em tempo de execução e eficácia do caminho gerado.

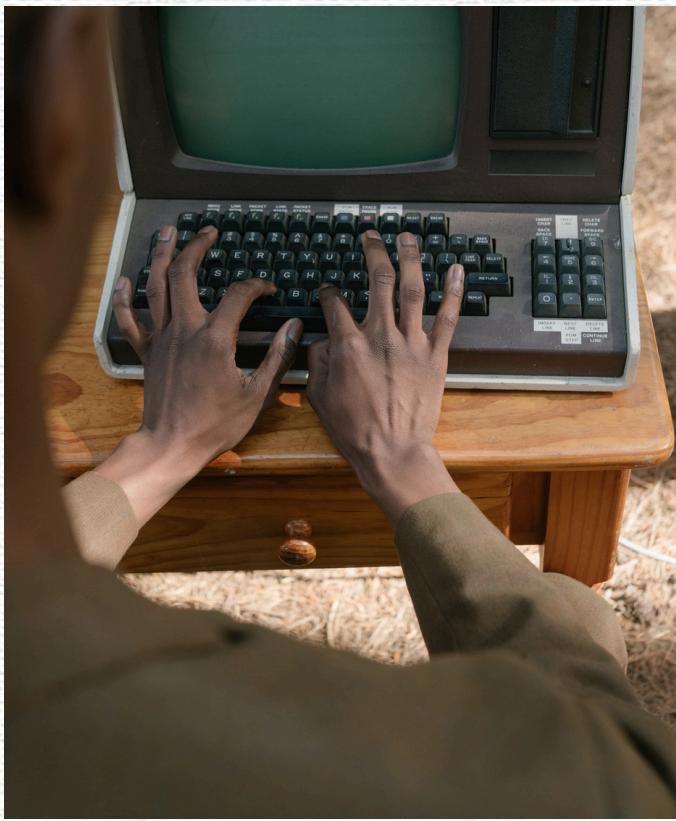


Representação do Labirinto



O labirinto é uma matriz NxN onde:

- Células vazias representam caminhos livres.
- Células marcadas representam obstáculos.
- Há uma célula especial representando a saída.
- Duas células representam os pontos.



