ACM International Collegiate Programming Contest 2015

Latin American Regional Contests

November 13th-14th, 2015

Contest Session

This problem set contains 11 problems; pages are numbered from 1 to 15.

This problem set is used in simultaneous contests hosted in the following countries:

Argentina, Bolivia, Brasil, Chile, Colombia, Cuba,
México, Panamá, Perú, República Dominicana and Venezuela

v1.0

General information

Unless otherwise stated, the following conditions hold for all problems.

Program name

1. Your solution must be called *codename*`.c`, *codename*`.cpp` or *codename*`.java`, where *codename* is the capital letter which identifies the problem.

Input

1. The input must be read from standard input.
2. The input consists of a single test case, which is described using a number of lines that depends on the problem. No extra data appear in the input.
3. When a line of data contains several values, they are separated by *single* spaces. No other spaces appear in the input. There are no empty lines.
4. The English alphabet is used. There are no letters with tildes, accents, diaereses or other diacritical marks (ñ, Ã, é, Ì, ô, Ü, ç, etcetera).
5. Every line, including the last one, has the usual end-of-line mark.

Output

1. The output must be written to standard output.
2. The result of the test case must appear in the output using a number of lines that depends on the problem. No extra data should appear in the output.
3. When a line of results contains several values, they must be separated by *single* spaces. No other spaces should appear in the output. There should be no empty lines.
4. The English alphabet must be used. There should be no letters with tildes, accents, diaereses or other diacritical marks (ñ, Ã, é, Ì, ô, Ü, ç, etcetera).
5. Every line, including the last one, must have the usual end-of-line mark.
6. To output real numbers, round them to the closest rational with the required number of digits after the decimal point. Test case is such that there are no ties when rounding as specified.

Development team

The following persons helped to develop the problem set by creating and improving statements, solutions, test cases and input and output checkers:

Alejandro Strejilevich de Loma, Argentina

Bruno Junqueira Adami, Brazil

Fidel I. Schaposnik Massolo, Argentina

Guilherme Albuquerque Pinto, Brazil

Pablo Ariel Heiber, Argentina

Problem A – At most twice

Author: Pablo Ariel Heiber, Argentina

Given a positive integer U, find the largest integer L such that $L \leq U$ and L does not contain any digit more than twice.

Input

The input consists of a single line that contains an integer U ($1 \leq U \leq 10^{18}$).

Output

Output a line with an integer representing the largest number less than or equal to U that does not contain any digit more than twice.

Sample input 1	Sample output 1
2210102960	2210099887

Sample input 2	Sample output 2
1000000000000000000	998877665544332211

Sample input 3	Sample output 3
1001223343	998877665

Sample input 4	Sample output 4
20152015	20152015

Problem B – Blood groups

Author: Guilherme Albuquerque Pinto, Brazil

There are four possible blood groups for humans: AB, A, B and O, meaning that the red blood cells have antigens of types, respectively, A and B, only A, only B, and no antigen at all. Our blood group is determined by two alleles in our DNA. Each allele is of type either A, B or O. The following table lists the possible allele combinations someone may have for each blood group:

Blood group	AB	A	B	O
Possible alleles	AB	OA,AA	OB,BB	OO

We inherit exactly one allele from each of our two parents. So, given the blood groups of the two parents, we can say for sure if some blood group is possible, or not, in their offspring. For example, if the blood groups of the two parents are AB and B, then the possible allele combinations for them are, respectively, {AB} and {OB,BB}. Since the order of the alleles does not matter, the possible allele combinations for the offspring are {OA,AB,OB,BB}. That means the blood groups AB, A and B are possible in their offspring, but the blood group O is not. Very nice indeed! But what if life on Earth had evolved so that a person had three parents, three alleles, and three different antigen types? The allele combinations would look like this:

Blood group	ABC	AB	AC	BC	A	B	C	O
Possible alleles	ABC	OAB,AAB ABB	OAC,AAC ACC	OBC,BBC BCC	OOA,OAA AAA	OOB,OBB BBB	OOC,OCC CCC	OOO

If the blood groups of the three parents are A, BC and O, then all blood groups are possible in their offspring, except groups BC and ABC.

