

## Lista 2 de Complexidade de Algoritmos - 2018.03

Data de entrega: 15/10/2018

Em cada uma das questões a seguir, o termo “resolver o problema” significa descrever uma solução completa para o problema considerado (através de pseudocódigo ou por linguagem natural), fornecendo prova de corretude e análise de complexidade da solução dada.

1. (*Dividir para conquistar*). Resolva o problema de encontrar um *par de pontos mais próximos* no  $\mathbb{R}^2$ , definido a seguir:

Dados  $n$  pontos em um espaço métrico, encontrar um par de pontos com a menor distância dentre as distâncias entre quaisquer dois destes pontos.

2. (*Programação dinâmica*). Resolva o problema a seguir:

Dada uma sequência  $s$  de caracteres, encontre a maior subsequência de  $s$  que é um palíndromo.

3. (*Algoritmo guloso*). Resolva o problema do troco:

Considerando que existem moedas de 1, 5, 10, 25, 50 e 100 unidades monetárias, ao considerar uma quantidade de dinheiro  $T$  (o troco) determina este valor com a menor quantidade de moedas.

Mostre que se as moedas são de 1, 15 e 20 unidades monetárias, então a estratégia gulosa de escolher a de maior valor primeiro pode não produzir uma solução ótima.

4. Resolva o problema da mochila 0-1:

Sejam  $n$  itens tais que o  $i$ -ésimo item tem valor  $v_i$  e peso  $w_i$ . Com uma mochila de capacidade  $W$ , determinar o valor máximo que podemos carregar.

5. Resolva o problema da mochila fracionária:

Sejam  $n$  itens tais que o  $i$ -ésimo item tem valor  $v_i$  e peso  $w_i$ . Com uma mochila de capacidade  $W$  e assumindo que podemos tomar uma frações dos itens (por exemplo, podemos tomar  $\frac{3}{4}$  do item  $i$  com valor  $\frac{3}{4}v_i$  e peso  $\frac{3}{4}w_i$ ), determinar o valor máximo que podemos carregar.

**Observação.** Resolver cada uma das questões em uma folha separada para agilizar a correção.