Universidade Estadual de Campinas – Instituto de Computação

MO417A – Complexidade de Algoritmos – 1º Semestre de 2020

`https://www.ic.unicamp.br/~lehilton/mo417a`

Caminhos mínimos

Caminhos mínimos com uma origem

Questão 1. (CLRS) Exercícios: 24.2-1, 24.2-2, 24.2-3, 24.2-4,

Algoritmo de Dijkstra

Observação: há inconsistência entre os números de exercícios da 2ª e 3ª edição.

Questão 2. (CLRS) Exercícios: 24.3-1, 24.3-2, 24.3-3, 24.3-4 (3ed), 24.3-5 (3ed), 24.3-6 (3ed), 24.3-4 (2ed), 24.3-7 (3ed), 24.3-7 (2ed), 24.3-8 (3ed), 24.3-6 (2ed), 24.3-10 (3ed), 24.3-8 (2ed),

Questão 3. Considere um grafo direcionado $G = (V, E)$ cujas arestas têm pesos 0 ou 1. Projete um algoritmo de tempo $O(V + E)$ que obtém uma árvore de caminhos mínimos a partir de um vértice s.

Questão 4. Considere um grafo direcionado $G = (V, E)$ com pesos nas arestas w que valem 0 ou 1, i.e., para todo $e \in E$, $w(e) \in \{0, 1\}$. Nessa situação, é possível modificar o algoritmo de Dijkstra para que ele calcule os caminhos mais curtos a partir de um vértice de origem s em tempo total $O(V + E)$.

Algoritmo de Bellman-Ford

Questão 5. (CLRS) Exercícios: 24.1-1, 24.1-2, 24.1-4, 24.1-6(*)

Questão 6. (CLRS) Problemas: 24-2

Questão 7. (Adaptado de (CLRS)) 24.1-3 Dado um gráfico ponderado e direcionado $G = (V, E)$ sem ciclos de peso negativo, seja m o máximo, entre todos os vértices $v \in V$, do número mínimo de arestas em um caminho mínimo da fonte s para v. (Aqui, o caminho mínimo é por peso, e não por número de arestas.) Reescreva o algoritmo Bellman-Ford para ele termine em $m + 1$ passos, mesmo que m não seja conhecido com antecedência.

Questão 8. Escreva um algoritmo que verifica se há ciclos negativos em um grafo direcionado e, se houver, devolva um tal ciclo.

Sistemas de diferenças

Questão 9. (CLRS) Exercícios: 24.4-1, 24.4-2, 24.4-3, 24.4-6, 24.4-8,

Caminhos mínimos entre todos os pares de vértices

Questão 10. (CLRS) Exercícios: 25.2-1, 25.2-2, 25.2-3, 25.2-4, 25.2-5, 25.2-6, 25.2-7, 25.2-8, 25.2-9,

Questão 11. (CLRS) Problemas: 25-1

Aplicações

Questão 12. Milda é a presidenta de um determinado país B. Esse país é dividido em estados e cada estado possui uma capital. Milda quer reestruturar o sistema de estradas e ferrovias e precisa da sua ajuda. As ferrovias são muito antigas e seu custo de manutenção é alto. Seu trabalho é ajudar a presidenta a decidir quais ferrovias podem ser desativadas. No entanto, como B é um país democrático, uma ferrovia só pode ser desativada se isso não piorar a qualidade do sistema de transporte, isso é, uma ferrovia pode ser desativada apenas se a distância de cada cidade à capital mais próxima não for modificada. Considere que o país B tem n cidades e que se uma estrada ou ferrovia liga a cidade i à cidade j, então ela pode ser utilizada para ir tanto de i para j como de j para i. Obtenha um algoritmo eficiente para descobrir quantas ferrovias podem ser desativadas.

Questão 13. Enquanto grafos acíclicos têm propriedades estruturais muito boas, eles são muito restritos. Em algumas aplicações podem aparecer grafos que são "quase" acíclicos. Para uma constante k, um grafo direcionado G é *k-quase-acíclico* se o passeio fechado com maior número de vértices tem no máximo k vértices distintos. (Pense em um valor de k bem pequeno e tente desenhar um exemplo de um grafo k-quase-acíclico. O que você sobe sobre G^{CFC}?).

(a) Seja $G = (V, E)$ um grafo direcionado k-quase-acíclico sem ciclos negativos e s um vértice de G. Projete um algoritmo para encontrar uma árvore de caminhos mínimos a partir de s. Seu algoritmo deve executar em tempo $O(f(k) \cdot E + V)$, onde f é uma função. Você deve descrever o algoritmo em alto nível (i.e., você pode utilizar sub-rotinas conhecidas e ignorar detalhes de implementação)

Observe que $f(k)$ pode ser muito grande, mas se k for pequeno, então o tempo de execução é dominado por $O(V + E)$. Uma implementação em tempo $\Theta(k^3 \cdot E + V)$ (ou mesmo mais rápida) pode ser encontrada; mas nessa questão basta qualquer algoritmo com a complexidade pedida.

(b) Argumente *brevemente* que o algoritmo acima está correto. Você não precisa ser completamente formal, mas seja claro e não utilize frases ou termos ambíguos. Releia duas vezes. Se tiver dúvida se algum termo é impreciso ou ambíguo, então ele é.