MAC 5711 - Análise de Algoritmos

Departamento de Ciência da Computação

Segundo semestre de 2018

Lista 7

1. Seja s um vértice de um digrafo G com custos positivos nos arcos. Para cada vértice v de G, seja $v.x$ o custo de *algum* caminho de s a v em G. Escreva um algoritmo eficiente que verifique se $v.x$, para todo v, é a distância de s a v em G. Explique porque seu algoritmo está correto.

2. Mostre que o algoritmo de Dijkstra pode produzir resultados errados se o digrafo tiver arcos de custo estritamente negativo.

3. Escreva um algoritmo que recebe conjuntos S e T de vértices de um grafo e calcula a distância de S a T, ou seja, o custo de um caminho de custo mínimo que começa em algum vértice em S e termina em algum vértice em T. O algoritmo deve consumir o mesmo tempo de execução que o algoritmo de Dijkstra. Justifique que seu algoritmo está correto. *Dica:* Basta introduzir pequenas modificações no algoritmo de Dijkstra.

4. Escreva um algoritmo que encontre um arco cuja remoção causa o maior aumento na distância de um vértice s a um vértice t.

5. Suponha que trocamos a linha 4 do algoritmo do Dijkstra como segue

   $$4.\quad \text{while } |Q| > 1$$

   Isso faz com que a execução do laço execute $|V|-1$ vezes no lugar de $|V|$ vezes. Será que o algoritmo continua correto?

6. Dado um digrafo $G = (V, E)$ em que cada aresta $(u, v) \in E$ tem associado um valor $r(u, v)$, que é um número real no intervalo $[0, 1]$ que representa a confiança de um canal de comunicação do vértice u até o vértice v. Interpretamos $r(u, v)$ como a probabilidade de que o canal de u a v não falhe, e supomos que tais probabilidades são independentes. Dê um algoritmo eficiente (mesmo tempo de execução que o de Dijkstra) que acha um caminho mais confiável entre dois vértices dados.

7. Seja $G = (V, E)$ um digrafo com pesos $w : E \to \{0, 1, \ldots, W\}$ para algum W. Modifique o algoritmo de Dijkstra para que compute os caminhos mínimos a partir de um vértice s em tempo $\mathrm{O}(W|V| + |E|)$.

8. Seja $G = (V, E)$ um digrafo com pesos inteiros $w : E \to \{0, 1, \ldots, W\}$ para algum W. Modifique o algoritmo de Dijkstra para que compute os caminhos mínimos a partir de um vértice s em tempo $\mathrm{O}((|V| + |E|)\lg W)$. (*Dica:* Quantas estimativas distintas de caminhos mínimos podem existir em $V - S$ em cada iteração do algoritmo?)

9. **(CRLS Ex. 23.1-1)** Seja e uma aresta de custo mínimo em um grafo G com custos nas arestas. É verdade que e pertence a alguma MST de G? É verdade que e pertence a toda MST de G?

10. Suponha que os custos das arestas de um grafo conexo são distintos dois a dois (ou seja, não há duas arestas com o mesmo custo). Mostre que o grafo tem uma única MST.

11. Suponha que os custos das arestas de um grafo conexo são distintos dois a dois. Seja C um ciclo não trivial. É verdade que a aresta de custo mínimo em C pertence à (única) MST do grafo?

12. Seja G um grafo conexo com custos nas arestas. Uma aresta e de G é crítica se o aumento do custo de e faz com que o custo de uma MST de G também aumente. Escreva uma função que determine todas as arestas críticas de G em tempo $\mathrm{O}(m \log n)$

13. Mostre que depois de cada execução da linha 6 do algoritmo de Prim tem-se u.key $< \infty$

14. Suponha que temos um grafo G com pesos nas arestas. Verdadeiro ou falso: Para qualquer MST T de G, existe uma execução válida do algoritmo de Kruskal que produz T como saída? Dê uma prova ou um contra-exemplo.

15. Seja G um grafo conexo com custos nas arestas e seja B um conjunto de arestas de G. Suponha que o grafo induzido por B não tem circuitos. Queremos encontrar uma subárvore geradora de custo mínimo dentre as que contêm B. Descreva um algoritmo eficiente para resolver o problema.

16. **(CRLS Ex. 23.2-4,5)** Suponha que todos os pesos num grafo com n vértices são inteiros no intervalo de 1 até n. Descreva como otimizar os algoritmos de Kruskal e Prim nesta situação. O que acontece se os pesos são inteiros no intervalo de 1 até W, onde esse valor é um inteiro dado, mas arbitrário?

17. Dado um grafo com n vértices, pesos distintos nas arestas, e no máximo $n + 8$ arestas, dê um algoritmo com complexidade $\mathrm{O}(n)$ para achar uma MST.

18. Uma coleção $\mathcal{C}$ de cláusulas sobre um conjunto X de variáveis booleanas é uma *tautologia* se toda atribuição a X satisfaz $\mathcal{C}$. O problema TAUTOLOGIA consiste em, dado X e $\mathcal{C}$, decidir se $\mathcal{C}$ é ou não uma tautologia. O problema TAUTOLOGIA está em NP? Está em coNP? Justifique suas respostas.

19. (difícil) O problema 2-SAT consiste na restrição de SAT a instâncias X e $\mathcal{C}$ em que toda cláusula de $\mathcal{C}$ tem exatamente dois literais. Mostre que o 2-SAT está em P, ou seja, descreva um algoritmo polinomial que resolva o 2-SAT.

20. Seja $G = (V, E)$ um grafo. Uma *3-coloração* de G é uma função $c : V \to \{1, 2, 3\}$ tal que $c(u) \neq c(v)$, para toda aresta $uv \in E$.

    Considere o

    Problema 3-COLORAÇÃO**:** Dado um grafo, determinar se ele tem uma 3-coloração.

    Mostre que o 3-COLORAÇÃO está em NP.

21. Mostre que 2-COLORAÇÃO está em P.

22. Mostre que o problema abaixo é NP-completo.

    Problema PARTIÇÃO**:** Dada uma coleção S de números, decidir se existe uma subcoleção S' de S cuja soma é igual a soma dos números em $S \setminus S'$, ou seja,

$$\sum_{x \in S} x = \sum_{x \notin S} x.$$

23. Mostre que o problema abaixo é NP-completo.

    Problema MOCHILA**:** Dado um número W, um número V, um número inteiro positivo n, uma coleção de números $w_1, \ldots, w_n$, e uma coleção de números $v_1, \ldots, v_n$, decidir se existe um subconjunto S de $\{1, \ldots, n\}$ tal que

$$\sum_{i \in S} w_i \leq W \quad \text{e} \quad \sum_{i \in S} v_i \geq V.$$