

Trabalho 3 – Caminhos mínimos: Central Telefônica

Prazo de entrega: 31/05/2014 no SSP

Uma empresa de telecomunicações está desenvolvendo um algoritmo para estabelecer conexões entre seus assinantes. Todo o *hardware* (pontos de repetição e cabeamento) já foi configurado. Você foi contratado para implementar esse software.

Os cabos conectam dois pontos de repetição e tem velocidades de transferência (*Bandwidth* – **BW**) diferentes. Dado um percurso **P**, composto por pontos de repetição **i**, **j**, ... , **k**, a *Bandwidth* do percurso será $\min\{ BW(i), BW(j), \dots, BW(k) \}$.

Dado que o usuário forneça o “grafo” ponderado não negativo e os pontos de repetição **P_i** e **P_j**, retorne a **BW** do percurso.

Modifique o algoritmo Dijkstra para resolver esse problema.

Entrada

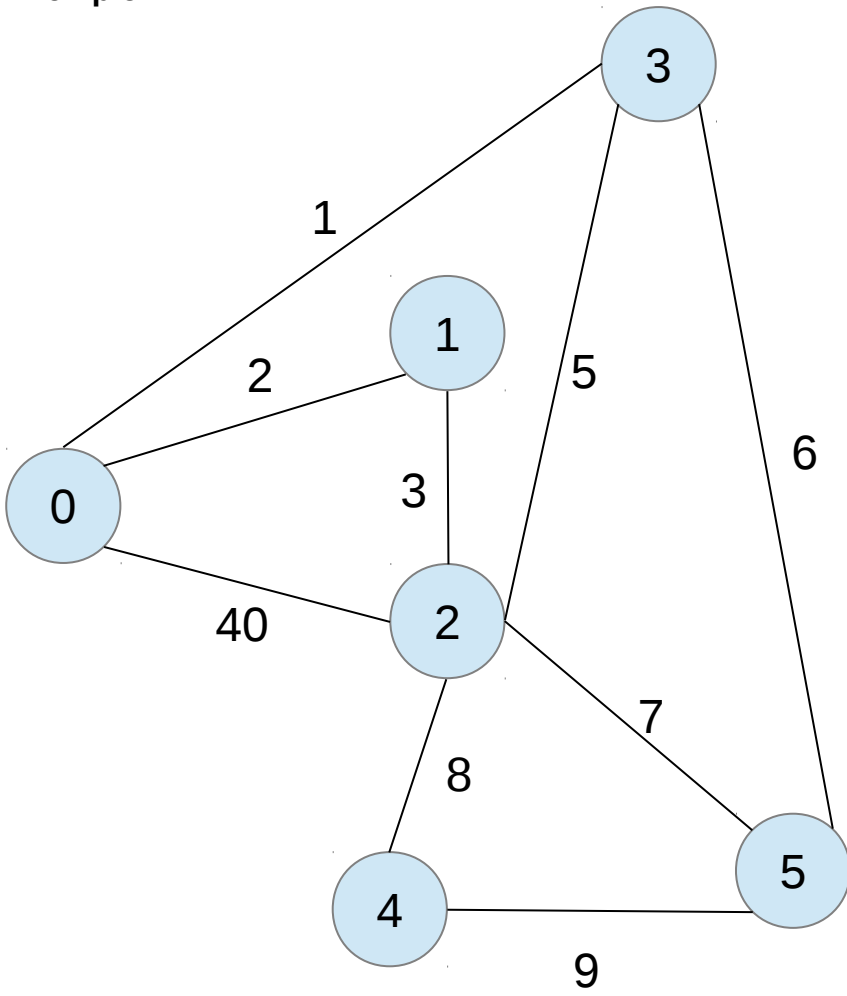
Os dados – fornecidos na entrada-padrão – contém apenas um caso de teste, que consiste nas informações do grafo de entrada.

A primeira linha de entrada contém dois números inteiros, **V** e **A**, que indicam respectivamente o número de Vértices (pontos de repetição) e o número de Arestas (cabos) do grafo.

As **A** linhas seguintes contém dois números inteiros, **V1**, **V2** e **BW**, vértices que devem ser conectados no grafo (de **V1** para **V2**) com peso (velocidade) **BW**.

A próxima linha contém dois inteiros, **P_i** e **P_j**, que representam os pontos de repetição que se deseja conectar.

Exemplo



- 6 9
- 0 3 1
- 0 1 2
- 2 1 3
- 0 2 40
- 2 3 5
- 3 5 6
- 2 5 7
- 2 4 8
- 4 5 9
- 0 3

Saída

A saída contém apenas um número: a velocidade de transferência (BW) do percurso ótimo encontrado.

Terminal:

6

Dicas

- A velocidade de transferência de um ponto para ele mesmo é infinita;
- Você deve modificar a fila de prioridades (*heap* máxima / mínima) para resolver o problema.