Algoritmos em Grafos
Lista de Exercícios V

1. Sejam $T_1 = (V_1, E_1)$ e $T_2 = (V_2, E_2)$ duas árvores geradoras de um grafo G. Mostre que para qualquer $(u, v) \in E_1 \setminus E_2$ há outra aresta $(x, y) \in E_2 \setminus E_1$ tal que
$$T' = T_1 - (u, v) + (x, y)$$
é uma árvore geradora de G, assim como
$$T'' = T_2 - (x, y) + (u, v)$$
também é uma árvore geradora de G.

2. Seja G um grafo com pesos nas arestas. Projete um algoritmo para se encontrar uma árvore geradora de G com peso **máximo**. Analise a complexidade e corretude de seu algoritmo.

3. Apresente uma implementação não recursiva da função **find-set** para estruturas de union-find.

4. Qual é o menor e maior número de elementos em um heap-binário de altura h.

5. Mostre que um heap-binário com n elementos tem altura $\lfloor \log_2 n \rfloor$.

6. Mostre que há no máximo $\lceil \frac{n}{2^{h+1}} \rceil$ nós de altura h em um heap com n elementos.

7. Seja (u, v) uma aresta de peso mínimo em um grafo conexo G. Mostre que (u, v) pertence a alguma MST de G.

8. Seja $G = (V, E)$ um grafo com pesos $w(e)$ nas arestas e seja $A \subseteq E$ um subconjunto de arestas de uma MST. Suponha $S \subseteq V$ tal que o corte $\delta(S)$ respeita A, e seja $(u, v) \in \delta(S)$ uma aresta segura para A. É verdade que (u, v) é uma aresta *light* para o corte $\delta(S)$? Prove sua resposta.

9. Mostre que se uma aresta (u, v) faz parte de alguma MST, então ela é uma aresta *light* para algum corte de G.

10. Seja T uma MST de G, e seja L uma lista contendo os custos das arestas de T ordenados por peso. Seja T' uma outra MST de G. Mostre que L também é uma lista contendo os custos das arestas de T' ordenados por peso.

11. O algoritmo de Kruskal pode devolver MSTs distintas dependendo de como o algoritmo faz a seleção de diferentes arestas mas de mesmo peso e que conectam componentes distintos. Mostre que para cada MST T do grafo G é possível criar uma ordenação de arestas (elas estarão em ordem

crescente, só arestas de mesmo custo podem mudar de posição entre si) de tal forma que a aplicação do algoritmo de Kruskal gera exatamente a árvore T.

12. Suponha que num grafo G tenhamos apenas custos inteiros no intervalo $[1, |V|]$ para as arestas. Projete um algoritmo $o(E \log V)$ para encontrar uma MST em grafos deste tipo.

13. Suponha que num grafo G tenhamos apenas custos inteiros no intervalo $[1, |V|]$ para as arestas. É possível alterar o algoritmo de Prim para grafos deste tipo de tal forma a obter um algoritmo com complexidade $o(E \log V)$.

14. Uma árvore geradora com gargalo mínimo (*minimum bottleneck spanning tree*) é uma árvore geradora tal que o custo da aresta mais pesada é mínimo dentre todas as árvores geradoras. Mostre que uma MST também é uma árvore geradora de gargalo mínimo.