Topic 7: Network Flow: Min-Cut, Min-Cost

CS 41100, CP3 Competitive Programming III (Spring 2025) **Zhongtang Luo**
Purdue University March 10, 2025

Learning Objectives

The students will be able to…

1. describe the **max-flow network flow problem** and the **min-cost network flow problem**;
2. describe the **max-flow min-cut theorem**;
3. apply well-established algorithms (**ISAP**, **Dinic**, **EK**, **ZKW**, etc.) to solve a network flow problem;
4. model programming problems (e.g. a **matching problem**) with network flow.

Sample Problems

Problem Name: Magic Potion
Link: `https://vjudge.net/problem/Gym-101981I`

Problem Name: Kejin Game
Link: `https://vjudge.net/problem/UVALive-7264`

Problem Name: Coding Contest
Link: `https://vjudge.net/problem/HDU-5988`

Problem Name: Hiring Employees
Link: `https://dmoj.ca/problem/noi08p3`

Magic Potion

There are n heroes and m monsters living on an island. The monsters have become very vicious recently, so the heroes have decided to reduce their numbers. However, the i-th hero can only kill one monster from the set M_i. Joe, the strategist, has k bottles of magic potion, each of which can buff one hero's power, allowing him to kill one additional monster. Since the potion is very powerful, a hero can only take at most one bottle of potion.

Please help Joe determine the maximum number of monsters that can be killed by the heroes if he uses the optimal strategy.

Input

The first line contains three integers n, m, k ($1 \leq n, m, k \leq 500$) — the number of heroes, the number of monsters, and the number of bottles of potion.

Each of the next n lines contains one integer t_i, the size of M_i, followed by t_i integers $M_{i,j}$ ($1 \leq j \leq t_i$), the indices (1-based) of monsters that can be killed by the i-th hero ($1 \leq t_i \leq m$, $1 \leq M_{i,j} \leq m$).

Output

Print the maximum number of monsters that can be killed by the heroes.

Examples

Input

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

Output

```
4
```

Input

```
5 10 2
2 3 10
5 1 3 4 6 10
5 3 4 6 8 9
3 1 9 10
5 1 3 6 7 10
```

Output

```
7
```

Source

2018 ACM-ICPC Asia Nanjing Regional Programming Contest

Kejin Game

In recent years, many free-to-play games, referred to as Kejin games, have emerged. These games are accessible without charge, but specific items or characters require payment. Examples include Love Live, Kankore, Puzzle & Dragon, Touken Ranbu, and Kakusansei Million Arthur, all of which have gained immense popularity and generate significant revenue daily.

In a Kejin game, your character possesses a skill graph that determines how skills can be acquired. This graph is a directed acyclic graph where vertices represent skills, and edges indicate dependencies: if there is an edge from skill A to skill B, A is a prerequisite for B. In cases where skill S has multiple dependencies, all must be acquired before obtaining S. Furthermore, each edge in the graph is unique, and cyclic dependencies are absent.

Acquiring skills typically involves time and effort, especially for advanced skills deeper in the graph. However, as a player with resources to spare, you can choose to pay money, denoted as "Ke," to bypass certain restrictions. Specifically, money can be used to remove edges from the dependency graph or to acquire skills directly, ignoring existing dependencies.

Given the constraints of time and money, you wish to optimize the balance between them. Each action, be it acquiring a skill through conventional means, removing an edge, or directly purchasing a skill, incurs a cost measured in units of "TA." You seek to determine the minimal cost required to acquire a desired skill S, starting without any initial skills.

Input

The input starts with an integer indicating the number of test cases (no more than 10). Each test case consists of:

- The first line containing three integers N, M, and S, where N is the number of vertices (skills), M is the number of arcs (dependencies), and S is the index of the target skill (1-based index).
- M subsequent lines each contain three integers A, B, and C, where there is an arc from skill A to skill B with C TAs cost to remove the dependency.
- A line with N integers representing the cost to acquire each skill through normal means.
- Another line with N integers representing the cost to directly acquire each skill via payment, bypassing any dependencies.

The costs for acquiring skills or removing dependencies range up to $1,000,000$.

Output

For each test case, output a single line containing the minimal cost to acquire the specified skill S.

Examples

Input

```
2
5 5 5
1 2 5
1 3 5
2 4 8
4 5 10
3 5 15
3 5 7 9 11
100 100 100 200 200
5 5 5
1 2 5
1 3 5
2 4 8
4 5 10
3 5 15
3 5 7 9 11
5 5 5 50 50
```

Output

```
31
26
```

Source

2015 ACM-ICPC Asia Beijing Regional Contest

Coding Contest

A coding contest will be held in this university, in a huge playground. The whole playground will be divided into N blocks, and there will be M directed paths linking these blocks. The i-th path goes from the u_i-th block to the v_i-th block. Your task is to solve the lunch issue. According to the arrangement, there are s_i competitors in the i-th block. Limited to the size of table, b_i bags of lunch including breads, sausages, and milk would be put in the i-th block. As a result, some competitors need to move to another block to access lunch.

However, the playground is temporary, and as a result there would be many wires on the path. For the i-th path, the wires have been stabilized at first and the first competitor who walks through it would not break the wires. Since then, however, when a person goes through the i-th path, there is a chance of p_i to touch the wires and affect the whole network. Moreover, to protect these wires, no more than c_i competitors are allowed to walk through the i-th path. Now you need to find a way for all competitors to get their lunch, and minimize the possibility of network crashing.

Input

The first line of input contains an integer t which is the number of test cases. Then t test cases follow. For each test case, the first line consists of two integers N ($N \le 100$) and M ($M \le 5000$). Each of the next N lines contains two integers s_i and b_i ($s_i, b_i \le 200$). Each of the next M lines contains three integers u_i, v_i, and c_i ($c_i \le 100$) and a floating-point number p_i ($0 < p_i < 1$). It is guaranteed that there is at least one way to let every competitor have lunch.

Output

For each test case, output the minimum possibility that the network would break down. Round it to 2 digits.

Examples

Input

```
1
4 4
2 0
0 3
3 0
0 3
1 2 5 0.5
3 2 5 0.5
1 4 5 0.5
3 4 5 0.5
```

Output

```
0.50
```

Source

2016 ACM-ICPC Asia Qingdao Regional Contest

Hiring Employees

BuBu has stepped into a challenging role as the head of the human resources department for a subsidiary of the Olympic committee. His task is to recruit a team of employees for a new Olympic project. The project spans N days, with day i requiring at least A_i employees.

The company has M types of employees available for hire. Employees of type i work from day S_i to day T_i and require a total salary of C_i dollars. BuBu's goal is to minimize the total cost of hiring enough employees for all necessary days.

Input

The first line of input contains two integers N and M, where N is the number of days and M is the number of employee types. The second line has N nonnegative integers, representing the minimum number of employees required each day. The next M lines each contain three integers S_i, T_i, and C_i, describing the availability and cost of each type of employee.

For 100% of the test cases, $1 \le N \le 1000$, and $1 \le M \le 10000$. Also, other values in the data will not exceed $2^{31} - 1$.

Output

Output one integer, the cost of the optimal hiring strategy.

Examples

Input

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

Output

```
14
```

Source

2008 China National Olympiad in Informatics