

 <p>INSTITUTO FEDERAL ESPIRITO SANTO Campus Serra</p>	<p style="text-align: center;">IFES</p> <p style="text-align: center;">Inteligência Artificial</p> <p style="text-align: center;">Exercício de Programação 1: Solução de Problemas por Busca</p>	<p style="text-align: center;">Nota</p>
--	--	---

Professor: Sérgio Nery Simões

Data: 05/10/2023

Nome: _____

Turma: _____

Informações:

- O trabalho deve ser submetido até a data limite na atividade do AVA.
- Deve ser submetido um arquivo compactado contendo o código em Python dos programas desenvolvidos e o relatório em formato pdf.
- É permitido que os alunos conversem e discutam suas soluções, mas os códigos-fonte devem ser produzidos individualmente. Se forem identificados casos de cola, serão abertos processos que podem culminar no desligamento do aluno.

Descrição do Trabalho

O trabalho consiste em implementar e comparar o desempenho dos algoritmos de busca: **(i) Depth-first search**, **(ii) Uniform cost search** e **(iii) A* search**. Para ajudá-los a começar e prover um ambiente para testes dos algoritmos, é fornecida uma implementação do algoritmo *breadth-first search* para busca de caminho em um labirinto em anexo à esta especificação. Você deve utilizar a estrutura de dados fornecida por essa implementação para implementar os demais algoritmos solicitados neste trabalho.

Para o desenvolvimento do trabalho não é permitido o uso de bibliotecas que implementem os algoritmos, mas é permitido usar bibliotecas auxiliares, e.g., que implementem estruturas de dados (ex: NetworkX, etc).

Os algoritmos de busca podem ser úteis em entrevistas de programação em empresas de grande porte, então considere este tempo de implementação como um investimento na sua carreira.

Para comparar os algoritmos, deve ser usado um labirinto com tamanho 100x100 com percentual de bloqueio 40%, e as métricas de comparação são:

1. **tempo de execução;**
2. **número de nós expandidos;**
3. **custo do caminho e**
4. **tamanho do caminho.**

Você deverá criar uma Tabela comparativa com os algoritmos de busca (nas linhas) e as respectivas métricas resultantes (nas colunas) para cada algoritmo.

Ao medir o tempo de execução, desligue a visualização porque o custo de atualização da visualização é maior que o custo do algoritmo. Para a comparação ser justa, tome o cuidado de usar o mesmo labirinto em todos os casos (e.g., fixe o parâmetro seed=42 da classe MazeProblem).

Bom trabalho!

Em adição aos códigos, deve ser escrito um **relatório curto (2-3 páginas)** com a estrutura abaixo:

- **Fundamentação Teórica:** Descrever brevemente os algoritmos que serão implementados e suas diferenças.
- **Experimentos:** Descrever o experimento que será realizado e os resultados esperados considerando a teoria. Incluir a configuração do computador que será usado nos experimentos.
- **Resultados:** Apresentar a comparação dos algoritmos como uma tabela em que linhas são os algoritmos e colunas são as métricas. Discutir se os resultados foram consistentes com o esperado pela teoria. Se não, apresentar hipóteses do porquê.

Pontuação:

Atividade	Pontuação
Implementação do depth-first search:	10 pontos
Implementação do uniform cost search:	20 pontos.
Implementação do A*:	30 pontos.
Relatório:	40 pontos.

Obs:

- Deve ser criada uma visualização, mesmo que simplificada, que permita acompanhar a evolução do algoritmo.
- Em caso de dúvida, converse com o professor.

Bom trabalho!