

LISTA 2 - ALGORITMOS EM GRAFOS

Prof. Uéverton Souza

1. Mostre como um percurso em profundidade pode ser utilizado para verificar se um grafo $G = (V, E)$ possui ciclos.
2. Para um grafo $G=(V,E)$ conexo, determine em função de $|V|$ e $|E|$, o número de arestas de retorno que qualquer percurso em profundidade sobre G irá produzir.
3. Idem, para um grafo $G = (V, E)$ qualquer, com $k > 1$ componentes conexas.
4. Prove ou dê um contra-exemplo: Sejam o dígrafo $D = (V, E)$ e o vértice s de $V(G)$ tal que s alcança todos os demais vértices. Para qualquer percurso em profundidade realizado sobre D tendo s como raiz, o conjunto de arestas de retorno produzido é o mesmo.
5. Um *caminho hamiltoniano* em um digrafo é um caminho simples que contém todos os vértices do digrafo. Mostre que, se um digrafo possui um caminho hamiltoniano, é sempre possível percorrê-lo em profundidade de forma que não sejam produzidas arestas de cruzamento.
6. Seja T um conjunto finito e não vazio de tarefas a serem executadas por um indivíduo motivado. Uma tarefa t_1 é denominada um *pré-requisito* para outra tarefa t_2 quando a execução de t_2 só puder ser iniciada após o término da execução de t_1 . Se forem conhecidos todos os pré-requisitos de cada tarefa, é possível modelar a situação através de um digrafo $D = (T, E)$, onde o conjunto E é constituído por pares da forma (t_1, t_2) sempre que a tarefa t_1 for um pré-requisito para a tarefa t_2 . Esboce um algoritmo que, recebendo como entrada este digrafo, sugere ao indivíduo motivado uma ordem viável para execução das tarefas, tal ordem sempre existe.
7. Seja T_G uma tabela contendo os intervalos de vida dos vértices de um grafo G , T_G foi gerada a partir de uma busca em profundidade e contém apenas os intervalos de vida dos vértices. Um exemplo de tabela T_G foi apresentado no slide 87. Faça um algoritmo que receba como entrada uma tabela T_G , construa a floresta em profundidade de G que deu origem a T_G . Em seguida verifique o número de componentes conexas de G .
8. Faça os devidos acréscimos no código de uma busca em profundidade, para determinar as *articulações* e as *pontes* de um grafo G .
9. Faça um algoritmo que: receba como entrada um digrafo D e uma tabela T_D contendo os intervalos de vida dos vértices de D (obtida a partir de uma busca em profundidade em D); e retorne quatro listas L_P, L_R, L_A, L_C contendo respectivamente as arestas de profundidade, retorno, avanço e cruzamento de D com relação a busca em profundidade que resultou em T_D .
10. Seja G uma rede de atividades onde cada vértice representa uma atividade, e cada aresta direcionada de a para b indica que a atividade b só pode ser executada depois da atividade a . O que seria um deadlock nesta rede? Faça um algoritmo para verificar se a rede contém atividades em deadlock.