BYTEOTIA2

Introduction:

After the latest disease outbreak in Byteotia was over, it was decided that the road structure should be converted something more secure. After months of work, the road structure between the **N** cities of Byteotia was converted such that the roads were now *directed*, and it was impossible to get back to a city you visited using this new road system after leaving the city (meaning there were no cycles in the country – it was *acyclic*).

However, this new introduction of roads proved to be quite cumbersome; the most annoying part was that the distance between some cities were too long. For this reason, the king of Byteotia hired a professional programmer to calculate the longest distance between two given cities.

Input Format:

• In the first line, four space separated integers; the number of cities **N** (2<=**N**<=100 000), the number of directed roads **M** (2<=**M**<=1 000 000), the starting city **s** (1<=**s**<=**N**) and the destination city **t** (1<=**t**<=**N**).

• In each of the following **M** lines, three integers $\mathbf{a_i}$, $\mathbf{b_i}$ (1<= $\mathbf{a_i}$, $\mathbf{b_i}$ <= **N**) and $\mathbf{d_i}$ (1<=$\mathbf{d_i}$<=1000) describing a single directional road of length $\mathbf{d_i}$ from $\mathbf{a_i}$ to $\mathbf{b_i}$.

Output Format:

• In a single line; the longest distance from **s** to **t**. This line should contain **-1** if **t** can not be reached from **s**.

Example Input/Output:

Input

9 13 1 9
1 2 5
1 3 10
1 4 15
2 5 3
2 6 12
3 6 7
3 7 14
4 7 5
4 8 1
5 9 20
6 9 10
7 9 5
8 9 8

Output

29
//The longest road from 1 to 9 is from 1→3→7→9.

Input

5 4 5 1
1 2 1
2 3 1
3 4 1
4 5 1

Output

-1

No roads exist from 5 to 1, so we output -1.