Projeto e Análise de Algoritmos (2023-2)

Exercícios

Prof. Flávio L. C. de Moura

11 de dezembro de 2023

Exercício 1. *Mostre como podemos ordenar n inteiros contidos no intervalo de 0 a $n^3 - 1$ em tempo linear, ou seja, em tempo $O(n)$.*

Exercício 2. *Dizemos que um algoritmo de ordenação é* estável *se não altera a posição relativa dos elementos que têm o mesmo valor. Considere o pseudocódigo do algoritmo Counting Sort:*

```
 1 let C[0..k] be a new array;
 2 for i = 1 to k do
 3 |  C[i] ← 0;
 4 end
 5 for j = 1 to A.length do
 6 |  C[A[j]] ← C[A[j]] + 1;
 7 end
 8 for i = 1 to k do
 9 |  C[i] ← C[i] + C[i − 1];
10 end
11 for j = A.length downto 1 do
12 |  B[C[A[j]]] ← A[j];
13 |  C[A[j]] ← C[A[j]] − 1;
14 end
```

Algoritmo 1: Counting-Sort(A, B, k)

1. *Mostre que este algoritmo é estável.*
2. *Suponha que o laço* ***for*** *da linha 11 seja reescrito como*
   for $j = 1$ *to A.length* ***do***
   (a) *O algoritmo modificado é correto?*
   (b) *O algoritmo modificado é estável?*

Exercício 3. *Considere o algoritmo a seguir, onde $G = (V, E)$ é um grafo, $u \in V$ é um vértice de G e F uma fila inicialmente vazia:*

```
 1 marque u;
 2 insira u na fila F;
 3 while F ≠ ∅ do
 4 |  retire o primeiro elemento v de F;
 5 |  for each w ∈ G.Adj[v] do
 6 |  |  if w não está marcado then
 7 |  |  |  marque w;
 8 |  |  |  insira w em F;
 9 |  |  end
10 |  end
11 end
```

Algoritmo 2: algoritmo(G, u)

Faça a análise assintótica do tempo de execução deste algoritmo.

Exercício 4. *Seja $G = (V, E)$ um grafo (não-dirigido) conexo com função peso $w : E \to \mathbb{R}$. Dizemos que uma aresta de G é* útil *se não ocorre em nenhum ciclo de G. Prove que qualquer árvore geradora mínima de G contém todas as arestas úteis de G.*

Exercício 5. *Considere o seguinte problema: Há uma fila de n moedas cujos valores são alguns inteiros positivos $c_1, c_2, ..., c_n$, não necessariamente distintos. O objetivo é pegar a quantidade máxima de dinheiro sujeita à restrição de que não se pode pegar duas moedas adjacentes na fila inicial.*

1. *Construa uma recorrência para calcular o montante máximo $F(n)$ que pode ser obtido de uma fila com n moedas.*
2. *Qual a complexidade da abordagem de força bruta para este problema?*
3. *Construa uma solução utilizando programação dinâmica e faça a análise assintótica da sua solução.*

Exercício 6. *Construa um algoritmo eficiente para calcular o coeficiente binomial $C(n, k)$, também denotado por $\binom{n}{k}$, sem utilizar multiplicações. Em seguida, faça a análise assintótica do seu algoritmo.*

Exercício 7. *Considere o problema da mochila booleana (ou problema da mochima 0-1): Dado um conjunto de n objetos com peso w_i e valor v_i $1 \leq i \leq n$, e uma mochila com capacidade de carregar o peso W, onde W, w_i e v_i são inteiros para $1 \leq i \leq n$, quais objetos devem ser colocados na mochila para que o valor total seja máximo? Mostre que a estratégia gulosa não resolve este problema, e construa uma solução utilizando programação dinâmica para este problema.*

Exercício 8. *Considere o problema da mochila fracionária: Considere n objetos com peso w_i e valor v_i $1 \leq i \leq n$, e uma mochila com capacidade de peso W, de forma que frações de cada objeto podem ser selecionadas. Que fração de cada objeto deve ser colocada na mochila de modo a maximizar o valor total? Em outras palavras, selecione frações $f_i \in [0, 1]$ dos itens tais que $\sum_{i=1}^{n} f_i.w_i \leq W$ e $\sum_{i=1}^{n} f_i.v_i$ é máximo. Mostre que este problema possui a propriedade da escolha gulosa.*

***Observação**: Propriedade da escolha gulosa: uma solução ótima global pode ser obtida a partir de escolhas gulosas ótimas locais.*

Exercício 9. *Seja A um problema de decisão na classe P. Demonstre que se um problema B pode ser reduzido polinomialmente ao problema A, então $B \in P$.*

Exercício 10. *Considere um tabuleiro $m \times m$, onde cada uma das m^2 posições contém uma pedra azul, uma pedra vermelha ou não contém nada. Você joga removendo pedras do tabuleiro de forma que cada coluna do tabuleiro contenha pedras de uma única cor, e cada linha contenha pelo menos uma pedra. Você ganha se atinge este objetivo. Dependendo da configuração inicial, pode-se ganhar ou não. Prove que este problema, isto é, do tabuleiro inicial possuir uma configuração vencedora, é NP-completo.*
***Dica**: Faça uma redução a partir de 3-SAT.*

Exercício 11. *Considere o seguinte jogo em um grafo (não-dirigido) G, que inicialmente contém 0 ou mais bolas de gude em seus vértices: um movimento deste jogo consiste em remover duas bolas de gude de um vértice $v \in G$, e adicionar uma bola a algum vértice adjacente de v. Agora, considere o seguinte problema: Dado um grafo G, e uma função $p(v)$ que retorna o número de bolas de gude no vértice v, existe uma sequência de movimentos que remove todas as bolas de G, exceto uma? Mostre que este problema é NP-completo.*