

Trabalho - Algoritmos em Grafos - 2019.2

Universidade Federal do Ceará - Campus Sobral

Curso de Engenharia da Computação

Professor: Antonio Josefran

1 Proposta

O trabalho descrito substituirá a nota da última AP e consiste em duas partes, descritas abaixo. Cada parte vale metade da nota do trabalho.

1.1 Parte 1

Implementar os algoritmos de menores caminhos dentre todos os pares de vértices apresentados em sala. Cada algoritmo deve imprimir a matriz de menores caminhos e os menores a partir de vértice fixo fornecido como entrada. Caso seu grafo tenha circuito negativo, seu algoritmo deverá ser capaz de identificar a sua existência.

Por exemplo, se for passado como entrada o seguinte arquivo (o formato está explicado na Seção Formatação de Entrada)

```
3
0
0 1 1
0 2 1
```

a saída do seu algoritmo deverá ser a matriz

$$\begin{vmatrix} 0 & 1 & 1 \\ \infty & 0 & \infty \\ \infty & \infty & 0 \end{vmatrix}$$

seguido dos menores caminhos do vértice 0 ao restante dos vértices, i.e.,

$$0- > 1$$

$$0- > 2.$$

1.2 Parte 2

Considere os seguintes algoritmos para resolver o problema de fluxo máximo:

- Ford-Fulkerson;
- Push-relabel maximum flow algorithm.

Implemente os algoritmos acima. Utilizando a ideia por trás de cada algoritmo acima, implemente uma forma eficiente, para cada algoritmo, de encontrar o corte mínimo do grafo fornecido. Seu algoritmo deverá receber um grafo orientado com vértices fonte e sorvedouro explicitados e então retornar a função fluxo máximo para o grafo e as arestas que pertencem ao corte máximo.

Por fim, enuncie e prove o Teorema de Fluxo Máximo Corte Mínimo.

2 Formato de Entrada dos Algoritmos

Os algoritmos deverão receber um arquivo no formato *txt* da seguinte forma.

- A primeira linha fornece o número de vértices do grafo;
- A segunda linha varia conforme o problema
 - Problema de Caminhos Mínimos: consiste em um único vértice explicitando qual vértice deverá ser impresso os menores caminhos;
 - Problema Fluxo: consiste em um par de vértices separados por espaço onde o primeiro vértice é a fonte e o seguindo o sorvedouro;
- A linhas seguintes são compostas de dois inteiros e um valor real que representam uma aresta e seu peso. Por exemplo a tripla $i\ j\ s$ diz que o grafo possui uma aresta de i para j com peso s ;
- Todas as arestas que não forem listadas acima não existem no grafo.

3 Linguagem de Programação e outros detalhes de implementação

Poderá ser utilizada qualquer linguagem de programação. Porém, é terminantemente proibido utilizar qualquer biblioteca de grafos. Todas as rotinas de grafos terão que ser implementadas pelo aluno. As mesmas rotinas poderão ser aproveitadas em todo o trabalho.

4 Entrega do Trabalho

O trabalho deverá ser entregue até as 23:59 do dia 29/11/2019 através do e-mail josefran@ufc.br.

O trabalho deverá ser enviado em um arquivo zipado contendo os seguintes arquivos.

- Um pdf contendo os pseudos códigos comentados dos algoritmos, suas complexidades e um resumo do funcionamento de cada algoritmo implementado. Em particular, cada algoritmo deverá conter um exemplo não trivial de sua execução.
- Uma pasta para cada algoritmo contendo os arquivos necessários para compilação daquele algoritmo e um arquivo README.txt explicando como utilizar cada algoritmo.

5 Correção e Considerações Finais

O trabalho vale de 0 a 10 e pode ser feito em equipe de no máximo 3 alunos. O trabalho será corrigido em sistema operacional Ubuntu 18.04. O descumprimento de qualquer um dos requisitos pedidos aqui implicará na rejeição do trabalho e portanto a nota será zerada. Em particular, algoritmos que não compilarem terão suas respectivas notas zeradas.

A nota será igualmente dividida por algoritmo e o Teorema pedido. E, para cada algoritmo, a nota será dividida igualmente entre o que foi pedido.