



Universidade Federal Fluminense
Curso: Ciência da Computação
Disciplina: Algoritmos em Grafos
Professor: Luís Felipe

Nome: _____

Atividade em Sala 3

1. Considere um dígrafo ponderado $G = (V, E)$ com n vértices numerados de 0 a $n - 1$, sem ciclos negativos.

Implemente uma adaptação do Algoritmo de Floyd-Warshall que retorne:

- a matriz D , em que $D[i][j]$ armazena a distância mínima de i até j ;
- a matriz N , em que $N[i][j]$ armazena o próximo vértice após i em um caminho mínimo de i até j .

Além disso, implemente uma função

`obter_caminho(N, i, j)`

que retorna uma lista com os vértices de um caminho mínimo de i até j .

Caso não exista caminho de i até j , a função deve retornar uma lista vazia.

Também explique como a matriz N deve ser atualizada sempre que uma distância $D[i][j]$ for melhorada usando um vértice intermediário k .

2. Considere um dígrafo ponderado $G = (V, E)$, com pesos nas arestas dados por uma função W , e um vértice raiz r .

Implemente, em Python, uma adaptação do Algoritmo de Bellman-Ford que, além de detectar a existência de ciclo negativo alcançável a partir de r , também retorne as arestas de um ciclo negativo, caso ele exista.

A função deve receber como entrada:

- a lista de adjacências `adj`;
- o dicionário de pesos `W`;
- o vértice raiz `r`.

A função deve retornar:

- `L`, `pai`, `False` e uma lista vazia, caso não exista ciclo negativo;
- `None`, `pai`, `True` e uma lista contendo as arestas de um ciclo negativo, caso exista.