# MAC 338 - Análise de Algoritmos

*Departamento de Ciência da Computação*

Primeiro semestre de 2013

## Lista 5

1. Mostre um exemplo para cada um dos três primeiros critérios gulosos apresentados para o problema da coleção máxima de intervalos disjuntos visto em aula que prove que o algoritmo obtido usando estes critérios pode produzir um conjunto de intervalos disjuntos que não é máximo.

2. Considere o algoritmo visto em aula para o problema da coloração de intervalos. Modifique-o para que, além da $m$-coloração $c$, ele devolva um conjunto $S$ de $m$ intervalos e um instante $t$ tal que $s[i] \leq t < f[i]$ para todo $i$ em $S$.

3. Considere o algoritmo do problema da coloração de intervalos visto em aula. Nele, os intervalos são ordenados no começo pelo valor de $s[i]$. O que acontece se ordenarmos os intervalos por $f[i]$ em vez de $s[i]$? O algoritmo continua correto? Prove, apresentando um certificado como o do exercício anterior para ele, ou dê um contra-exemplo.

4. Use códigos de Huffman para o conjunto de caracteres do enunciado deste exercício. Inclua todos os caracteres. Quantos bits foram economizados no armazenamento do enunciado desse exercício usando códigos de Huffman versus uma codificação onde todos os caracteres são codificados por cadeias de bits do mesmo comprimento?

5. Considere um conjunto de livros numerados de 1 a $n$. Suponha que o livro $i$ tem peso $p_i$ e que $0 < p_i < 1$ para cada $i$.

   **Problema:** Dado $n$ e os números $p_1, \ldots, p_n$, acondicionar os livros no menor número possível de envelopes de modo que cada envelope tenha no máximo 2 livros e o peso do conteúdo de cada envelope seja no máximo 1.

   Escreva um algoritmo guloso que resolva o problema em tempo $O(n \log n)$. (Sugestão: comece por escrever um algoritmo recursivo que apenas calcula o número mínimo de envelopes.) Aplique seu algoritmo a um exemplo interessante. Mostre que seu algoritmo está correto.

6. **(CLRS 16-4)** Seja $1, \ldots, n$ um conjunto de *tarefas*. Cada tarefa consome um dia de trabalho; durante um dia de trabalho somente uma das tarefas pode ser executada. Os dias de trabalho são numerados de 1 a $n$. A cada tarefa $t$ está associado um *prazo* $p_t$: a tarefa deveria ser executada em algum dia do intervalo $1 \,.\,.\, p_t$. A cada tarefa $t$ está associada uma *multa* não-negativa $m_t$. Se uma dada tarefa $t$ é executada depois do prazo $p_t$, sou obrigado a pagar a multa $m_t$ (mas a multa não depende do número de dias de atraso).

   **Problema:** Programar as tarefas (ou seja, estabelecer uma bijeção entre as tarefas e os dias de trabalho) de modo a minimizar a multa total.

   Escreva um algoritmo guloso para resolver o problema. Prove que seu algoritmo está correto. Analise o consumo de tempo.

7. A entrada é uma seqüência de números $x_1, x_2, \ldots, x_n$ onde $n$ é par. Projete que particione a entrada em $n/2$ pares da seguinte maneira. Para cada par, computamos a soma de seus números. Denote por $s_1, s_2, \ldots, s_{n/2}$ as $n/2$ somas. O algoritmo deve encontrar uma partição que minimize a máximo das somas e deve ser tão eficiente quanto possível. Explique porque ele funciona e determine a sua complexidade.

8. Descreva um algoritmo eficiente que, dado um conjunto $\{x_1, x_2, \ldots, x_n\}$ de pontos na reta real, determine o menor conjunto de intervalos fechados de comprimento um que contém todos os pontos dados. Justifique informalmente o seu algoritmo e analise a sua complexidade.

9. Considere uma estrada calma e longa, com algumas poucas casas à beira. (Podemos imaginar a estrada como uma linha reta, com uma extremidade leste e uma extremidade oeste.) Suponha que, apesar da atmosfera bucólica, os moradores dessas casas são ávidos usuários de telefones celulares. Você quer colocar estações-base de telefonia celular ao longo da estrada, de maneira que toda casa esteja a no máximo 6 quilômetros de uma das estações-base.

   Dê um algoritmo eficiente que atinge esse objetivo usando um número mínimo de estações-base. Justifique sua resposta.