

---

Aluno:	Matrícula:	Valor: 20,0	Nota:
--------	------------	-------------	-------

---

## Trabalho I

### 1. Objetivos.

- Desenvolver a habilidade de programação de algoritmos em grafos.
- Reforçar o aprendizado sobre os algoritmos para o problema do caminho mínimo.
- Avaliar o impacto da complexidade computacional no tempo de execução.

### 2. Descrição.

O trabalho consiste em implementar algoritmos para o problema do caminho mínimo em grafos. Os algoritmos a serem desenvolvidos são: Dijkstra, Bellman-Ford e Floyd-Warshall. É de livre escolha a forma de representação computacional para os grafos em cada algoritmo (veja que essa escolha terá implicações nos tempos de execução e uso de memória).

Adicionalmente, deve-se gravar um vídeo de até 10 minutos descrevendo detalhadamente sua implementação e os resultados observados para um conjunto de arquivos de teste. Sugiro o uso da ferramenta Google Meet para tal. Para assegurar a autoria do vídeo, é solicitado que o mesmo seja gravado com a câmera ligada mostrando o(s) autor(es) do trabalho.

O seu programa deve solicitar ao usuário o nome do arquivo contendo o grafo (no formato Dimacs), o algoritmo a ser executado e os vértices de origem e de destino  $s$  e  $t$ . Como saída, seu programa deve informar a sequência de vértices do caminho mínimo de  $s$  a  $t$ , bem como o custo do caminho e o tempo de execução (em segundos) do algoritmo. Ex.:

```
Informe o arquivo: <toy.txt>
```

```
Algoritmo:
```

- 1 Dijkstra
- 2 Bellman-Ford
- 3 Floyd-Warshall

```
<1>
```

```
Origem: <0>
```

```
Destino: <3>
```

```
Processando...
```

```
Caminho: [0, 1, 2, 3]
```

```
Custo: 5
```

```
Tempo: 0.003s
```

### 3. Teste dos algoritmos.

Cada trabalho deverá validar sua implementação sobre os conjuntos de dados disponíveis no arquivo “Datasets.zip”. Esses conjuntos de dados são provenientes de aplicações reais, tais como locomoção em grandes cidades e conexões do Facebook. A tabela abaixo apresenta uma breve descrição sobre cada grafo de teste:

Arquivo	Descrição
toy.txt	Exemplo dado em aula, que pode ser checado manualmente.
rg300_4730.txt	Grafo aleatório com 300 vértices e 4730 arestas.
rome99c.txt	Descreve um mapa das estradas de Roma em 1999.
facebook_combined.txt	Conexões entre contatos do Facebook (com 4039 usuários).
USA-road-dt.DC.txt	Descreve um mapa das estradas de Washington DC.

Sugiro utilizar a instância “toy” para testar a corretude dos algoritmos visto que é fácil verificar seu resultado manualmente.

### 4. Relatório

Marque abaixo o tempo de execução (em segundos) e o custo do caminho mínimo encontrado entre os vértices 0 e 100 <sup>1</sup>.

Arquivo	Dijkstra		Bellman-Ford		Floyd-Warshall	
	Tempo	Custo	Tempo	Custo	Tempo	Custo
rg300_4730.txt						
rome99c.txt						
facebook_connections.txt						
USA-road-dt.DC.txt						

Marque abaixo o caminho mínimo (sequência de vértices a ser tomada) encontrado.

Arquivo	Caminho Mínimo
rg300_4730.txt	
rome99c.txt	
facebook_connections.txt	
USA-road-dt.DC.txt	

Liste abaixo suas principais conclusões sobre o desempenho dos algoritmos.

---

<sup>1</sup>Caso o algoritmo não tenha encerrado após 1000 segundos de execução, marque TO (*time-out*), caso não tenha sido possível carregar o grafo em memória, marque MO (*memory-out*)

## 5. Avaliação.

O trabalho deverá ser feito **individualmente ou em dupla** e enviado via Moodle (código, relatório e apresentação) até as 23:59h do dia 19/11/21. A distribuição dos pontos será a seguinte:

- Corretude e eficiência das implementações dos algoritmos - 10 pontos.
- Qualidade e segurança da apresentação - 10 pontos.

Bom trabalho!