

Nota 1

Competição de Algoritmos de Busca

1. Definição do Trabalho

Este trabalho deverá ser realizado em um trio (3 alunos) e o restante em duplas (2 alunos), e tem como objetivo implementar e comparar algoritmos de busca heurísticos em problemas práticos, criando uma competição entre equipes para analisar desempenho, eficiência e adequação a diferentes cenários. Valerá 70 pontos para a Nota 1 da disciplina.

Cada **2 equipes** resolverão o **mesmo problema**, porém cada uma implementará um **algoritmo de busca diferente**. Os algoritmos a serem escolhidos são os heurísticos vistos durante o conteúdo dado. O mesmo algoritmo de busca só poderá ser utilizado novamente se for em outro problema.

Os problemas sugeridos são os que seguem:

- **Labirinto:** encontrar o caminho do início ao fim em um labirinto gerado de forma automática e aleatória ou garantindo que o labirinto tenha pelo menos um caminho válido.
- **Ligue 4:** jogo adversarial onde dois jogadores competem para alinhar 4 peças.
- **Damas:** jogo de tabuleiro com captura obrigatória de peças adversárias.
- **Jogo da Velha:** clássico jogo de 3x3, onde o objetivo é alinhar 3 símbolos.
- **Batalha Naval:** encontrar e afundar navios escondidos em um tabuleiro.
- **Quebra-cabeça 8:** rearranjar um tabuleiro de 8 peças deslizantes na ordem correta.
- **As 8 Rainhas:** posicionar 8 rainhas em um tabuleiro de xadrez sem que se ataquem.
- **Passeio do Cavalo:** percorrer todas as casas de um tabuleiro com movimentos de cavalo de xadrez.

Como são 1 trio e 14 duplas, o problema do **Labirinto** terá 3 equipes disputando. Caso alguma equipe queira outro problema, deverá encontrar uma equipe que tope o novo problema e deve ser informado à professora no dia da discussão dos problemas.

2. Datas de Entrega

Dia 10/06/2025, no horário da aula: discussão com as equipes sobre os problemas pretendidos.

Até dia 17/06/2025, 17hs: Constituição das equipes e escolha dos algoritmos e o problema. Essa escolha deverá ser feita no documento `Busca.xlsx`, disponível na pasta `Arquivos`, na plataforma Teams. Não preencher a coluna “Data de Apresentação”. Equipes que não respeitarem esse prazo terão suas notas descontadas.

Até 13/07/2025, 23hs: Entrega, via plataforma Teams, do trabalho. Não serão aceitos trabalhos entregues em outros meios, nem trabalhos atrasados. A implementação e o relatório técnico (com o nome das duas equipes e os algoritmos de busca implementados pelas equipes) devem ser compactados em um arquivo .zip, nomeado com o nome do problema. Basta que apenas um dos integrantes submeta o trabalho.

Dias 14 e 15/07/2025, no horário da aula: Apresentação, pelas equipes, do problema escolhido e das soluções implementadas, durante o horário da aula. A avaliação da apresentação fará parte da nota e será individual. Os alunos que não participarem da apresentação ou não demonstrarem conhecimento no assunto, receberão nota 0 (zero) no item “apresentação”. Caso alguma equipe ou aluno não apresente o trabalho, no dia definido, a equipe/aluno irá receber nota 0 (zero) no trabalho.

3. Como Funcionará a Competição

A competição deve ser adaptada ao tipo de jogo/problema:

Para jogos adversariais (ex: Ligue 4, Damas, Jogo da Velha, Batalha naval):

- Cada algoritmo atuará como um jogador.
- A interface exibirá um único tabuleiro onde os algoritmos alternam jogadas.
- Critério de vitória:
 - Melhor de 5 partidas (alternando quem começa).
 - Métricas: tempo médio por jogada, taxa de vitórias.

Para problemas de caminho/otimização (ex: Labirinto, 8-Puzzle):

- As duas duplas usarão o mesmo problema (ex: labirinto idêntico, mesma configuração inicial do quebra-cabeça).
- A interface exibirá duas janelas sincronizadas, uma para cada algoritmo. Ao final, cada janela mostrará o caminho percorrido (no labirinto) ou a imagem montada (no 8-Puzzle).
- Critério de vitória:
 - Menor tempo para encontrar a solução.
 - Menor custo/número de passos (se aplicável).

Para problemas de posicionamento (ex: 8 Rainhas, Passeio do Cavalo):

- Os algoritmos competirão para encontrar uma solução válida.
- A interface mostrará duas janelas sincronizadas, uma para cada algoritmo. Ao final, cada janela mostrará a solução encontrada.
- Critério de vitória:
 - Menor número de iterações para chegar à solução.

