Practice Contest for ICPC SouthWestern Europe Regional Contest 2018

Paris, 1 December 2018

Judges and Problem Setters

- Mehdi Bouaziz (Facebook)
- Christoph Dürr (CNRS, LIP6)
- Jean-Christophe Filliâtre (CNRS, LRI)
- Louis Jachiet (CNRS, CRIStAL)
- Jacques-Henri Jourdan (CNRS, LRI)
- Vincent Jugé (Université Paris-Est Marne-la-Vallée)
- Silviu Maniu (Université Paris-Sud)
- Raphaël Marinier (Google)
- Carine Pivoteau (Université Paris-Est Marne-la-Vallée)
- Pierre Senellart (École normale supérieure), *chief judge*
- Samuel Tardieu (Télécom ParisTech)

This problem set consists of 5 problems, on 13 pages.

Page intentionally left blank

A: Late Party

Time limit: 5 seconds

After the banquet tonight, you will leave the banquet venue together with the F new friends you have made and walk back to your hotel. You want to keep chatting with them as long as possible, though you may unfortunately not all be in the same hotel.

Everybody, you included, is taking one of the shortest paths back to their hotel.

Paris is made of N intersections numbered from 0 to $N - 1$ and S streets linking two intersections. Streets can of course be walked in both directions. All hotels are located at intersections and are reachable from the banquet. The banquet takes place at intersection number 0.

Input

The input comprises several lines, each consisting of integers separated with single spaces:

- The first line consists of the three integers N, S, and F.
- Each of the following S lines consists of three integers i, j, t, representing a street between intersections i and j, taking t minutes to walk through.
- The last line consists of $F + 1$ integers representing the intersections of the hotels where people stay; the first one is yours, the other ones are your friends' respective hotels.

Limits

- $1 \leqslant N \leqslant 100\,000$
- $0 \leqslant S \leqslant 300\,000$
- $0 \leqslant F \leqslant 10\,000$
- $1 \leqslant t \leqslant 1\,000\,000$

Output

The output should consist of a single line, whose content is an integer, the longest time (in minutes) you can stay accompanied by at least one friend.

Sample Input

```
7 12 1
0 1 1
0 2 2
0 2 4
1 3 2
2 3 1
1 4 2
4 2 2
1 5 3
2 5 1
5 6 2
4 6 2
3 6 3
3 6
```

Sample Output

```
2
```

B: Flood

Time limit: 4 seconds

Once again, the river Seine has overflowed. As moving all the precious items out of the basements of the Louvre museum takes a considerable amount of time, you are tasked with forecasting at which time the Louvre risks to be flooded. This time, you will rely on a new model of propagation of the flood that provides a "worst-case scenario" prediction of how much flood there will be in each neighborhood of Paris.

In this model, Paris is represented as an undirected graph. The set of vertices $V_0, \ldots, V_{N-1}$ corresponds to the various areas of Paris. For instance, the node V_0 corresponds to the Seine and the node V_1 to the Louvre. An edge (X, Y) indicates that water could flow directly from X to Y and from Y to X. However, due to gravity, the water flow will obviously depend on the altitude ($A(X)$ and $A(Y)$) of both areas.

Your model relies on discrete time, which means you study the flood at T instants indexed from 0 to $T-1$. In addition to the graph representing Paris, thanks to the engineers and meteorologists of the city, you also have access to some other data, used in the model:

- the altitude $A(n)$ corresponding to each area n;
- the water level $I(n)$ that there is initially in each area n (i.e., at time $t = 0$);
- a forecast $F(t)$ that predicts the water level the Seine will reach for each time $1 \leqslant t < T$. You may assume that $F(t)$ is increasing with time.

Then you rely on your model to predict a *maximum water level* $WL(n, t)$ that there can be at area n and at time $0 \leqslant t < T$.

