Problem F. Flow

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	256 mebibytes

We are considering a maximum flow problem on an infinite network.

You are given a bipartite graph G with n vertices in both parts and m directed edges. Each edge goes from the left part to the right part, and has its capacity specified. We want to construct a family of networks $\{F_k\}$.

Here are the steps to construct the network F_k.

- We first produce k copies of the graph G. We call these copies $G_1, G_2, \ldots, G_k$.
- For all $1 \le i \le k-1$ and $1 \le u \le n$, we add a directed edge from u-th vertex in the right part of G_i to u-th vertex in the left part of G_{i+1} with infinite capacity.
- We add directed edges from the source to all the vertices in the left part of G_1 with infinite capacity.
- We add directed edges from all the vertices in the right part of G_k to the sink with infinite capacity.

Let f_k be the maximum flow in the network F_k.

We want to know what the sequence $\{f_k\}$ looks like when k goes to infinity. If $\{f_k\}$ does not converge to a constant, output -1. Otherwise, output $\lim\limits_{k\to+\infty} f_k$.

Input

The first line contains two integers n and m ($1 \le n \le 2000$, $1 \le m \le 4000$).

Each of the following m lines contains three integers u, v, and w ($1 \le u, v \le n$, $1 \le w \le 10^5$) which indicate that there is a directed edge from the u-th vertex in the left part to the v-th vertex in the right part with capacity w.

Output

If the sequence $\{f_k\}$ does not converge to a constant, output -1. Otherwise, output $\lim\limits_{k\to+\infty} f_k$.

Example

standard input	standard output
5 5 1 2 3 2 3 4 3 1 2 4 5 6 5 4 3	12