MAC 338 - Análise de Algoritmos

Departamento de Ciência da Computação

Segundo semestre de 2019

Lista 6

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. A entrada é uma sequência de números $x_1, x_2, \ldots, x_n$ onde n é par. Projete um algoritmo 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.

5. 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.

6. 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* $m_t \geq 0$. 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.

   Considere o algoritmo que primeiramente ordena as tarefas pela multa; a seguir, considera uma a uma cada tarefa, escalonando-a no último dia livre dentro do seu prazo. Se não houver dia livre dentro do prazo, escalona a tarefa no último dia livre.

   Prove que esse algoritmo está correto, ou seja, produz um escalonamento com multa mínima. Analise seu consumo de tempo.

7. Um pequeno negócio, digamos, uma lojinha de xerox com uma única máquina de xerox, enfrenta o seguinte problema de escalonamento. Toda manhã, eles recebem uma coleção de tarefas de seus clientes. O dono do negócio quer executar essas tarefas em sua máquina, numa ordem que mantenha os seus clientes tão satisfeitos quanto possível. A tarefa do cliente i leva t_i unidades de tempo para ser completada. Dado um escalonamento (ou seja, uma ordem das tarefas), seja C_i o momento em que a tarefa do cliente i terminou de ser executada. Suponha ainda que cada cliente i tenha uma importância para o negócio, dada pelo número w_i. A satisfação do cliente i é dependente do tempo C_i em que sua tarefa é completada. O dono do negócio deseja determinar um escalonamento que minimize a soma ponderada $\sum_{i=1}^{n} w_i C_i$.

   Projete um algoritmo eficiente para resolver esse problema. Os dados são a duração $t_1, \ldots, t_n$ das n tarefas e a importância $w_1, \ldots, w_n$ de n clientes. Seu algoritmo deve produzir uma ordem das n tarefas que minimize a soma ponderada $\sum_{i=1}^{n} w_i C_i$. Mostre que seu algoritmo produz uma resposta correta.

   Exemplo: Considere $n = 2$, $t_1 = 1$ e $w_1 = 10$, $t_2 = 3$ e $w_2 = 2$. Se a primeira tarefa for executada primeiro, o valor da solução é $10 \cdot 1 + 2 \cdot 4 = 18$, enquanto que se a segunda tarefa for executada primeiro, é $10 \cdot 4 + 2 \cdot 3 = 46$.

8. Você foi contratado como consultor para uma companhia de caminhões que faz uma grande quantidade de entregas entre Rio e São Paulo. O volume de entregas é grande o suficiente para que a companhia tenha que enviar um número de caminhões entre as duas localidades. Os caminhões têm uma quantia máxima de peso W que eles podem carregar. Caixas chegam a São Paulo uma a uma, e cada caixa i tem um peso w_i. A estação de caminhões é pequena, então no máximo um caminhão pode estar na estação por vez. A política da companhia é que as caixas que chegam primeiro são enviadas primeiro; do contrário um cliente pode ficar chateado ao saber que uma caixa que chegou depois da dele foi entregue primeiro no Rio. No momento, a companhia está usando uma estratégia gulosa para carregar os caminhões: eles colocam no caminhão as caixas na ordem em que chegam e, quando a próxima caixa não cabe no caminhão, eles liberam o caminhão para seguir viagem.

   Mas a companhia está questionando se não estão usando muitos caminhões e querem a sua opinião sobre um possível modo de melhorar a situação. Eles estão pensando no seguinte. Talvez eles possam diminuir o número de caminhões necessários mandando algumas vezes um caminhão menos cheio de modo que os próximos caminhões estejam melhor carregados.

   Prove que, dada uma sequência de caixas com os seus pesos especificados, o método guloso correntemente em uso minimiza o número de caminhões usados. Sua prova deve seguir o tipo de análise usado na coleção máxima de intervalos disjuntos: estabeleça que a solução gulosa é ótima identificando uma medida sobre a qual o algoritmo guloso fica "sempre à frente".

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 quilómetros de uma das estações-base.

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

10. Seu amigo está trabalhando como coordenador de atividades do CEPÊ, e ele foi escolhido para organizar um mini-triatlon, onde cada participante deve nadar 20 voltas na piscina, depois correr 15km de bicicleta, e finalmente correr 5km. O plano é mandar os competidores, que não são muitos, em diferentes estágios, de acordo com a seguinte restrição: apenas uma faixa da piscina pode ser usada para o mini-triatlon, ou seja, apenas um competidor pode nadar de cada vez. O primeiro participante é liberado para nadar suas 20 voltas e, apenas quando ele termina, um segundo competidor é autorizado a nadar suas 20 voltas. Competidores podem fazer o percurso de bicicleta e a corrida ao mesmo tempo sem problemas. Apenas o uso da piscina tem essa restrição particular.

    Cada competidor tem um tempo esperado que leva para nadar suas 20 voltas, um tempo esperado para fazer 15km de bicicleta e um tempo esperado para correr 5km. Seu amigo, de posse dessa informação, deve montar um escalonamento dos competidores, que diz em que ordem os competidores iniciam sua prova, que minimize o término previsto. O término previsto da competição é o primeiro instante em que todos os competidores terminaram as três fases da prova considerando seu tempo esperado em cada prova.