The universe is vast! There may be, out there in space, some form of life whose individuals have N parents, N alleles, and N different antigen types. Given the blood groups for the N parents, and a list of Q blood groups to test, your program has to determine which ones are possible, and which ones are not, in the offspring of the given parents.

Input

The first line contains two integers N and Q, representing respectively the number of parents (and alleles, and antigen types) and the number of queries ($1 \leq N \leq 100$ and $1 \leq Q \leq 40$). Each of the next N lines describes the blood group of a parent. After that, each of the next Q lines describes a blood group to test. Antigen types are identified with distinct integers from 1 to N, not letters. Each line describing a blood group contains an integer B indicating the number of antigen types in the blood group ($0 \leq B \leq N$), followed by B different integers $C_1, C_2, \ldots, C_B$ representing the antigen types present in the blood group ($1 \leq C_i \leq N$ for $i = 1, 2, \ldots, B$).

Output

For each of the Q queries, output a line with the uppercase letter "Y" if the corresponding blood group is possible in the offspring of the given parents; otherwise output the uppercase letter "N". Write the results in the same order that the queries appear in the input.

Sample input 1

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

Sample output 1

```
N
```

Sample input 2

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

Sample output 2

```
Y
N
Y
N
```

Sample input 3	**Sample output 3**

`4 3`
`4 2 1 3 4`
`4 2 1 3 4`
`1 1`
`1 2`
`1 3`
`2 2 1`
`0`
	`Y`
	`Y`
	`N`

Problem C – Cake cut

Author: Pablo Ariel Heiber, Argentina

Carol and Carla are roommates. Yesterday they threw a big party and today they have a partially eaten cake that they want to divide. Since people were careless when cutting themselves a slice, the cake is now shaped as a prism with its top and bottom faces being the same simple convex polygon.

To add some fun to the process of dividing the cake, the girls came up with the following game. Carol chooses a vertex v of the top face of the cake. Carla chooses another vertex w of the top face that is not adjacent to v. Then, they cut the cake into two pieces by extending downwards the segment vw, so as to obtain two separate pieces of cake, each in the shape of a prism. Finally, Carol chooses the piece that she prefers, and Carla gets the other one. Carla immediately saw that this system gives Carol an advantage. Carla wants to know exactly how much of an advantage Carol has.

You are given a polygon that represents both the top and bottom faces of the cake. The height of the cake is 2, so the volume of a piece of cake is 2 times the area of its top face. Assuming the cake is divided as explained, and that both girls make their decisions to maximize the volume of the piece they have at the end, compute the volume of the piece each girl will get.

Input

The first line contains an integer N representing the number of vertices of the polygonal top face of the cake ($4 \leq N \leq 10^5$). Each of the next N lines describes a vertex of the polygon with two integers X and Y, indicating the coordinates of the vertex in the XY plane ($-10^8 \leq X, Y \leq 10^8$). Vertices are given in counter clockwise order and define a simple convex polygon. No three points in the input are collinear.

Output

Output a line with two integers representing the volume of the piece Carol and Carla will get, in that order, if both make their decisions optimally.

Sample input 1	**Sample output 1**
5 0 0 3 0 3 1 2 2 0 1	7 2

Sample input 2	**Sample output 2**
6 0 1 1 0 2 0 3 1 2 2 0 2	6 3

Sample input 3	**Sample output 3**
4 -100000000 -100000000 100000000 -100000000 100000000 100000000 -100000000 100000000	40000000000000000 40000000000000000

Sample input 4	Sample output 4
4 -99999995 -100000000 100000000 -100000000 100000000 99999995 -100000000 100000000	39999999999999975 39999998000000025

