

Projeto prático 4

SCC 0224 – Estrutura de Dados II
Prof. Maria Cristina Ferreira de Oliveira
10 de junho de 2024

1. Descrição

A melhor época do ano começou, e com ela, as festas juninas! João gosta muito de viajar pelo Brasil em busca de conhecer festas juninas de cidades diferentes. Uma dificuldade é que os custos das viagens são muito altos, mas a boa notícia é que, no caso de algumas cidades, o João é pago para viajar! João lhe passou uma lista de conexões entre as cidades, com o custo estimado da viagem (inclui pedágio, combustível, alimentação, etc.). Caso seja um trajeto em que ele ganha, ao invés de gastar, ele informa um custo negativo, do contrário o custo informado é positivo. João pede que você calcule qual é o custo mínimo para viajar entre todos os pares de cidades que ele informar.

As cidades são representadas por números inteiros de 0 a $N-1$, o custo estimado entre um par de cidades que se conectam é um número inteiro D , só é possível ir de uma cidade a outra na ordem que ela for dada na entrada (o grafo é direcional).

Será passado o algoritmo T que você deve usar para calcular o menor custo entre os pares de cidades: se $T = 1$, utilize o algoritmo de *Dijkstra*, se $T = 2$, utilize o algoritmo de *Floyd-Warshall*. Eles possuem as seguintes restrições de entrada:

Se $T = 1$:

$2 \leq N \leq 1000$

$1 \leq D \leq 10000$

Se $T = 2$:

$2 \leq N \leq 200$

$-10000 \leq D \leq 10000$

2. Entrada e Saída

Primeiramente, seu programa receberá o valor T ; em seguida dois inteiros N e M , que informam, respectivamente, o número de cidades e o número de conexões.

As próximas M linhas contêm os inteiros X_i , Y_i , D_i , $1 \leq i \leq M$; representando a estrada i entre as cidades X_i e Y_i , cujo custo associado estimado é D_i .

Em seguida, será passado o inteiro K , e as próximas K linhas informam os valores X_k e Y_k , $1 \leq k \leq K$, que identificam um par de cidades para o qual João deseja saber o custo mínimo. Na saída imprima para cada linha da entrada o custo mínimo

possível para chegar na cidade Yk a partir da cidade Xk, imprima -1 caso não exista uma maneira de chegar até Yk a partir de Xk.

3. Conteúdo e data de entrega

O trabalho é individual e a data limite de submissão é **30/06/2024** (trabalhos em atraso não serão aceitos), no sistema RunCodes (<https://runcodes.icmc.usp.br/>). O formato da entrega deve ser um arquivo com o código fonte desenvolvido.

- O código deverá estar adequadamente comentado/documentado;
- O trabalho pode ser desenvolvido utilizando C ou Python, a escolha;
- Não é permitido utilizar bibliotecas que já disponibilizam os algoritmos de forma parcial ou total, o que inclui bibliotecas com algoritmos de ordenação;
- Todo o código deve ser desenvolvido pelo próprio aluno. Os códigos serão submetidos a um detector de plágio, e a utilização de código criado por terceiros será tratada como plágio (nota zero).

Dica: caso seu programa esteja gerando alguma saída errada, mas está difícil identificar a diferença entre a saída esperada e a resultante, este site para comparação de textos pode ser útil: <https://www.diffchecker.com/text-compare/>

Dica 2: se quiser visualizar melhor os grafos utilize o site: https://csacademy.com/app/graph_editor/

4. Critérios de avaliação

- 9,0 pontos pela implementação dos algoritmos solicitados, de forma correta (avaliado segundo o resultado do RunCodes, e por inspeção do código).
- 1,0 ponto por código legível, bem estruturado e bem comentado.

5. Exemplos de entrada e saída

Entrada	Saída
1 6 6 0 1 20 0 2 50 0 3 35 0 4 2 1 2 60 1 3 10 4 0 3 0 2 4 1 2 5	30 50 -1 -1

Entrada	Saída
2 6 6 0 1 20 0 2 50 0 3 35 0 4 2 1 2 60 1 3 10 4 0 3 0 2 4 1 2 5	30 50 -1 -1

Entrada	Saída
2 6 6 0 1 20 0 3 35 0 4 -2 1 2 -60 1 3 10 2 0 50 4 0 3 0 2 4 1 2 5	30 -40 -1 -1

Entrada	Saída
2 7 11 0 1 1 0 2 1 2 1 1 1 3 4 3 2 -2 3 4 -1 3 5 1 4 5 1 4 6 3 4 5 1 5 6 1 6 0 6 0 4 1 5 3 0 3 1 4 6	6 4 4 -1 -1 2

Entrada	Salida
2	-1
5 10	3
0 1 6	0
0 2 3	-5
0 3 11	-1
0 4 20	-1
2 1 10	-8
1 3 2	-5
4 1 4	
2 3 -3	
2 4 4	
3 4 -5	
8	
0 1	
0 2	
0 3	
0 4	
1 0	
2 0	
2 4	
3 4	