The model is based on the following idea: if the water level is L in some area n at some instant t, then this water might flow at time $t+1$ to all the areas that are neighbors of n (taking into account the altitude difference). Formally, $WL(n, t)$ is defined as follows:

- $WL(n, 0) = I(n)$ for all n.
- $WL(0, t) = F(t)$ for all $0 < t < T$.
- At time $t+1$, the maximum water level in area n is the maximum between the maximum water level at time t and the water propagated from a flooded neighbor area; formally:

$$WL(n, t+1) = \max \begin{cases} WL(n, t) \\ WL(v, t) + A(v) - A(n) \quad \text{for } v \text{ neighbor of } n \text{ such that } WL(v, t) > 0. \end{cases}$$

  Note that the propagation of the water level takes into account the difference of altitude and that the water only propagates if there actually is some water.

Your goal is to determine the smallest t ($0 \leqslant t < T$) such that the Louvre can be flooded at this time t (that is, such that $WL(1, t) > 0$).

Input

The input comprises several lines, each consisting of integers separated with single spaces:

- The first line contains three integers N, E, T where:
  - $2 \leqslant N \leqslant 5\,000$ is the number of areas;
  - $2 \leqslant E \leqslant 10\,000$ is the number of edges;
  - $2 \leqslant T \leqslant 100\,000$ is the number of instants.
- The second line contains N integers: $A(V_0) \ldots A(V_{N-1})$.
- The third line contains N integers: $I(V_0) \ldots I(V_{N-1})$.
- The fourth line contains $T - 1$ integers: $F(1) \ldots F(T-1)$.
- The final E lines each contain a pair of integers describing an edge in the graph.

All the altitudes, initial measurements, and forecasts are integers in the range between 0 and 1 000 000 000.

Output

The output should consist of a single line, whose content is the smallest t such that the Louvre is flooded. If the Louvre is not flooded at a time $t < T$ then your program should answer -1.

Sample Input 1

```
5 4 10
0 1 4 4 5
1 0 0 0 0
2 3 4 5 6 7 8 9 10
0 2
0 4
2 3
3 1
```

Sample Output 1

```
7
```

Sample Explanation 1

In the sample input above, there are 5 nodes, 4 edges and 10 predictions. The evolution of WL in depicted in the following array, which shows that the Louvre is flooded at time $t = 7$.

Areas:	0	1	2	3	4
$t = 0$	1	0	0	0	0
$t = 1$	2	0	0	0	0
$t = 2$	3	0	0	0	0
$t = 3$	4	0	0	0	0
$t = 4$	5	0	0	0	0
$t = 5$	6	0	1	0	0
$t = 6$	7	0	2	1	1
$t = 7$	8	4	3	2	2
$t = 8$	9	5	4	3	3
$t = 9$	10	6	5	4	4

Sample Input 2

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

Sample Output 2

```
-1
```

Sample Explanation 2

In the sample input above, there are 5 nodes, 4 edges, and 5 predictions. The evolution of *WL* in depicted in the following array, which shows that the Louvre does not get flooded in the studied period.

Areas:	0	1	2	3	4
$t = 0$	1	0	0	1	0
$t = 1$	2	0	0	1	1
$t = 2$	3	0	0	1	1
$t = 3$	4	0	0	1	1
$t = 4$	5	0	0	1	1

Page intentionally left blank

C: Big Money

Time limit: 2 seconds

Every year, Paris welcomes dozens of millions of tourists, who bring a considerable revenue to the city and its tourist industry. As an employee of the tourism bureau of the city of Paris, you are tasked with computing how much money was spent by tourists on a particular week. You are given credit card receipts for that week and asked to tally them. As money is important, you want to obtain the exact number, to the cent.

Input

The input consists of n lines, all of which contain a decimal number representing the expense e in euros. All decimal numbers are formatted as a non-empty sequence of decimal digits, a point (`.`) used as decimal separator, and two decimal digits for the fraction in cents.

Limits

$$\begin{array}{lcccl} 1 & \leqslant & n & \leqslant & 1\,000\,000 \\ 0.00 & \leqslant & e & \leqslant & 1\,000\,000.00 \end{array}$$

Output

The output should consist of a single line, whose content is the total expenses, a decimal number in the same format as the input.

Sample Input

```
1432.23
216.09
1475.09
1327.20
2457.18
```

Sample Output

```
6907.79
```

Page intentionally left blank

D: Paris Sightseeing for Groups

Time limit: 6 seconds

Welcome to your new job at *Paris Sightseeing for Groups* (PSG). Just like every other employee at PSG, you are in charge of planning trips for groups of people. Your salary will be indexed on your "customer score". This "customer score" is computed as the maximal h such that there are at least h groups that gave your trip a grade of at least h.