Problem D – D as in Daedalus

Author: Guilherme Albuquerque Pinto, Brazil

Daedalus is playing the game of "Don't be greedy", in which N players sit around a table having each of them five cards labelled 1, 10, 100, 1000 and 10000 points. In "Don't be greedy" the players may not talk to each other once the game starts, and there are M rounds. In each round, the bank announces a budget B. Then each player chooses one of the cards and places it, face down, on the table. The bank then turns the cards, so that all players can see all N cards. If the sum of the points in the chosen cards is less than or equal to B, then the bank gives to each player exactly the amount of points in the card he or she chose. Otherwise, no one gets anything. Each player gets his or her card back before the next round. The players are very rational and would like to maximize their points and minimize their regrets! What would you do in this situation? Cooperate or defect?

Take the following table as an example. Daedalus won a total of 10 points, in the end, because only the first round was successful. But, looking back on the game, he sees that he could have won 110 points, if he had chosen 100 points in the first round and 10 points in the third round. That is, Daedalus could have won 100 extra points! This holds only, of course, assuming the cards chosen by the other players remain unchanged.

round	budget B	Daedalus	Iapyx	Icarus	Ariadne	Minos	sum	result
1	300	10	100	10	1	10	131	success
2	1100	100	10	100	1	1000	1211	fail
3	1200	100	100	10	1	1000	1211	fail

Given the budget and the cards chosen in each round, we need to compute the maximum total number of extra points Daedalus could have won, in the end, if he had chosen the best possible card in each round, assuming the cards chosen by the other players remain unchanged.

Input

The first line contains two integers N and M, representing respectively the number of players and the number of rounds ($1 \leq N \leq 20$ and $1 \leq M \leq 50$). Each of the next M lines describes a round with an integer B indicating the budget ($1 \leq B \leq 10^6$), followed by N integers $C_1, C_2, \ldots, C_N$ representing that the i-th player chose the card labelled with C_i points during that round ($C_i \in \{1, 10, 100, 1000, 10000\}$ for $i = 1, 2, \ldots, N$). Daedalus is the first player.

Output

Output a line with an integer representing the maximum total number of extra points Daedalus could have won, if he had chosen the best possible card in each round, assuming the cards chosen by the other players remain unchanged.

Sample input 1	Sample output 1
5 3 300 10 100 10 1 10 1100 100 10 100 1 1000 1200 100 100 10 1 1000	100

Sample input 2	Sample output 2
3 2 2000 1000 1000 1000 21 1 1 10	9

Problem E – Exposing corruption

Author: Walter Erquínigo Pezo, Perú

The Central Committee in Nlogonia is formed by many congress members. As the political system is dichotomic, each congress member belongs to one of two parties: the Deadly Serious Party and the Party! Party! Party. Historical tradition calls them DSP and PPP, respectively.

Edward is an investigative journalist. He discovered that congress members are corrupt and will switch parties if offered a given amount of Nlogmoney to do so. Each congress member has his or her own specific price, but they all have a price. As usual in politics, there exist rivalries between some pairs of congress members. Rivals would never accept to be in the same party.

Edward has a budget and wants to use it to make some congress members switch parties, thus gathering indisputable evidence for his investigation. In doing so, he has to respect rivalries: after everyone who was offered money switches, rivals must be left belonging to different parties.

Edward wants to cause maximum impact. Can you help him find out the maximum number of congress members that can belong to DSP if he uses at most all of his budget towards that goal? Similarly, what is the maximum number of congress members that can belong to PPP under the same constraints?

