

Prova

Questão 1 (Complexidades, 2.5pt)

Para os algoritmos a seguir, liste suas complexidades e forneça uma breve explicação de como as complexidades são obtidas (i.e. o que explica os diferentes termos da complexidade).

Dijkstra, Edmonds-Karp, Caminho mais gordo, Hopcroft-Karp,
Algoritmo Húngaro.

Questão 2 (Caminhos mais curtos com distâncias negativas, 2.5pt)

- a) Sob quais condições os algoritmos de Bellman-Ford e Johnson produzem resultados corretos?
- b) No contexto do algoritmo de Johnson, explique a ideia de potenciais, como eles podem ser obtidos e o que eles alcançam.

Questão 3 (Fluxo máximo, 2.5pt)

Considere o problema do fluxo máximo num grafo direcionado capacitado entre dois vértices s e t .

- a) Descreva brevemente, mas de forma precisa, o conteúdo do teorema do corte mínimo do fluxo máximo.
- b) Dê um exemplo em que temos vários fluxos máximos diferentes (obviamente de mesmo valor), mas um único corte mínimo, ou mostra que isso é impossível.
- c) Dê um exemplo em que temos um único fluxo máximo, mas vários cortes mínimos, ou mostra que isso é impossível.
- d) Pergunta bônus: se um grafo tiver vários cortes mínimos, qual corte mínimo devolvem os algoritmos do tipo Ford-Fulkerson? Caracterize os cortes devolvidos.

Questão 4 (Algoritmos randomizados, 2.5pt)

Você tem um algoritmo de decisão do tipo Las Vegas que executa em tempo polinomial, mas por um problema na sua máquina de execução, cada resposta é invertida com probabilidade $p = 0.1$. Forneça uma estratégia para reduzir a probabilidade de erro para menos de 0.03. Demonstra o resultado.

Questão 5 (Heaps, 2.5pt)

Uma colega tem três ideias para fazer com que os heaps funcionem melhor. Elas funcionam? (Nesta pergunta, assumiremos min heaps.)

- a) Suponha que implementemos o deletemin com um aumento para ∞ seguido de heapify-down – que torna a chave uma folha – e, em seguida, removendo a chave. Isso funciona?
- b) Ela também observa que sempre há a possibilidade de permutar os irmãos (juntamente com suas subárvores), já que, pela propriedade de heap, as chaves dos pais são sempre menores ou iguais. Isso pode ser usado, por exemplo, para garantir que a chave do filho da esquerda seja sempre menor do que a do filho da direita.
- c) Finalmente ela observa que sempre há a possibilidade de permutar todas as folhas, por um argumento similar ao item b).

Para cada item, apresentar uma prova ou um contra-exemplo.

Questão 6 (Caminhos mais curtos, 2.5pt)

Um colega afirma que o algoritmo de Dijkstra não apenas calcula as distâncias mais curtas para todos os vértices a partir de um vértice de origem s , mas também é capaz ordenar os vértices em ordem crescente de distâncias (por exemplo, por guardar em um vetor em ordem de processamento). Portanto, ele conclui que um tal algoritmo tem complexidade de pelo menos $\Omega(n \log n)$. Ele está certo? Discuta.

Sucesso!