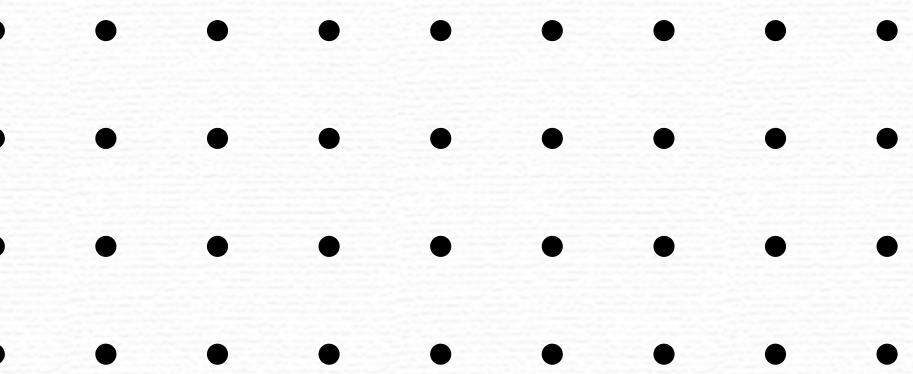
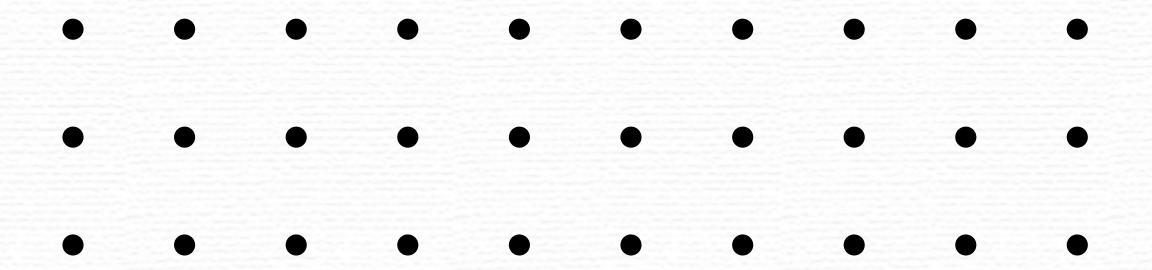
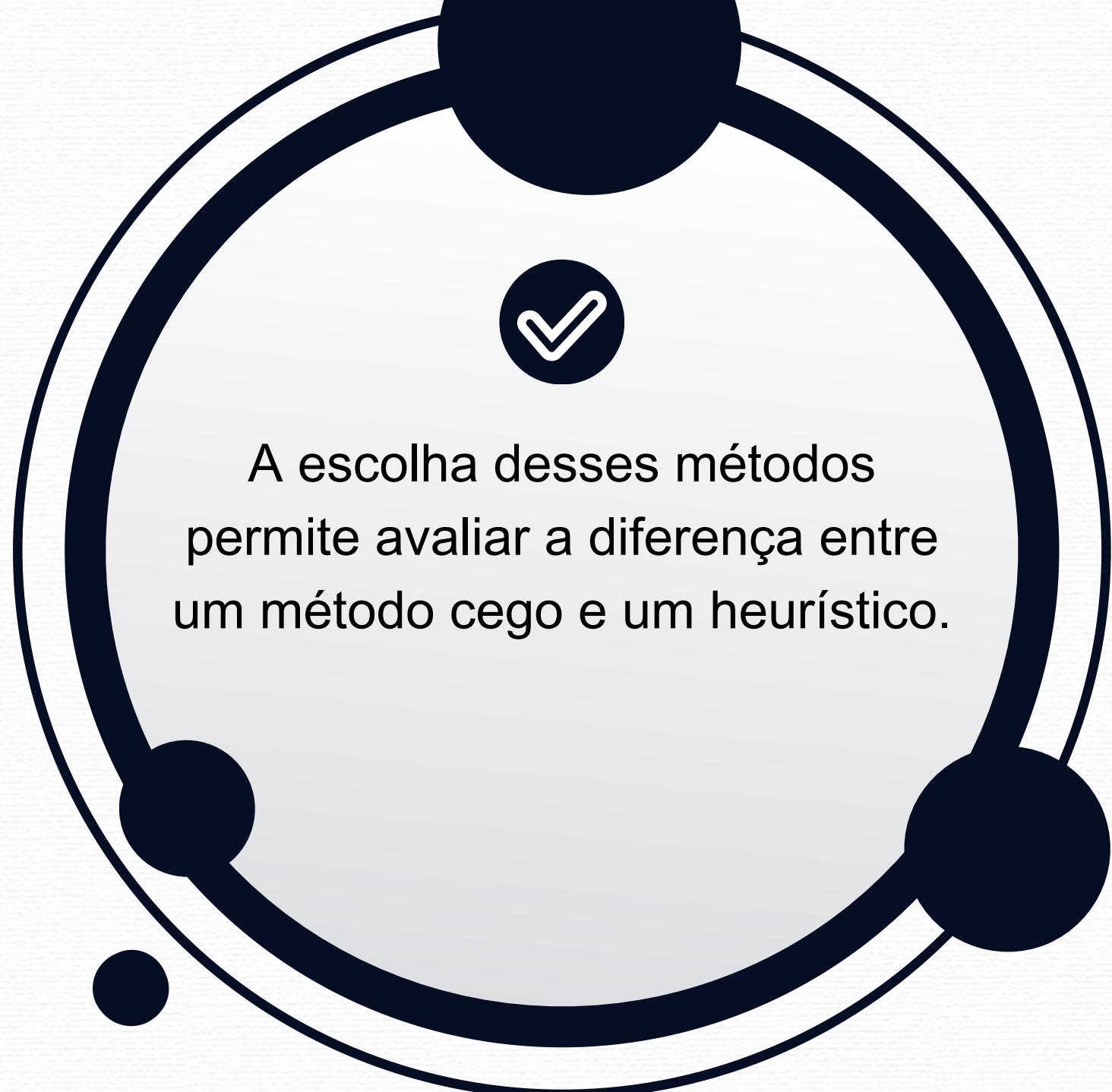
Métodos de busca escolhidos

Para cada entrada do labirinto, utilizaremos um método de busca diferente.
Os métodos escolhidos são:

Entrada 1: Busca em Largura (BFS) – Garante o menor número de passos.

Entrada 2: Algoritmo A* – Utiliza uma heurística para encontrar o caminho mais eficiente.

A escolha desses métodos permite avaliar a diferença entre um método cego e um heurístico.



Implementação

O código foi baseado no pacote disponibilizado pelo professor Jomi Hübner.

O labirinto é gerado aleatoriamente com base nos parâmetros do usuário.

Cada entrada inicia uma busca até encontrar a saída.

A solução de cada método é desenhada sobre a matriz do labirinto.

Comparação

Os métodos foram avaliados de acordo com:

- Tempo de execução: Qual foi mais rápido?
- Número de passos: Qual percorreu menos células até a saída?
- Eficiência: Qual método encontrou o caminho mais curto?

Resultados esperados:

- BFS pode ser mais lento, mas sempre encontra o caminho ótimo.
- *A* pode ser mais rápido e eficiente se a heurística for bem definida.

Conclusão

O uso de diferentes métodos de busca no mesmo problema permite entender seus pontos fortes e fracos.

Métodos heurísticos como *A* tendem a ser mais eficientes, enquanto métodos cegos garantem a melhor solução, mas podem ser mais custosos.

Melhorias futuras podem incluir novas heurísticas e otimizações na busca.