Input

The first line contains four integers D, P, R and B, representing respectively the number of congress members that initially belong to DSP ($1 \leq D \leq 100$), the number of congress members that initially belong to PPP ($1 \leq P \leq 100$), the number of rivalries among congress members ($1 \leq R \leq 2000$), and the budget of the journalist expressed in Nlogmoney ($1 \leq B \leq 10^4$). Members of DSP are identified with distinct integers from 1 to D, while members of PPP are identified with distinct integers from 1 to P. The second line contains D integers $S_1, S_2, \ldots, S_D$, indicating that member i of DSP will switch parties if offered S_i Nlogmoney ($1 \leq S_i \leq 100$ for $i = 1, 2, \ldots, D$). The third line contains P integers $T_1, T_2, \ldots, T_P$, indicating that member j of PPP will switch parties if offered T_j Nlogmoney ($1 \leq T_j \leq 100$ for $j = 1, 2, \ldots, P$). Each of the next R lines describes a rivalry with two integers X and Y, representing that member X of DSP and member Y of PPP are rivals ($1 \leq X \leq D$ and $1 \leq Y \leq P$).

Output

Output a line with two integers representing the maximum number of congress members that can belong to DSP using the given budget and, similarly, the maximum number of congress members that can belong to PPP using the given budget.

Sample input 1	Sample output 1
2 3 2 55 20 30 40 30 1 2 3 1 3	3 4

Sample input 2	Sample output 2
3 2 6 30 5 5 5 5 5 2 1 2 2 1 1 1 2 3 1 3 2	3 3

Problem F – Fence the vegetables fail

Author: Pablo Ariel Heiber, Argentina

At the early age of 40, Alice and Bob decided to retire. After more than two decades working as examples for networking protocols, game theoretical books and several other texts, they were tired. To remain active, they decided to get into gardening.

Alice and Bob planted several vegetable plants in a huge field. After finishing, they realized that their plants needed protection from wild animals, so they decided to build a fence around them. The field is represented as the XY plane, and each vegetable plant as a different point in it. A fence is represented as a polygon in the plane. However, not every polygon is a valid fence. The fence needs to be a single simple polygon with each of its sides parallel to one of the axes. Of course, the polygon should contain all the points representing vegetable plants. A fence too close to the plants or to itself could make it difficult to walk around, so each side of the polygon needs to be away from all plants and all non-adjacent sides.

Unfortunately, Alice and Bob subcontracted the construction of the fence to a nasty multinational. The company had a lot of lawyers on payroll, but no good fence designers, so they failed to comply to all requirements. They built a fence which is a simple polygon with sides parallel to the axes and whose sides are away from plants and itself. However, they forgot to make the fence contain all the plants!

Alice and Bob want to assess the extent of the problem. Since not all plants are equally valuable to them, they want to know the total value of the plants that were left outside the fence.

Input

The first line contains two integers P and V, representing respectively the number of plants and the number of vertices of the polygonal fence ($1 \leq P, V \leq 10^5$). Each of the next P lines describes a different plant with two integers X_p and Y_p, indicating the coordinates of the plant ($-10^9 \leq X_p, Y_p \leq 10^9$). The value of the p-th plant in the input is p, for $p = 1, 2, \ldots, P$. Each of the next V lines describes a vertex of the fence with two integers X_v and Y_v, indicating the coordinates of the vertex ($-10^9 \leq X_v, Y_v \leq 10^9$). Vertices are given in counter clockwise order. Each of these points is an actual vertex of the polygon, that is, it is not collinear with its two adjacent vertices. The represented polygon is a simple polygon with each side parallel to an axis. No two plants are in the same position, and no plant lies on a fence's side.

Output

Output a line with an integer representing the sum of the values of all the plants that lie outside the fence.

Sample input 1	**Sample output 1**
4 8 1 2 1 0 5 3 3 4 0 1 6 1 6 4 4 4 4 3 2 3 2 5 0 5	6

Sample input 2	Sample output 2

<pre>6 12
6 5
1 9
3 6
3 4
2 0
4 4
5 8
5 3
2 3
2 5
4 5
4 7
0 7
0 1
7 1
7 10
0 10
0 8</pre>
	<pre>15</pre>

Sample input 3	Sample output 3

<pre>1 4
1 1
2 0
2 2
0 2
0 0</pre>
	<pre>0</pre>

Problem G – Galactic taxes

Author: Rafael Armando Garcia Gomez, Colombia

The year is 2115. The Interplanetary Commercial Planning Center (ICPC) is supported by the Autonomous Communication Ministry (ACM).