4. O que deve ser entregue

4.1. Implementação

O trabalho pode ser implementado em C/C++, Java ou Python. Todos os códigos devem estar bem comentados e com o nome dos integrantes das equipes. Devem ser entregues todos os códigos-fontes da implementação e todos os arquivos necessários à compilação (incluir arquivo com as instruções, se necessário). Deve estar bem dividido: a parte do algoritmo de cada equipe, além do arquivo principal, que irá permitir a competição.

Os algoritmos de busca escolhidos pela equipe devem ser implementados pela equipe, não será permitido o uso de bibliotecas prontas, que possuam os algoritmos de busca já implementados. As equipes devem chegar a um consenso sobre as linguagens e ferramentas que serão utilizadas no projeto.

Requisitos da implementação:

- Cada equipe deverá implementar o algoritmo de busca escolhido.
- Após, os algoritmos devem ser integrados à interface para permitir a competição.

- Os códigos (tanto de cada algoritmo quanto da interface comum) devem estar documentados e modulares (ex: separação entre lógica do jogo e algoritmos).

As equipes deverão apresentar o trabalho executando para a professora em data e horários a serem disponibilizados. A apresentação acontecerá no LRI (Sala 13, Bloco F).

4.2. Interface comum

Além da implementação dos algoritmos, as equipes deverão desenvolver um aplicativo (*Web* ou *Mobile*) que mostre a competição entre os algoritmos de busca. Se forem jogos adversariais (por exemplo, Jogo da Velha), haverá uma única janela, com a interface do jogo, e cada equipe será um dos jogadores (no exemplo, uma equipe será o X e a outra o O). Se forem jogos do tipo caminho ou labirinto, devem ser mostradas duas janelas, com o mesmo caminho ou labirinto, e cada algoritmo irá executar em uma janela.

A interface comum deve permitir:

- Visualização clara do estado do jogo/problema.
- Controle de início/pausa da competição.
- Exibição de métricas em tempo real (ex: tempo, passos, nós explorados).

Ferramentas sugeridas: Pygame, Tkinter (Python), p5.js (JavaScript), ou Unity.

4.3. Relatório Técnico

As equipes deverão desenvolver um único relatório técnico, sobre o problema que será resolvido, que deverá conter, pelo menos:

- **Introdução:** descrição do problema escolhido pelas equipes e as regras do problema.
- **Metodologia:**
 - Algoritmos escolhidos e justificativa para as escolhas (por que é adequado para o problema?) e incluir o pseudocódigo de cada algoritmo.
 - Detalhes da implementação (ex: heurísticas usadas, limitações).
- **CrITÉrios de vitória:**
 - Em jogos adversariais, número de vitórias em múltiplas partidas (alternando quem começa).
 - Em labirintos/puzzles: tempo de execução, número de nós explorados ou custo do caminho.
- **Métricas de comparação entre os algoritmos:**
 - **Jogos Adversariais** (Damas, Lige 4, Jogo da Velha, Batalha Naval):
 - Vitórias/derrotas em múltiplas partidas.
 - Tempo médio por jogada.
 - **Problemas de Caminho/Otimização** (Labirinto, 8-Puzzle):
 - Custo total do caminho.
 - Número de nós explorados.
 - Tempo de processamento.
 - **Problemas de Posicionamento** (8 Rainhas, Passeio do Cavalo):
 - Iterações para encontrar uma solução válida.
- **Resultados:** média de 5 partidas (alternando quem começa) para jogos adversariais, média de 5 instâncias para caminhos/labirintos.
 - Métricas comparativas (tabelas/gráficos):
 - Tempo de execução.
 - Nós/estados explorados.
 - Custo da solução (se aplicável).
 - Taxa de vitórias (para jogos adversarial).

- **Análise Crítica:**
 - Pontos fortes e fracos de cada algoritmo no contexto do problema.
 - Situações em que um algoritmo superou o outro e por quê.
- **Conclusão:** lições aprendidas e possíveis melhorias.
- **Referências bibliográficas:** para a geração das referências, utilize a ferramenta disponível em: <http://novo.more.ufsc.br/usuarios/login/>. As referências devem estar em ordem alfabética.

O relatório técnico deve ser incluído no arquivo .zip que a equipe irá submeter ao Teams. O relatório deverá estar em formato .pdf.

5. Sobre a apresentação

Nos dias 14 e 15/07/2025, no horário da aula, as equipes de cada problema terão até 15 minutos para apresentar o problema, os algoritmos escolhidos, as soluções implementadas e uma competição entre os dois (ou 3) algoritmos. Não é necessário gastar tempo explicando a teoria dos algoritmos de busca nem mostrando códigos, apenas o projeto desenvolvido pelas duas equipes.