

# Atividade de Implementação 2

GCC218 - Algoritmos em Grafos

Mayron Moreira - DCC/UFLA

22 de outubro de 2018

## Problema 1

Adamantina é uma cidade modelo pela sua rede de transportes. Isso porque seus arquitetos e urbanistas decidiram que qualquer ponto da cidade poderia ser alcançável a partir de outro por meio de um automóvel (em ambos os sentidos). Os estabelecimentos comerciais e residências de Adamantina estão localizados nos quarteirões, e cada quarteirão conecta duas intersecções (esquinas). As ruas são de mão dupla. Dadas duas intersecções  $a$  e  $b$  em Adamantina, a distância entre  $a$  e  $b$  é dada pelo número mínimo de quarteirões que devem ser percorridos de  $a$  para  $b$ . A Figura 1 ilustra esta ideia a partir de uma parte da cidade (as intersecções estão marcadas por um círculo vermelho e os blocos por quadrados pretos).

Recentemente, devido a trechos carregados de intenso tráfego de veículos, os engenheiros da prefeitura sugeriram uma reformulação no *layout* de sua rede de transportes. A ideia é transformar algumas ruas de mão dupla em ruas de mão única. Esta proposta deve ser planejada de maneira cautelosa, pois o prefeito atual não quer perder em hipótese alguma a acessibilidade. Por outro lado, mesmo que o alcance seja mantido, é possível que as distâncias entre intersecções específicas possam ser significativamente aumentadas. A Figura 2 apresenta um exemplo de alteração sugerida.

Após diversas assembléias (inclusive com participação popular), a prefeitura decidiu aceitar as propostas que aumentem as distâncias entre quaisquer intersecções em um fator  $x$ , acrescido de uma constante  $y$ . Por exemplo: se as intersecções  $u$  e  $v$  possuem distância  $d_{uv}$ , então com a nova proposta, a distância não poderá passar de  $xd_{uv} + y$ .

A sua empresa, denominada *GrafUFLA*, foi contemplada no processo licitatório em Adamantina e agora terá que encontrar uma solução para este problema. Assim, desenvolva um algoritmo polinomial que auxilie a prefeitura nesta tomada de decisão.

## Entrada dos dados

- A primeira linha contém o número de intersecções da cidade ( $n$ );
- As próximas  $n$  linhas contém, em cada linha, a intersecção e uma sequência de outras intersecções que ligam à primeira;
- As próximas  $n$  linhas contém, em cada linha, a intersecção e uma sequência de novas intersecções que ligam à primeira;

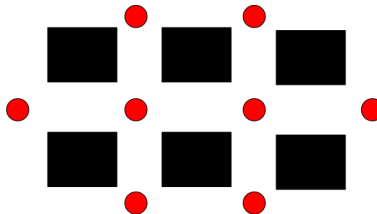


Figure 1: Exemplo de uma parte da cidade de Adamantina.

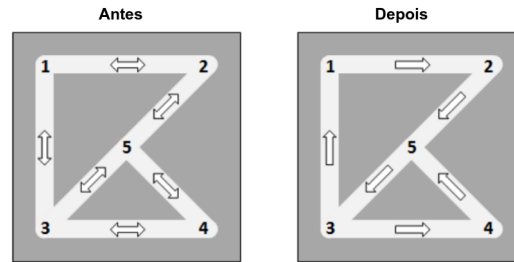


Figure 2: Exemplo de alteração no tráfego urbano de Adamantina.

- A última linha contém os valores de  $x$  e  $y$ .

- **Exemplo 1:**

```

5
1 2 3
2 1 5
3 4 5 1
4 3 5
5 2 3 4
1 2
2 5
3 1 4
4 5
5 3
1 2

```

- **Exemplo 2:**

```

5
1 2 3
2 1 5
3 4 5 1
4 3 5
5 2 3 4
1 2
2 5
3 1 4
4 5
5 3
2 0

```

- **Exemplo 3:**

```

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

```

## Saída dos dados

A saída de dados imprimirá “Sim” se o plano proposto obedece todas as restrições do problema, e “Não”, caso contrário.