A commercial operation is performed executing transactions between connected ACM offices throughout the galaxy. The execution of a transaction between two connected ACM offices involves a non-negative tax whose value increases, or decreases, continuously as a linear function $A \times t + B$ of time t, where t is a real number measured in minutes during the day ($0 \leq t \leq 24 \times 60$).

The total tax of a commercial operation performed between a source ACM office and a destination ACM office at some time t, is calculated as the minimum possible sum of the taxes of the executed transactions between the ACM offices visited along some path from the source ACM office to the destination ACM office. The tax of each transaction is calculated at the same time t.

Since the tax of the transactions between connected ACM offices is continually changing during the day, it would be better to perform the commercial operation at some specific time in the day, in order to maximize the collected tax. At that time, ACM decides to perform the commercial operation, and not before or after.

Your task is to write a program that receives as input the description of the ACM office network and returns as output the maximum total tax of the commercial operation that can be achieved during the day, that is, the maximum total tax that ACM can collect.

Input

The first line contains two integers N and M, representing respectively the number of ACM offices in the network, and the number of connections ($2 \leq N \leq 1000$ and $1 \leq M \leq 10^4$). The ACM offices are identified with distinct integers from 1 to N, being 1 the source ACM office and N the destination ACM office. Each of the next M lines describes a connection with four integers I, J, A and B, indicating that there is a bidirectional connection between office I and office J ($1 \leq I < J \leq N$), such that the tax of a transaction executed between office I and office J at time t is defined by the formula $A \times t + B$ ($-100 \leq A \leq 100$ and $0 \leq B \leq 10^6$). Taxes are non-negative, so $A \times t + B \geq 0$ for $0 \leq t \leq 24 \times 60$. There is at most one connection between each pair of ACM offices, and there is at least one path between the source ACM office and the destination ACM office.

Output

Output a line with a rational number representing the maximum total tax that ACM can collect. The result must be output as a rational number with exactly five digits after the decimal point, rounded if necessary.

Sample input 1	Sample output 1
2 1 1 2 1 0	1440.00000

Sample input 2	Sample output 2
5 8 1 2 27 610658 2 3 -48 529553 3 4 -6 174696 4 5 47 158238 3 5 84 460166 1 3 -21 74502 2 4 -13 858673 1 5 -90 473410	419431.27273

Sample input 3	Sample output 3
3 3 1 2 1 0 2 3 1 0 1 3 -1 1440	960.00000

Sample input 4	Sample output 4
4 5 1 2 1 0 2 4 2 0 1 4 0 500 1 3 -1 1440 3 4 -2 2880	500.00000

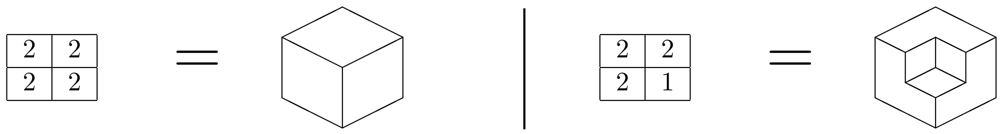
Sample input 5	Sample output 5
2 1 1 2 0 0	0.00000

Problem H – Height map

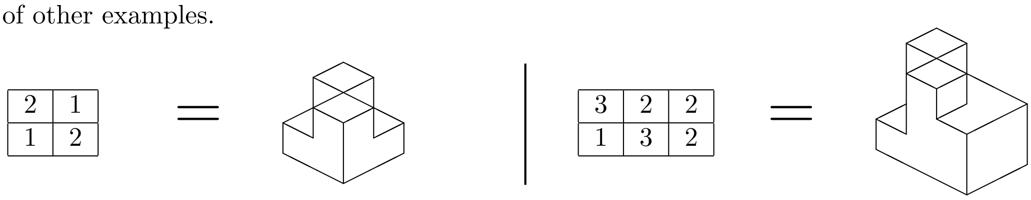
Author: Pablo Ariel Heiber, Argentina

A height map is a two-dimensional matrix of positive integers that represents a polyhedron. Each cell of the matrix with value V represents a parallelepiped shaped column of $1 \times 1 \times V$ that is laid on one of its 1×1 faces onto the cell. This creates a polyhedron with a single face in the bottom made up of all the downwards facing 1×1 faces combined, and possibly several faces on the top and on the sides.

