Primeira lista de tarefas de grafos e otimização combinatória

Roberto Imbuzeiro Oliveira*

21 de Setembro de 2017

1 Observações

1. Ao menos por enquanto, você não precisa escrever um relatório formal sobre as tarefas abaixo.
2. A ideia é que vocês procurem fazer alguma coisa até a próxima aula e depois dela. Teremos reuniões para vocês me relatarem pessoalmente seu progresso parcial.
3. Não esqueca de anotar as fontes (livros ou online) que você eventualmente utilizar.

2 O problema do aluno de graduação

O problema desta seção é um caso típico do problema de ordenar tarefas, que têm as seguintes características:

- Há pares de tarefas (i, j) tais que i deve ser realizada antes de j.
- Tarefas que não são deste tipo podem ser realizadas em paralelo, mas há um limite para quantas podem ser realizadas ao mesmo tempo.

PROBLEMA: uma universidade tem um conjunto V_0 de n disciplinas. Cada uma destas disciplinas dura um semestre e vale $c(i) \in \mathbb{N}$ "créditos". Além disso, há disciplinas j que têm como pré-requisitos outras disciplinas i, que devem ser cursadas em um semestre anterior ao de j. Para facilitar, supomos que todas as disciplinas são oferecidas em todos os semestres e que não há "eletivas"

Queremos um algoritmo que toma como entrada um subconjunto $V \subset V_0$ de disciplinas a serem cursadas e um número L, que é um limite superior do número de créditos por semestre. A saída do algoritmo deve ser:

- Um número $k \in \mathbb{N}$ que diz quantos semestres um aluno precisaria para cursar todas as disciplinas em V.
- Para cada $t = 1, 2, \ldots, k$, o conjunto $D_t \subset V$ das disciplinas a serem cursadas no semestre t. Deve ser satisfeita a restrição do limite do número de créditos, que é:

$$\forall t \in \{1, 2, \ldots, k\} : \sum_{i \in D_t} c(i) \leq L.$$

Obviamente, gostaríamos que k fosse o menor possível, dadas as condições.

*IMPA, Rio de Janeiro, RJ, Brazil, 22430-040.

Exercício 1 Formalize o problema de maneira mais precisa. Que condições L, os números $c(i)$ e o conjunto de "relações de pré-requisito" devem satisfazer para que o problema seja solúvel?

Exercício 2 Prove que, quando $L = \infty$ (ou seja, não há limite de créditos por período), podemos encontrar um algoritmo de tempo polinomial que resolve este problema.

Exercício 3 O problema para L geral é bastante difícil. Procure um algoritmo "ótimo" para ele, ou, se não conseguir, tente uma ou duas heurísticas e veja qual se sai melhor em termos de tempo e número de semestres. *(A única restrição sobre as heurísticas é que elas devem retornar resultados corretos; fora isso, elas não precisam achar o menor k possível.)*

3 Árvores mínimas e a mudança de uma aresta

PROBLEMA: Suponha que $G = K_n$ é o grafo completo (não orientado) com n vértices e que cada aresta e tem um peso $c(e) \in [0, +\infty]$. Sabemos que os algoritmos de Prim e Kruskal ambos acham a árvore geradora de custo mínimo. Queremos saber como é que esta árvore se altera se o peso de uma das arestas e_0 é alterado.

Exercício 4 Primeiramente escolha um dos algoritmos e o implemente. Teste seu tempo de execução. De preferência, você deve escolher o algoritmo que se "sairá melhor"no exercício abaixo.

Exercício 5 Agora busque modificar seu algoritmo de modo que, se você altera uma aresta, ele acha a nova árvore geradora no *menor tempo possível*. (Isso pode requerer armazenar mais dados sobre a árvore quando ela é calculada.) Avalie a complexidade do seu método.