06 - Árvores geradoras mínimas

Marco A L Barbosa malbarbo.pro.br

1. Exercício 23.1-1 do CLRS3 ou CLRS2.
2. Exercício 23.1-2 do CLRS3 ou CLRS2.
3. Exercício 23.1-3 do CLRS3 ou CLRS2.
4. Exercício 23.1-4 do CLRS3 ou CLRS2.
5. Exercício 23.1-5 do CLRS3 ou CLRS2.
6. Exercício 23.2-2 do CLRS3 ou CLRS2.
7. Exercício 23.2-4 do CLRS3 ou CLRS2.
8. Exercício 23.2-5 do CLRS3 ou CLRS2.
9. Exercício 23.2-6 do CLRS3 ou CLRS2.
10. Exercício 23.2-8 do CLRS3 ou CLRS2.

Referências

- [CLRS2] - Thomas H. Cormen et al. Introduction to Algorithms. 2$^{\text{nd}}$ edition. Capítulo 23.
- [CLRS3] - Thomas H. Cormen et al. Introduction to Algorithms. 3$^{\text{rd}}$ edition. Capítulo 23.

Exercício 1 (23.1-1)

Vamos considerar execução do algoritmo GENERIC-MST. Na primeira iteração $A = \emptyset$, se considerarmos um corte $(S, V - S)$ com $S = \{u\}$, então o corte respeita A e (u, v) é uma aresta leve cruzando o corte, portanto, pelo teorema 23.1 (u, v) é segura para A. Desta forma (u, v) pode ser adicionada a A e no final do algoritmo A será uma árvore geradora mínima que contém (u, v).

Exercício 2 (23.1-2)

Considere um grafo com os vértices a, b, c e com as arestas (a, b) com peso 1 e a aresta (a, c) com peso 2. A árvore geradora mínima deste grafo é única e inclui as arestas (a, b) e (a, c). Quando $A = \emptyset$ e $S = \{a\}$, o corte $(S, V - S)$ respeita A, a aresta (a, c) cruza o corte e é segura para A (pode ser adicionada a A e A continua sendo subconjunto de uma árvore geradora mínima), mas ela não é leve. Portanto a conjectura do Professor Sabatier é falsa.

Exercício 3 (23.1-3)

Seja T uma árvore geradora mínima que contenha (u, v). Vamos supor que (u, v) não é uma aresta leve para algum corte do grafo e mostrar uma contradição encontrando uma árvore T' com menor peso que T. Iniciamos removendo a aresta (u, v) de T, gerando dois componentes conexos, $T_1 = (V_1, E_1)$ e $T_2 = (V_2, E_2)$. Seja (x, y) a aresta leve que cruza o corte $(V_1, V - V_1)$. Note que a aresta (u, v) também cruza o corte e $(x, y) \neq (u, v)$. Obtemos a árvore T' conectando os componentes T_1 e T_2 com a aresta (x, y), isto é $T' = T - \{(u, v)\} + \{(x, y)\}$. Como o peso de (x, y) é menor que o peso de (u, v), a árvore T' tem peso menor que a árvore T, o que é uma contradição pois T é uma árvore geradora mínima.

Exercício 9 (23.2-6)

Podemos usar o bucket sort para ordenar as arestas no algoritmo de Kruskal em tempo médio $O(E)$, desta forma o tempo de execução é dominado pelas operações de conjuntos disjuntos, com tempo $O(E\alpha(V))$. Veja que este tempo ainda é maior que para o algoritmo de Prim usando heap de Fibonacci, que é $O(E + V \lg V)$.