

# Ifes - Campus Serra Bacharelado em Sistemas de Informação Técnicas de Programação Avançada

## Lista de Exercícios

Prof. Mateus Costa

27 de janeiro de 2025

### Instruções

Lista individual para apoio ao estudo para a segunda prova de TPA. Responda todas as questões e inclua diagramas e códigos necessários para a resolução em um arquivo PDF único. Coloque seu nome e número de matrícula e enumere as respostas.

### Exercícios

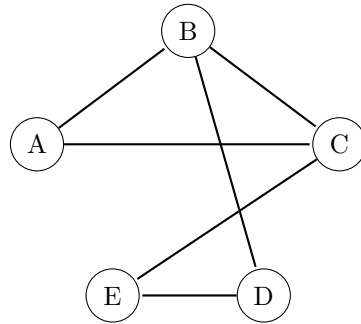
1. Conceitos de Hashing: Explique o conceito de função hash e sua importância na construção de tabelas hash. Dê exemplos de funções hash simples e complexas, explicando como elas distribuem os valores na tabela.
2. Considere uma tabela hash de tamanho 10 com a função hash  $h(k) = k \bmod 10$ . Insira as chaves 12, 22, 42, 39, 47. Mostre o estado da tabela para:
  - a) Listas Encadeadas.
  - b) Endereçamento aberto com sondagem linear.
3. Implemente uma tabela hash em Java utilizando encadeamento separado para resolver colisões. Inclua métodos para inserção, remoção e busca de elementos.
4. Explique o impacto do fator de carga (load factor) no desempenho das operações em uma tabela hash. O que acontece se o fator de carga exceder um limite aceitável? Proponha uma solução.
5. Implemente em Java uma tabela hash com tratamento de colisões usando sondagem quadrática. Mostre como o comportamento da tabela muda em comparação com a sondagem linear.

6. Considere um grafo não direcionado com os vértices  $A, B, C, D, E$  e as seguintes arestas com pesos associados:

$$(A, B, 4), (A, C, 3), (B, C, 2), (B, D, 5), (C, D, 7), (D, E, 6), (C, E, 4)$$

Desenhe o grafo e represente-o usando uma matriz de adjacências.

7. Caminhamento em Grafos: Considere o seguinte grafo não direcionado:

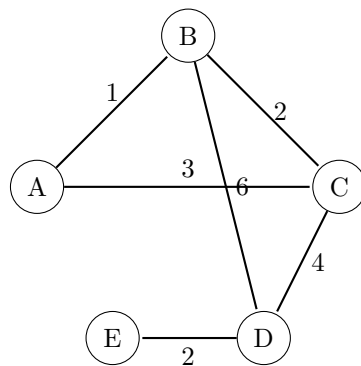


Escreva a ordem de visitação para:

- a) Busca em profundidade (DFS).
- b) Busca em largura (BFS).

Utilize o vértice  $A$  como ponto de partida.

8. Menor Caminho: Considere o seguinte grafo ponderado:



Implemente o algoritmo de Dijkstra para encontrar o menor caminho entre o vértice  $A$  e os demais vértices.

9. Dado um grafo ponderado não direcionado, implemente o algoritmo de Kruskal para encontrar a árvore geradora mínima. Mostre o passo a passo da execução do algoritmo em um exemplo prático.
10. Explique o problema do caixeiro viajante e sua relação com circuitos hamiltonianos. Em seguida, para um grafo com 4 vértices, resolva o problema utilizando força bruta (permutações).
11. Modifique o algoritmo do TSP implementado em aula para encontrar a rota de maior custo em vez do menor custo. Teste com o mesmo grafo do exercício anterior.
12. Adicione um novo vértice ao grafo do exercício anterior e conecte-o com pesos aleatórios aos demais vértices. Recalcule a solução do TSP e compare os resultados.
13. Compare os desempenhos dos algoritmos de Dijkstra e Bellman-Ford em um grafo com 10 vértices e 15 arestas. Explique em quais casos cada algoritmo é mais eficiente.
14. Utilize o grafo do exercício 7 e simule o progresso do algoritmo de busca em largura, imprimindo o estado da fila de vértices em cada passo da execução.
15. Usando a biblioteca JUNG, crie um método para visualizar o grafo do exercício 6 e destacar as arestas de um circuito hamiltoniano encontrado pelo TSP. As arestas devem ser destacadas em uma cor diferente.