You have prepared several trips for each group according to their specific taste. You know in advance how each group will like each of your trips. However some of these possibilities will take more of your time and some will cost more than others. Since you cannot work 100 hours per week and you have a limited budget, you cannot simply maximize your "customer score" by taking the most liked trip for each group.

You want a program that tells you the maximal customer score that is possible to achieve considering your time and money budgets. Note that you have to plan exactly one trip for each group!

Input

The input comprises several lines, each consisting of integers separated with single spaces.

- The first line contains three integers:
  - N the number of groups;
  - M_{tot} the total money that you have;
  - T_{tot} the total time dedicated for visits.
- Then each of the N groups is described by several lines:
  - The first line contains the integer P_i which is the number of possibilities for the i-th group.
  - This line is followed by P_i lines, each containing three integers $M_{i,j}$, $T_{i,j}$, and $S_{i,j}$ describing the j-th possibility to the i-th group: $M_{i,j}$ is the amount of money needed, $T_{i,j}$ is the time for the visit, $S_{i,j}$ is the grade.

Limits

- $0 \leqslant M_{i,j} \leqslant M_{\text{tot}} \leqslant 2\,500$;
- $0 \leqslant T_{i,j} \leqslant T_{\text{tot}} \leqslant 2\,500$;
- $0 \leqslant S_{i,j} \leqslant 2\,500$;
- $1 \leqslant P_i \leqslant 5$;
- $3 \leqslant N \leqslant 100$.

Output

The output should consist of a single line, whose content is an integer h, the maximal h such that it is possible to give a grade of at least h to at least h groups. If it is not possible to plan a trip for each group then the output should be -1.

Sample Input 1

```
3 3 3
1
1 1 1
2
2 0 1
0 3 2
2
3 0 2
0 2 1
```

Sample Output 1

```
1
```

Sample Explanation 1

There is only one assignment possible. We select the first possibility for the groups 1 and 2, and the second possibility for the third group. This amounts to 3 units of time and 3 units of money.

Sample Input 2

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

Sample Output 2

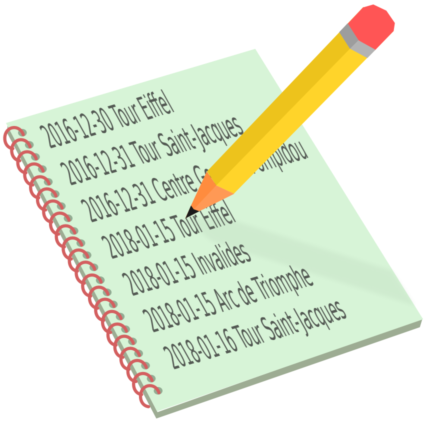
```
3
```

Sample Explanation 2

If we select the first possibility for the first group, then we can take the most liked possibility for all other groups. We can therefore serve all groups and ensure that 4 groups have a grade of 3.

E: Counting Monuments

Time limit: 5 seconds



Alice loves visiting Paris. Every time she is in the city, she tries to see as many monuments as she can, and every day of her visit she notes down in her travel notebook which monument she saw that day. After many years and many trips to Paris, Alice is wondering how many different monuments she saw in her combined visits. She shows you her travel notebook, and asks for your help in computing this number.

Input

The input consists of n lines, all of which contain, first, a date (in the format `YYYY-MM-DD`), then a space, then the name of a monument. Names of monuments are character strings of arbitrary length and containing arbitrary printable characters except for the newline character.

Limits

$1 \leqslant n \leqslant 1\,000\,000$

Output

The output should consist of a single line, whose content is an integer, the number m of distinct monuments in the input.

Sample Input

```
2016-12-30 Tour Eiffel
2016-12-31 Tour Saint-Jacques
2016-12-31 Centre Georges Pompidou
2018-01-15 Tour Eiffel
2018-01-15 Invalides
2018-01-15 Arc de Triomphe
2018-01-16 Tour Saint-Jacques
2018-01-16 Panthéon
2018-01-17 Sacré Cœur
```

Sample Output

```
7
```