Problemas da P1

1. Diâmetro de Árvore

O diâmetro de uma árvore é **a maior distância entre duas folhas da mesma**. Esse valor pode ser obtido através de uma **Busca em Profundidade** que retorna dois parâmetros para **cada subárvore** de um nó: **o diâmetro dessa subárvore e o número de níveis da mesma. Reescreva** o algoritmo de Busca em Profundidade para obter esse diâmetro de uma árvore.

2. Cobertura mínima em árvore

A cobertura **mínima** em uma árvore pode ser obtida a partir de uma **busca em profundidade,** incluindo na cobertura todo vértice que, na busca, **tenha um filho ainda não coberto**. Escrever um algoritmo para obter essa cobertura mínima.

3. Localização de hospital

O governo estadual quer instalar um hospital especializado em **uma única cidade do estado**. Ele quer escolher um local tal que **o tempo da cidade mais distante ao hospital seja o menor possível, para viagens pela malha rodoviária**. Suponha que a distância entre duas cidades vizinhas seja constante. Explicar como resolver esse problema usando grafos (não precisa escrever o algoritmo).

4. Caminhos especiais em um DAG

Considere a **ordenação topológica** em um digrafo. Suponha que só exista uma fonte e um sumidouro. Um caminho especial nessa ordenação é o **maior caminho da fonte ao sumidouro**. Pode haver mais de um caminho especial. **Modifique** o algoritmo que determina o caminho máximo para determinar **quantos caminhos especiais existem** no digrafo.

5. Caminhos máximos em um DAG

Considere um DAG e suponha para o qual já foi preenchido o vetor OT, de ordenação topológica. O digrafo é representado por matriz de adjacências E. Escreva um algoritmo para **determinar se o DAG possui mais de um caminho máximo.**

6. Empilhamento dos caixotes

Considere o problema do empilhamento de caixotes discutido em sala e que é resolvido buscando o **maior caminho em um DAG**. Suponha que possam haver caixotes com **dimensões idênticas.** Explicar qual o problema de admitir caixotes com mesma dimensão no modelo usado e como poderia ser **modificada a modelagem** para considerar esse fato. Exemplificar para o seguinte caso com 5 caixotes de dimensões: (5 x 4), (2 x 3), (4 x 5), (6 x 3), (3 x 2).

7. Grupos de confiança

Uma empresa quer melhorar a produtividade dos diversos grupos de trabalho. Observou que um grupo melhora a produtividade se **todos do grupo confiam uns nos outros**. Então pediu-se a cada funcionário para dizer às pessoas em quem confia diretamente. Note que a relação de confiança é **transitiva** (A confia em B e B em C, então A confia em C). Conhecendo esses resultados, e modelando o problema usando grafos, escreva um algoritmo para determinar qual **o menor número de grupos que podem ser formados.** Explicar a modelagem utilizada.

Exemplo: suponhamos n = número de empregados = 3. Empregados: A, B, C. Relações de confiança: A confia em B, B confia em A. NEsse caso, o número mínimo de grupos é 2.

8. Caminhos simples de tamanho k

Escrever um algoritmo que lista todos os caminhos simples distintos de tamanho k em um grafo simples.

9. Reconhecimento de grafo bipartido

Raquel verificou que um grafo é bipartido sse a árvore de profundidade de uma busca em profundidade qualquer não detectar ciclo ímpar. Escreva um algoritmo baseando-se nessa idéia e verificando que, para determinar a paridade de um ciclo, basta checar a paridade da diferença de níveis, na árvore de profundidade, entre o vértice inicial e final do ciclo.

10. A viagem de Teobaldo

Teobaldo é muito sistemático. Ele quer fazer uma viagem onde, **em cada dia ele quer dormir em uma cidade diferente** e, somente no **k-ésimo dia**, chegar em determinado lugar que é o seu objetivo final. Pode-se viajar entre cidades vizinhas em menos de um dia. Dado um grafo simples e conexo com **n vértices**, que representa o país onde Teobaldo vive, **a cidade onde Teobaldo mora, a** (as cidades são numeradas de 1 a n), **a cidade final b,** e **k**, determinar se é possível a Teobaldo atingir seu objetivo.

