A. The Two Routes

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

In Absurdistan, there are n towns (numbered 1 through n) and m bidirectional railways. There is also an absurdly simple road network — for each pair of different towns x and y, there is a bidirectional road between towns x and y **if and only if** there is no railway between them. Travelling to a different town using one railway or one road always takes exactly one hour.

A train and a bus leave town 1 at the same time. They both have the same destination, town n, and don't make any stops on the way (but they can wait in town n). The train can move only along railways and the bus can move only along roads.

You've been asked to plan out routes for the vehicles; each route can use any road/railway multiple times. One of the most important aspects to consider is safety — in order to avoid accidents at railway crossings, the train and the bus must not arrive at the same town (except town n) simultaneously.

Under these constraints, what is the minimum number of hours needed for both vehicles to reach town n (the maximum of arrival times of the bus and the train)? Note, that bus and train are not required to arrive to the town n at the same moment of time, but are allowed to do so.

Input

The first line of the input contains two integers n and m $(2 \le n \le 400, 0 \le m \le n(n - 1) / 2)$ — the number of towns and the number of railways respectively.

Each of the next m lines contains two integers u and v, denoting a railway between towns u and v $(1 \le u, v \le n, u \ne v)$.

You may assume that there is at most one railway connecting any two towns.

Output

Output one integer — the smallest possible time of the later vehicle's arrival in town n. If it's impossible for at least one of the vehicles to reach town n, output -1.

Sample test(s)

input
```
4 2
1 3
3 4
```
output
```
2
```

input
```
4 6
1 2
1 3
1 4
2 3
2 4
3 4
```
output
```
-1
```

input
```
5 5
4 2
3 5
4 5
5 1
1 2
```
output
```
3
```

Note

In the first sample, the train can take the route $1 \to 3 \to 4$ and the bus can take the route $1 \to 2 \to 4$. Note that they can arrive at town 4 at the same time.

In the second sample, Absurdistan is ruled by railwaymen. There are no roads, so there's no way for the bus to reach town 4.

B. Lipshitz Sequence

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

A function $f : \mathbb{R} \to \mathbb{R}$ is called Lipschitz continuous if there is a real constant K such that the inequality $|f(x) - f(y)| \le K \cdot |x - y|$ holds for all $x, y \in \mathbb{R}$. We'll deal with a more... discrete version of this term.

For an array $\mathrm{h}[1..n]$, we define it's Lipschitz constant $L(\mathrm{h})$ as follows:

- if $n < 2$, $L(\mathrm{h}) = 0$
- if $n \ge 2$, $L(\mathrm{h}) = \max \left\lceil \frac{|\mathrm{h}[j] - \mathrm{h}[i]|}{j - i} \right\rceil$ over all $1 \le i < j \le n$

In other words, $L = L(\mathrm{h})$ is the smallest non-negative integer such that $|h[i] - h[j]| \le L \cdot |i - j|$ holds for all $1 \le i, j \le n$.

You are given an array a of size n and q queries of the form $[l, r]$. For each query, consider the subarray $s = \mathrm{a}[l..r]$; determine the sum of Lipschitz constants of **all subarrays** of s.

Input

The first line of the input contains two space-separated integers n and q ($2 \le n \le 100\,000$ and $1 \le q \le 100$) — the number of elements in array a and the number of queries respectively.

The second line contains n space-separated integers $\mathrm{a}[1..n]$ ($0 \le \mathrm{a}[i] \le 10^8$).

The following q lines describe queries. The i-th of those lines contains two space-separated integers l_i and r_i ($1 \le l_i < r_i \le n$).

Output

Print the answers to all queries in the order in which they are given in the input. For the i-th query, print one line containing a single integer — the sum of Lipschitz constants of all subarrays of $\mathrm{a}[l_i..r_i]$.

Sample test(s)

input

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

output

```
17
82
23
210
```

input

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

output

```
2
0
22
59
16
8
```

Note

In the first query of the first sample, the Lipschitz constants of subarrays of $[5,2,9]$ with length at least 2 are:

- $L([5,2]) = 3$
- $L([2,9]) = 7$
- $L([5,2,9]) = 7$

The answer to the query is their sum.

C. Kleofáš and the n-thlon

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Kleofáš is participating in an n-thlon - a tournament consisting of n different competitions in n different disciplines (numbered 1 through n). There are m participants in the n-thlon and each of them participates in all competitions.