For instance, a 2×2 matrix with all values equal to 2 represents a cube of side 2. However, if one of the values is 1, the represented polyhedron is the same cube with one corner cut off. The following picture represents both alternatives.



While not every polyhedron can be represented in this fashion, there are several that can. Here are a couple of other examples.



Given a height map, you are asked to count the number of faces of the represented polyhedron. Notice that a face is defined as a simple polygon that describes a contiguous and maximal boundary of the polyhedron. As you can see in the last two examples, it is possible for two different coplanar faces to share a common vertex, or even a side, or portions of a side.

Input

The first line contains two integers R and C, representing respectively the number of rows and columns of the height map ($1 \leq R, C \leq 100$). Each of the next R lines contains C integers; the j-th integer in the i-th line is the value $V_{i,j}$ located in the i-th row and j-th column of the matrix ($1 \leq V_{i,j} \leq 10^9$ for $i = 1, 2, \ldots, R$ and $j = 1, 2, \ldots, C$).

Output

Output a line with an integer representing the number of faces of the polyhedron represented by the input height map.

Sample input 1	Sample output 1
2 2 2 2 2 2	6

Sample input 2	Sample output 2
2 2 2 2 2 1	9

Sample input 3	Sample output 3
2 2 2 1 1 2	13

Sample input 4	Sample output 4
2 3 3 2 2 1 3 2	14

Problem I – Identifying tea

Author: Inés Kereki, Uruguay

Blind tea tasting is the skill of identifying a tea by using only your senses of smell and taste.

As part of the Ideal Challenge of Pure-Tea Consumers (ICPC), a local TV show is organized. During the show, a full teapot is prepared and five contestants are handed a cup of tea each. The participants must smell, taste and assess the sample so as to identify the tea type, which can be: (1) white tea; (2) green tea; (3) black tea; or (4) herbal tea. At the end, the answers are checked to determine the number of correct guesses.

Given the actual tea type and the answers provided, determine the number of contestants who got the correct answer.

Input

The first line contains an integer T representing the tea type ($1 \leq T \leq 4$). The second line contains five integers A, B, C, D and E, indicating the answer given by each contestant ($1 \leq A, B, C, D, E \leq 4$).

Output

Output a line with an integer representing the number of contestants who got the correct answer.

Sample input 1	**Sample output 1**
1 1 2 3 2 1	2

Sample input 2	**Sample output 2**
3 4 1 1 2 1	0

Problem J – Just a bit sorted

Author: Jorge Enrique Moreira Broche, Cuba

Jurgen Guntherswarchzhaffenstrassen is known for his virtuous guitar playing and the cruel teaching methods he employs with his students. What most people ignore about him is that he is also a fan of numbers.

Lately Jurgen has been studying sorted lists, but he is getting bored. He thinks that such lists are too predictable and not very abundant, so he decided to spice things up a bit.

Jurgen says that a list ℓ of N not necessarily different positive integers is just a bit sorted if and only if for each positive integer $x > 1$ that occurs in ℓ, the number $x - 1$ appears at least once before the last occurrence of x in ℓ. For example

- $[\mathbf{2}, \mathbf{3}, \mathbf{1}, \mathbf{2}]$ is just a bit sorted because a 1 appears before the last 2, and a 2 appears before the last 3;
- $[\mathbf{2}, \mathbf{3}, \mathbf{4}, \mathbf{3}, \mathbf{2}, \mathbf{1}, \mathbf{3}, \mathbf{4}]$ is not just a bit sorted because every 1 appears after the last 2;
- $[\mathbf{1}, \mathbf{1}, \mathbf{3}, \mathbf{1}, \mathbf{3}, \mathbf{3}, \mathbf{1}, \mathbf{3}]$ is not just a bit sorted because no 2 appears before the last 3 (since 2 doesn't appear at all in this list).