- **Exemplo 1:**

Sim

- **Exemplo 2:**

Não

- **Exemplo 3:**

Sim

## Problema 2

A *GrafUFLA* foi novamente contratada para a resolução de um problema no contexto de transportes. A CPTM (Companhia Paulista de Trens Metropolitanos) pretende criar um aplicativo que auxilie o usuário a planejar suas rotas de maneira eficiente na região metropolitana de São Paulo. A ideia é tentar evitar atrasos de passageiros e fazer com que os mesmos cheguem o mais rápido possível no destino. No entanto, uma pesquisa de opinião mostrou que a maioria dos passageiros não gosta de chegar tão cedo a estação de trens. Assim, se existe mais de uma opção de horário de saída que tenha o mesmo horário de chegada, o usuário desejará escolher a opção saída mais tarde. A *GrafUFLA*, através de sua *expertise*, deverá implementar uma solução para este problema. Assim, dada a rede de transportes da CPTM e um conjunto de horários de partida de trens, sua solução deve encontrar a conexão de trens com o horário de chegada ao destino final mais cedo possível, dada uma origem<sup>1</sup>.

## Entrada dos dados

```
#qtdeCidades (N)
nomeCidade1
nomeCidade2
...
nomeCidadeN
numeroTrens (T)
InfoTrem1
NumeroPartidasTrem1
HoraPartida1 Cidade1
HoraPartida2 Cidade2
...
InfoTrem2
NumeroPartidasTrem2
HoraPartida1 Cidade1 (não necessariamente a mesma da anterior)
HoraPartida2 Cidade2 (não necessariamente a mesma da anterior)
...
InfoTremT
NumeroPartidasTremT
HoraPartida1 Cidade1 (não necessariamente a mesma da anterior)
HoraPartida2 Cidade2 (não necessariamente a mesma da anterior)
...
```

---

<sup>1</sup>Vamos desconsiderar o tempo de troca de trens.

Hora mais cedo possível que um passageiro sai  
Cidade Origem  
Cidade Destino

- **Exemplo 1:**

3  
Itapevi  
Grajau  
Guaianases  
3  
2  
0900 Itapevi  
1200 Grajau  
2  
1200 Grajau  
2200 Guaianases  
3  
0900 Itapevi  
1200 Grajau  
2300 Guaianases  
0800  
Itapevi  
Guaianases

- **Exemplo 2:**

3  
Itapevi  
Grajau  
Guaianases  
1  
3  
0900 Itapevi  
1200 Grajau  
2300 Guaianases  
1000  
Itapevi  
Guaianases

## Saída dos dados

Se uma solução existir, o programa deve imprimir o horário de partida e o horário de chegada. Em caso de não haver solução, o programa deverá imprimir “Sem solução viável”.

- **Exemplo 1:**

0900 2200

- **Exemplo 2:**

(Descubra a solução!)

## Regras para a entrega

- Os alunos deverão se dividir em grupos de 2 ou 3 pessoas.

- O algoritmo deverá ser implementado em C++ ou Python.
- A forma de se representar o Grafo poderá ser livremente escolhida por cada grupo: (a) lista de adjacências; (b) matriz de adjacências.
- **Data e hora de entrega: 06/11/2018, até às 23h55.**
- **Qualquer constatação de cópia ou plágio de trabalhos acarretará em nulidade das notas de todos os membros dos grupos envolvidos.**
- O *upload* do código fonte referente a este trabalho deve ser feito no sistema Moodle, na sala da disciplina “Algoritmos em Grafos (10A/14A)”, em local devidamente especificado.
- Apenas um dos alunos do grupo deve depositar o conteúdo do trabalho do sistema. Para tanto, a pasta compactada com todos os arquivos deve ter o seguinte formato:
- matriculaAluno1 matriculaAluno2 matriculaAluno3.zip
- Será descontado uma determinada pontuação para os membros de cada grupo cujo trabalho não seguir as especificações estabelecidas neste documento.
- Soluções do tipo força bruta não serão consideradas.
- Trabalhos entregues após a data limite não serão aceitos.
- Clareza e organização do código fonte serão itens levados em conta na avaliação. Trabalhos individuais terão desconto de nota.
- A nota desta atividade será completamente distribuída após uma entrevista referente a tarefa, com os membros do grupo.