In each of these n competitions, the participants are given *ranks* from 1 to m in such a way that no two participants are given the same rank - in other words, the ranks in each competition form a permutation of numbers from 1 to m. The *score* of a participant in a competition is equal to his/her rank in it.

The *overall score* of each participant is computed as the sum of that participant's scores in all competitions.

The *overall rank* of each participant is equal to $1 + k$, where k is the number of participants with **strictly smaller** overall score.

The n-thlon is over now, but the results haven't been published yet. Kleofáš still remembers his ranks in each particular competition; however, he doesn't remember anything about how well the other participants did. Therefore, Kleofáš would like to know his expected overall rank.

All competitors are equally good at each discipline, so all rankings (permutations of ranks of everyone except Kleofáš) in each competition are equiprobable.

Input

The first line of the input contains two space-separated integers n ($1 \le n \le 100$) and m ($1 \le m \le 1000$) — the number of competitions and the number of participants respectively.

Then, n lines follow. The i-th of them contains one integer x_i ($1 \le x_i \le m$) — the rank of Kleofáš in the i-th competition.

Output

Output a single real number – the expected overall rank of Kleofáš. Your answer will be considered correct if its relative or absolute error doesn't exceed 10^{-9}.

Namely: let's assume that your answer is a, and the answer of the jury is b. The checker program will consider your answer correct, if $\frac{|a-b|}{\max(1,b)} \le 10^{-9}$.

Sample test(s)

input

```
4 10
2
1
2
1
```

output

```
1.0000000000000000
```

input

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

output

```
2.7500000000000000
```

input

```
3 6
2
4
2
```

output

```
1.6799999999999999
```

Note

In the first sample, Kleofáš has overall score 6. Nobody else can have overall score less than 6 (but it's possible for one other person to have overall score 6 as well), so his overall rank must be 1.

D. Acyclic Organic Compounds

time limit per test: 3 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

You are given a tree T with n vertices (numbered 1 through n) and a letter in each vertex. The tree is rooted at vertex 1.

Let's look at the subtree T_v of some vertex v. It is possible to read a string along each simple path starting at v and ending at some vertex in T_v (possibly v itself). Let's denote the number of **distinct** strings which can be read this way as $\mathrm{dif}(v)$.

Also, there's a number c_v assigned to each vertex v. We are interested in vertices with the maximum value of $\mathrm{dif}(v) + c_v$.

You should compute two statistics: the maximum value of $\mathrm{dif}(v) + c_v$ and the number of vertices v with the maximum $\mathrm{dif}(v) + c_v$.

Input

The first line of the input contains one integer n ($1 \le n \le 300\,000$) — the number of vertices of the tree.

The second line contains n space-separated integers c_i ($0 \le c_i \le 10^9$).

The third line contains a string s consisting of n lowercase English letters — the i-th character of this string is the letter in vertex i.

The following $n - 1$ lines describe the tree T. Each of them contains two space-separated integers u and v ($1 \le u, v \le n$) indicating an edge between vertices u and v.

It's guaranteed that the input will describe a tree.

Output

Print two lines.

On the first line, print $m = \max(\mathrm{dif}(i) + c_i)$ over all $1 \le i \le n$.

On the second line, print the number of vertices v for which $m = \mathrm{dif}(v) + c_v$.

Sample test(s)

input

```
10
1 2 7 20 20 30 40 50 50 50
cacabbcddd
1 2
6 8
7 2
6 2
5 4
5 9
3 10
2 5
2 3
```

output

```
51
3
```

input

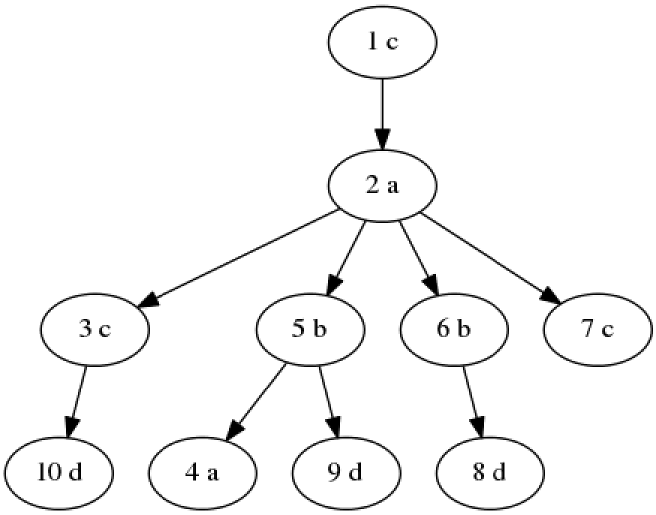
```
6
0 2 4 1 1 1
raaaba
1 2
2 3
2 4
2 5
3 6
```

