Construção e Análise de Algoritmos

aula 16: Árvores geradoras mínimas

1. **Árvore geradora máxima**

   Considere o problema de construir uma árvore geradora máxima para um grafo G com pesos associados às arestas.

   Isto é, o problema consiste em encontrar um subconjunto de arestas S sem ciclos que conecta todos os vértices do grafo, cuja soma dos pesos é a maior possível.

   a) Apresente um grafo exemplo, e encontre uma árvore geradora máxima desse grafo.

   b) Descreva uma estratégia gulosa para o problema da árvore geradora máxima.

   c) Argumente que a sua estratégia gulosa sempre encontra uma solução ótima para o problema.

   d) Apresente o pseudo-código do seu algoritmo guloso e analise a sua complexidade.

2. **Remoção de ciclos**

   Nós já sabemos que uma árvore geradora mínima não contém ciclos.

   Então, faz sentido pensar em construir a árvore removendo todos os ciclos do grafo.

   Mais especificamente, como o objetivo é minimizar o custo total da árvore, a ideia seria a seguinte:

   - para cada ciclo de G,
     eliminar a aresta de maior custo que aparece nesse ciclo

   Abaixo nós temos uma maneira simples de implementar essa ideia:

```
- organizar as arestas do grafo em ordem decrescente de custo
- para cada aresta (u,v) nessa lista
-   se existe algum ciclo no grafo que contém a aresta (u,v)
-      remover a aresta (u,v) do grafo
```

   a) Argumente que se uma aresta (u, v) é removida por esse procedimento, então ela não fazia parte de nenhuma árvore geradora mínima de G.

   b) Apresente um algoritmo eficiente que verifica se o grafo possui um ciclo contendo a aresta (u, v) ou não.

   c) Qual o tempo de execução do algoritmo acima?

3. **Minimizando o custo máximo**

   Em alguns casos, pode ser conveniente minimizar o custo da aresta de maior custo na árvore geradora, ao invés da soma total dos custos.

   Nesse caso, nós poderíamos chamar a árvore resultante de uma *árvore geradora de custo máximo mínimo*.

   a) É o caso que toda árvore geradora de custo máximo mínimo é também uma árvore geradora mínima?

      Prove ou dê um contra-exemplo.

   b) É o caso que toda árvore geradora mínima é também uma árvore geradora de custo máximo mínimo?

      Prove ou dê um contra-exemplo.

4. **Atualização de uma árvore geradora**

   Suponha que você já possui uma árvore geradora mínima S para um grafo G.

   E suponha que alguém adiciona a esse grafo uma nova aresta (u, v) com custo c.

   Apresente um algoritmo eficiente que encontra uma árvore geradora do novo grafo G'.

   Nota: Executar o algoritmo de Kruskal no grafo G' leva tempo $O(m \log m)$.

   Portanto, a ideia é que o seu algoritmo execute em menos tempo do que isso.

5. **Tempo de espera**

   Suponha que n programas estão aguardando para serem executados em um supercomputador.

   Cada programa p requer tempo T_p de processamento, e deve ser executado do início ao fim sem interrupção.

   Os processos podem ser executados em qualquer ordem, mas o j-ésimo processo p_j terá um tempo total de espera igual a

   $$\sum_{i=1}^{j-1} T_{p_i}$$

   a) Apresente uma ordem para a execução dos processos que minimiza a soma total dos tempos de espera de todos os processos.

   b) Argumente que a sua solução é a melhor possível.