11. Diâmetro de grafo

O diâmetro de um grafo é a maior distância entre dois de seus vértices. Escrever um algoritmo para determinar o diâmetro de um grafo.

12. Localização de exércitos

O Ministério do Exército de determinado país quer **posicionar as principais tropas terrestres** em uma cidade tal que **o deslocamento dessas tropas, por terra, seja o mais rápido possível, no pior caso**, para qualquer uma outra cidade do país. É conhecido o grafo de interligação entre as cidades e, por simplicidade, podemos **supor que a distância entre duas cidades interligadas é sempre constante**. Ajude o Ministério escrevendo um algoritmo, utilizando o método de caminhos mínimos de Floyd, que determina **a(s) cidade(s) ideal(is)** para essa localização.

13 - Emparelhamento maximal

Um **emparelhamento M** em um grafo simples G, como se sabe, é um **subconjunto de arestas que não se interceptam duas a duas**. Um **emparelhamento maximal M_m** em um grafo simples G pode ser obtido a partir da busca em profundidade, da seguinte forma: **ao final da busca** de cada vértice v, se o vizinho w na busca não foi emparelhado, então a aresta (v, w) é colocada em **M_m**. **Modifique** o algoritmo de Busca em Profundidade para **encontrar um emparelhamento maximal**. **Exemplifique** com o grafo de estimação.

14 - Fechadura

Considere o seguinte problema: **uma fechadura digital tem 4 dígitos e 3 botões associados**. Sempre que está ociosa, a fechadura fica com **'0000'**. A cada aperto do primeiro botão, o mecanismo soma **291** ao mostrador. A cada aperto do segundo, soma **103**. A cada aperto do terceiro, **19**. A cada aperto de um dos botões, **se a soma ultrapassar 4 dígitos**, o resultado é truncado para 4 dígitos. Quer-se saber o **número mínimo de apertos** que devem ser executados para se atingir a senha que abre a fechadura.
Explicar como modelar esse problema com grafos, de forma a resolvê-lo usando **busca em largura**. **Não precisa escrever o algoritmo.**

14 - Tamanho de ciclo

Reescreva o algoritmo de Busca em Profundidade em um grafo simples para **informar o tamanho de cada ciclo detectado** durante a busca.

15 - Conjunto independente maximal

Um **conjunto independente I** em um grafo simples G, como se sabe, é um **subconjunto de vértices de G tal que nenhum par de vértices de I é vizinho um do outro**. Um **conjunto independente maximal I_m** em um grafo simples G pode ser obtido a partir da busca em profundidade, da seguinte forma: **ao final da busca** de cada vértice v, este é colocado em I_m **caso nenhum de seus vizinhos tenha sido antes colocado nesse conjunto**. **Modifique** o algoritmo de Busca em Profundidade para **encontrar um conjunto independente maximal**. **Exemplifique** com o grafo de estimação.

16 - Jogo da transformação

Considere o seguinte **jogo das 3 letras** entre dois jogadores: cada um escolhe, previamente, um string com 3 letras MAIÚSCULAS. Cada string tem que ser transformado no string inicial do oponente, fazendo-se, sucessivamente, um dos seguintes tipos de transformação no string: **a letra do meio pode trocar de posição com uma das outras**, ou **uma das três letras pode ser modificada para a próxima do alfabeto** (Z pode transformar-se em A). Ganha quem conseguir fazer a transformação com o menor número de operações possível. **Modelar este problema usando grafos** e explicar como sua solução pode ser obtida com uma **busca em largura**. Por exemplo, se o primeiro oponente escolher o string **DMA** e o segundo **OAE**, ganha o primeiro, pois só precisa de 5 operações: **DMA** → **DNA** → **NDA** → **ODA** → **OAD** → **OAE**. O segundo precisará de muito mais que 5 operações.

16 - Bicoloração

Considere um grafo bipartido G. Reescreva o algoritmo de Busca em Profundidade para fazer a coloração desse grafo com apenas duas cores.

17 - Problemas de Maratonas

URI2962 - *Arte Valiosa*
URI2666 - *Imposto Real*
UVA12160 - *Unlock the Lock*
UVA12804 - *The Necronomicon of Computing*
UVA11686 - *Pick up sticks*
UVA10199 - *Tourist Guide*
URI1442 - *Desvio de Rua*