output

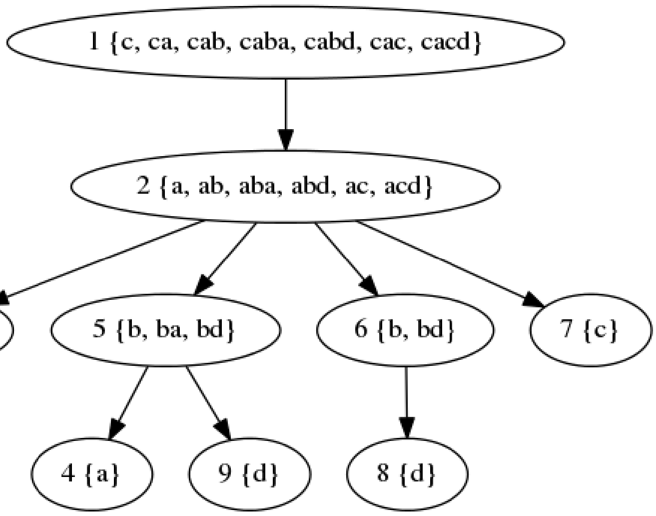
```
6
2
```

Note

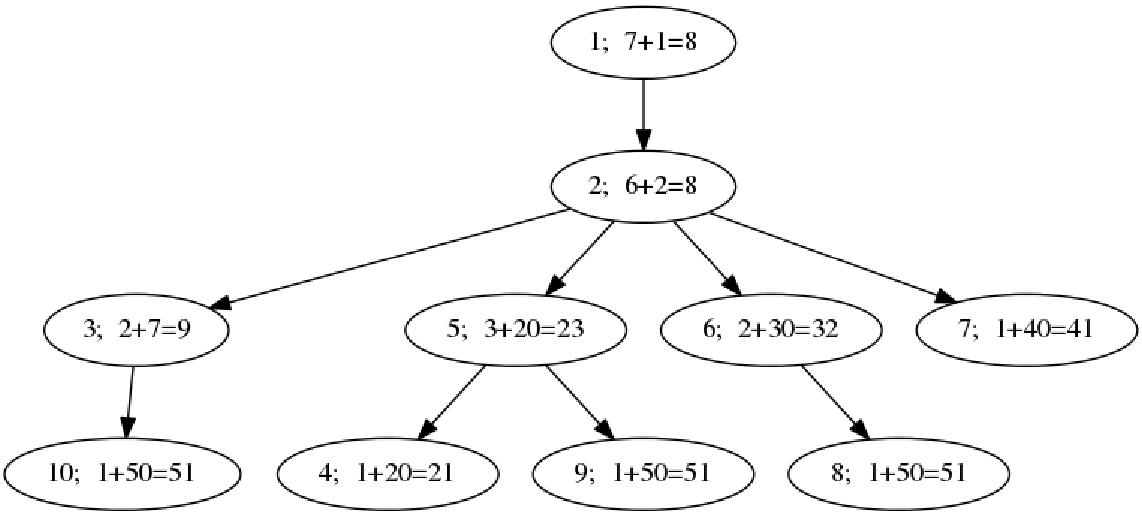
In the first sample, the tree looks like this:



The sets of strings that can be read from individual vertices are:



Finally, the values of $\mathrm{dif}(v) + c_v$ are:



In the second sample, the values of $\mathrm{dif}(1..n)$ are $(5, 4, 2, 1, 1, 1)$. The distinct strings read in T_2 are `a`, `aa`, `aaa`, `ab`; note that `aa` can be read down to vertices 3 or 4.

E. A Museum Robbery

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

There's a famous museum in the city where Kleofáš lives. In the museum, n exhibits (numbered 1 through n) had been displayed for a long time; the i-th of those exhibits has value v_i and mass w_i.

