



Oski's Network Bottleneck

Problem Statement

Oski is setting up a high-speed server network across N data centers, numbered 1 through N . The network is structured as a Directed Acyclic Graph (DAG), where data flows strictly from the central headquarters at data center 1 to the primary distribution hub at data center N .

The network contains M directed optical links. Each directed link $e = (u, v)$ connecting data center u to data center v has two parameters:

- **Cost (c_e):** The operational cost to route data through this link.
- **Bandwidth Capacity (w_e):** The maximum bandwidth capacity supported by this link.

For a data transmission route (path P) from data center 1 to data center N :

- The **Total Cost** $C(P)$ is the sum of costs of all links along the path:

$$C(P) = \sum_{e \in P} c_e$$

- The **Effective Bottleneck Bandwidth** $B(P)$ is the minimum bandwidth capacity among all links along the path:

$$B(P) = \min_{e \in P} w_e$$

Oski has Q potential operational budgets. For each budget K , he needs to select a valid path P from 1 to N whose total cost does not exceed K ($C(P) \leq K$), such that the effective bottleneck bandwidth $B(P)$ is maximized.

Help Oski answer all Q budget queries efficiently!

Input Format

- The first line contains two integers N and M ($2 \leq N \leq 500$, $1 \leq M \leq 2000$) — the number of data centers and optical links.
- The next M lines each contain four integers u_i, v_i, c_i, w_i ($1 \leq u_i, v_i \leq N$, $1 \leq c_i \leq 100$, $1 \leq w_i \leq 10^9$), describing a directed optical link from u_i to v_i with cost c_i and bandwidth capacity w_i . It is guaranteed that the graph is a DAG without multi-edges.
- The next line contains an integer Q ($1 \leq Q \leq 5 \cdot 10^4$) — the number of queries.
- The next Q lines each contain a single integer K ($0 \leq K \leq 5 \cdot 10^4$) — the budget constraint for each query.

Output Format

For each query, print the maximum bottleneck bandwidth of a path from 1 to N with total cost at most K . If no such path exists, print -1 .

Sample Input 1

```
4 5
1 2 2 10
1 3 5 20
2 4 3 15
3 4 1 8
1 4 10 5
3
5
6
3
```

Sample Output 1

```
10
10
-1
```

Explanation of Sample 1

The available paths from 1 to 4 are:

1. $P_1 = 1 \rightarrow 2 \rightarrow 4$: Total Cost = $2 + 3 = 5$, Bottleneck = $\min(10, 15) = 10$.
2. $P_2 = 1 \rightarrow 3 \rightarrow 4$: Total Cost = $5 + 1 = 6$, Bottleneck = $\min(20, 8) = 8$.
3. $P_3 = 1 \rightarrow 4$: Total Cost = 10, Bottleneck = 5.

Queries:

- **Query 1** ($K = 5$): Valid path P_1 has cost $5 \leq 5$. Maximum bottleneck = 10.
- **Query 2** ($K = 6$): Valid paths are P_1 (cost $5 \leq 6$, bottleneck 10) and P_2 (cost $6 \leq 6$, bottleneck 8). Maximum bottleneck = $\max(10, 8) = 10$.
- **Query 3** ($K = 3$): No path has cost ≤ 3 . Output -1 .