Jurgen is trying to find out how many different just a bit sorted lists of N positive integers not greater than K exist. Two lists are different if and only if there is at least one position in which the lists have distinct elements. Can you help Jurgen in counting the number of different lists?

Input

The first line contains two integers N and Q, representing respectively the number of elements in the just a bit sorted lists and the number of queries to answer ($1 \leq N \leq 5000$ and $1 \leq Q \leq 1000$). The second line contains Q integers $K_1, K_2, \ldots, K_Q$, indicating that the lists you must count in the i-th query cannot contain values greater than K_i ($1 \leq K_i \leq 10^9$ for $i = 1, 2, \ldots, Q$).

Output

Output a line with Q integers, such that the i-th integer represents the number of different just a bit sorted lists of N positive integers not greater than K_i (for $i = 1, 2, \ldots, Q$). Since this number can be very large, output the remainder of dividing it by $10^9 + 7$.

Sample input 1	Sample output 1
1 1 1	1

Sample input 2	Sample output 2
3 4 2 2 1 10	5 5 1 6

Sample input 3	Sample output 3
1000 3 100 5 300	265428620 285047952 668355714

Problem K – Keep it energized

Author: Fidel I. Schaposnik Massolo, Argentina

The Incredible Consoles Production Company (ICPC) is now designing its newest video game console model, the Super-Arcade Reloaded (SAR). The launch of the SAR will be accompanied by the release of a flagship game, which will only be available to its users. This game, which incidentally shall be called Adventures of Captain Mikado (ACM), even features an in-game currency which can be conveniently bought using real-world money!

The ACM is a very simple game consisting of N levels numbered $1, 2, \dots, N$, the i-th level requiring exactly E_i units of energy to be completed. This means that in order to complete that level, the user's energy should be at least E_i, and after doing so it will decrease in exactly that amount. To win the game the user should complete all the levels in increasing order, starting at level 1 and continuing until level N without ever going back to some already-completed level.

Initially the user starts with no energy, and in order to get some he must buy energy packs from the in-game shops distributed among the N levels. There are M such shops. Each shop sells an energy pack having a strength S and a cost C that depend on the shop. The user can only buy energy packs from the shops in the level he is currently in, before starting to complete that level. The effect of buying an energy pack of strength S is that the user's energy immediately turns into S, regardless of which value it had before.

In order to increase even further its sales, the ICPC has thought of a revolutionary promotion: it will reimburse the full cost of the SAR to whoever completes the ACM game using the minimum amount of in-game cash. Given the description of the game, can you help them find out what is the minimum amount of in-game cash required to finish the game?

Input

The first line contains two integers N and M, representing respectively the number of levels and the number of shops in the game ($1 \leq N, M \leq 10^5$). The second line contains N integers $E_1, E_2, \dots, E_N$, where E_i is the energy required to complete the i-th level ($1 \leq E_i \leq 10^4$ for $i = 1, 2, \dots, N$). Each of the next M lines describes a shop with three integers L, S and C, representing respectively the level where the shop is located, and the strength and cost of the energy pack it sells ($1 \leq L \leq N$, $1 \leq S \leq 10^9$ and $1 \leq C \leq 10^4$).

Output

Output a line with an integer representing the minimum amount of in-game cash that is required to complete all N levels in the game. If it is impossible to complete all the levels, write the value -1.

Sample input 1

```
5 4
1 2 3 4 5
1 6 5
2 14 10
5 5 4
3 7 5
```

Sample output 1

```
14
```

Sample input 2

```
3 4
14 11 2015
1 14 23
2 11 9
3 1987 1
1 2039 33
```

Sample output 2

```
-1
```