

# Teoria dos Grafos - COS 242

## 2021/2

### Segunda Lista de Exercícios

ATENÇÃO! Para um melhor rendimento do processo de aprendizagem, responda às perguntas de forma precisa e completa.

**Questão 1:** Considere um grafo  $G = (V, E)$  qualquer. Calcule o número de arestas do grafo utilizando apenas os vértices e seus respectivos graus. Ou seja, calcule  $m = |E|$  em função de  $d_v$ , onde  $d_v$  representa o grau do vértice  $v \in V$ .

**Questão 2:** Considere duas estruturas de dados para representação de grafos no computador: matriz de adjacência e lista de adjacência. Considere agora um grafo  $G(V, E)$  com  $n$  vértices e  $m$  arestas.

1. Calcule a quantidade de memória (em bytes) necessária para representar o grafo utilizando uma matriz de adjacência. Considere o caso onde o elemento da matriz é um byte (ex., um caractere) e o caso onde o elemento é um bit.
2. Calcule a quantidade de memória (em bytes) necessária para representar o grafo utilizando uma lista de adjacência. Considere o caso onde os vértices do grafo são representados por um número do tipo *inteiro*, que ocupa 4 bytes de memória, e que o ponteiro ocupa 8 bytes de memória (64 bits).
3. Estabeleça a condição tal que a quantidade de memória necessária para representar o grafo utilizando uma lista de adjacência seja menor do que uma matriz de adjacência. Sua condição deve ser em função da *densidade* do grafo. A densidade de um grafo é a fração de arestas que o grafo possui em relação ao grafo completo, ou seja,  $d = m / \binom{n}{2}$ .
4. O grafo dos atores utilizado para calcular o número de Kevin Bacon tem aproximadamente 250 mil vértices e 7,6 milhões de arestas. Calcule a quantidade de memória, em megabytes ( $1\text{MB} = 2^{20}$  bytes), que é necessária para representar o grafo utilizando uma matriz de adjacência e uma lista de adjacência. Qual é a estrutura mais eficiente (em termos de memória) neste caso?

**Questão 3:** Considere um conjunto de  $n$  objetos (i.e., dados) que serão armazenados em memória e assuma que cada objeto tem tamanho fixo igual a  $B$  bytes. Considere agora uma linguagem de programação que oferece suporte ao uso de estrutura de dados, mas que encapsula o acesso à estrutura em chamadas do tipo  $getItem(S, k)$ , onde  $S$  é a estrutura de dados e  $k$  o identificador do objeto,  $1 \leq k \leq n$ .

1. Determine a complexidade computacional (de pior caso) para executar a chamada acima em função de  $n$  e  $k$  quando  $S$  é um vetor ou uma lista encadeada. Explique claramente como este tempo é obtido.
2. Explique por que uma tabela hash (dicionário) não é uma estrutura de dados adequada para este problema.

**Questão 4:** Desenhe todas as árvores geradoras distintas do grafo  $K_3$ . Quantas árvores geradoras distintas este grafo possui? Quantas árvores geradoras distintas possui o grafo  $K_4$ ? Descubra a fórmula para o número de árvore geradoras distintas de  $K_n$  (tente deduzir antes de procurar). Verifique suas respostas.

**Questão 5:** Considere o grafo ilustrado na figura abaixo. Tomando o vértice  $A$  como ponto de partida da busca e assumindo que os vizinhos de um vértice são percorridos (encontrados) em ordem alfabética, determine:

1. A ordem em que os vértices são explorados (ou seja, removidos da fila) ao realizarmos uma busca em largura.
2. A árvore geradora induzida pela busca em largura.
3. A ordem em que os vértices são explorados (ou seja, removidos da pilha) ao realizarmos uma busca em profundidade.
4. A árvore geradora induzida pela busca em profundidade.

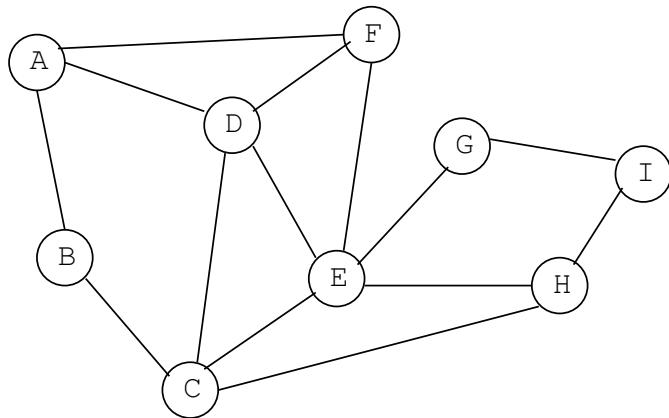


Figura 1: Um grafo não-direcionado.

**Questão 6:** Considere os dois algoritmos de busca apresentado em aula (BFS e DFS) e um grafo conexo  $G$ .

1. Mostre que cada vértice do grafo é explorado exatamente uma vez.
2. Mostre que cada aresta do grafo é examinada exatamente duas vezes.
3. Durante a execução da BFS, qual é o maior número de elementos que a fila pode ter? Qual grafo atinge este valor?
4. Durante a execução da DFS, qual é o maior número de elementos que a pilha pode ter? Qual grafo atinge este valor?

**Questão 7:** Descreva um algoritmo (em pseudo-código) para detectar se um determinado grafo  $G$  é uma árvore. Caso negativo, seu algoritmo deve imprimir um dos ciclos do grafo (mas não todos). O tempo de execução do algoritmo deve ser  $O(m + n)$ , onde  $n$  é o número de vértices e  $m$  é o número de arestas de  $G$ .

**Questão 8:** Modifique o algoritmo de busca em largura (BFS) apresentado em aula de forma a associar cada vértice do grafo ao seu respectivo nível e vértice pai na árvore geradora induzida pela busca. Ou seja, ao final da execução do algoritmo,  $nível[v]$  e  $pai[v]$  devem retornar o nível ao qual pertence o vértice  $v$  e o pai do vértice  $v$  na árvore, respectivamente. Por convenção, o nível do vértice raiz é zero.