Then, the museum was bought by a large financial group and started to vary the exhibits. At about the same time, Kleofáš... gained interest in the museum, so to say.

You should process q events of three types:

- type 1 — the museum displays an exhibit with value v and mass w; the exhibit displayed in the i-th event of this type is numbered $n + i$ (see sample explanation for more details)
- type 2 — the museum removes the exhibit with number x and stores it safely in its vault
- type 3 — Kleofáš visits the museum and wonders (for no important reason at all, of course): if there was a robbery and exhibits with total mass at most m were stolen, what would their maximum possible total value be?

For each event of type 3, let $s(m)$ be the maximum possible total value of stolen exhibits with total mass $\le m$.

Formally, let D be the set of numbers of all exhibits that are currently displayed (so initially D = {1, ..., n}). Let $P(D)$ be the set of all subsets of D and let

$$G = \left\{ S \in P(D) \,\middle|\, \sum_{i \in S} w_i \le m \right\} .$$

Then, $s(m)$ is defined as

$$s(m) = \max_{S \in G} \left(\sum_{i \in S} v_i \right) .$$

Compute $s(m)$ for each $m \in \{1, 2, \ldots, k\}$. Note that the output follows a special format.

Input

The first line of the input contains two space-separated integers n and k ($1 \le n \le 5000$, $1 \le k \le 1000$) — the initial number of exhibits in the museum and the maximum interesting mass of stolen exhibits.

Then, n lines follow. The i-th of them contains two space-separated positive integers v_i and w_i ($1 \le v_i \le 1\,000\,000$, $1 \le w_i \le 1000$) — the value and mass of the i-th exhibit.

The next line contains a single integer q ($1 \le q \le 30\,000$) — the number of events.

Each of the next q lines contains the description of one event in the following format:

- $1\ v\ w$ — an event of type 1, a new exhibit with value v and mass w has been added ($1 \le v \le 1\,000\,000$, $1 \le w \le 1000$)
- $2\ x$ — an event of type 2, the exhibit with number x has been removed; it's guaranteed that the removed exhibit had been displayed at that time
- 3 — an event of type 3, Kleofáš visits the museum and asks his question

There will be at most $10\,000$ events of type 1 and at least one event of type 3.

Output

As the number of values $s(m)$ can get large, output the answers to events of type 3 in a special format.

For each event of type 3, consider the values $s(m)$ computed for the question that Kleofáš asked in this event; print one line containing a single number

$$\sum_{m=1}^{k} s(m) \cdot p^{m-1} \mod q ,$$

where $p = 10^7 + 19$ and $q = 10^9 + 7$.

Print the answers to events of type 3 in the order in which they appear in the input.

Sample test(s)

input

```
3 10
30 4
60 6
5 1
9
3
1 42 5
1 20 3
3
2 2
```

```
2 4
3
1 40 6
3
```

output

```
556674384
168191145
947033915
181541912
```

input

```
3 1000
100 42
100 47
400 15
4
2 2
2 1
2 3
3
```

output

```
0
```

Note

In the first sample, the numbers of displayed exhibits and values $s(1)$, ..., $s(10)$ for individual events of type 3 are, in order:

$$\text{exhibits } 1,2,3; \quad (s(1),\ldots,s(10)) = (5,5,5,30,35,60,65,65,65,90),$$
$$\text{exhibits } 1,2,3,4,5; \quad (s(1),\ldots,s(10)) = (5,5,20,30,42,60,65,65,80,90),$$
$$\text{exhibits } 1,3,5; \quad (s(1),\ldots,s(10)) = (5,5,20,30,35,35,50,55,55,55),$$
$$\text{exhibits } 1,3,5,6; \quad (s(1),\ldots,s(10)) = (5,5,20,30,35,40,50,55,60,70).$$

The values of individual exhibits are $v_1 = 30$, $v_2 = 60$, $v_3 = 5$, $v_4 = 42$, $v_5 = 20$, $v_6 = 40$ and their masses are $w_1 = 4$, $w_2 = 6$, $w_3 = 1$, $w_4 = 5$, $w_5 = 3$, $w_6 = 6$.

In the second sample, the only question is asked after removing all exhibits, so $s(m) = 0$ for any m.