
Weighted shortest path (5)**P68936_en**

Write a program that, given a directed graph with positive and/or negative costs at the arcs (but no negative cycles), and two vertices x and y , computes the minimum cost to go from x to y .

Input

Input consists of several cases. Every case begins with the number of vertices n and the number of arcs m . Follow m triples u, v, c , indicating that there is an arc $u \rightarrow v$ of cost c , where $u \neq v$, $-1000 \leq c \leq 1000$ and $c \neq 0$. Finally, we have x and y . Assume $1 \leq n \leq 10^4$, $0 \leq m \leq 5n$, and that for every pair of vertices u and v there is at most one arc of the kind $u \rightarrow v$. All numbers are integers. Vertices are numbered from 0 to $n - 1$. The directed graph has no negative cycles.

Output

For every case, print the minimum cost to go from x to y , if this is possible. If there is no path from x to y , state so.

Sample input

```
6 10
  1 0 6
  1 5 15
  3 4 3
  3 1 8
  4 0 20
  0 5 5
  0 2 1
  5 1 10
  4 1 2
  2 3 4
3 5
2 1
  0 1 1000
1 0
8 11
  0 1 10
  0 7 8
  1 5 2
  2 1 1
  2 3 1
  3 4 3
  4 5 -1
  5 2 -2
  6 5 -1
  6 1 -4
  7 6 1
0 1
```

Sample output

```
16
no path from 1 to 0
5
```

Problem information

Author : Jordi Petit

Generation : 2018-06-13 15:47:12

© *Jutge.org*, 2006–2018.

<https://jutge.org>