A. Win or Freeze

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

You can't possibly imagine how cold our friends are this winter in Nvodsk! Two of them play the following game to warm up: initially a piece of paper has an integer q. During a move a player should write any integer number that is a *non-trivial* divisor of the last written number. Then he should run this number of circles around the hotel. Let us remind you that a number's divisor is called *non-trivial* if it is different from one and from the divided number itself.

The first person who **can't make a move wins** as he continues to lie in his warm bed under three blankets while the other one keeps running. Determine which player wins considering that both players play optimally. If the first player wins, print any winning first move.

Input

The first line contains the only integer q ($1 \le q \le 10^{13}$).

Please do not use the `%lld` specificator to read or write 64-bit integers in C++. It is preferred to use the `cin`, `cout` streams or the `%I64d` specificator.

Output

In the first line print the number of the winning player (1 or 2). If the first player wins then the second line should contain another integer — his first move (if the first player can't even make the first move, print 0). If there are multiple solutions, print any of them.

Examples

input
```
6
```
output
```
2
```

input
```
30
```
output
```
1
6
```

input
```
1
```
output
```
1
0
```

Note

Number 6 has only two non-trivial divisors: 2 and 3. It is impossible to make a move after the numbers 2 and 3 are written, so both of them are winning, thus, number 6 is the losing number. A player can make a move and write number 6 after number 30; 6, as we know, is a losing number. Thus, this move will bring us the victory.

B. Quantity of Strings

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

Just in case somebody missed it: this winter is totally cold in Nvodsk! It is so cold that one gets funny thoughts. For example, let's say there are strings with the length exactly n, based on the alphabet of size m. Any its substring with length equal to k is a palindrome. How many such strings exist? Your task is to find their quantity modulo 1000000007 ($10^9 + 7$). Be careful and don't miss a string or two!

Let us remind you that a string is a palindrome if it can be read the same way in either direction, from the left to the right and from the right to the left.

Input

The first and only line contains three integers: n, m and k ($1 \le n, m, k \le 2000$).

Output

Print a single integer — the number of strings of the described type modulo 1000000007 ($10^9 + 7$).

Examples

input
1 1 1

output
1

input
5 2 4

output
2

Note

In the first sample only one string is valid: "a" (let's denote the only letter of our alphabet as "a").

In the second sample (if we denote the alphabet letters as "a" and "b") the following strings are valid: "aaaaa" and "bbbbb".

C. Smart Cheater

time limit per test: 5 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

I guess there's not much point in reminding you that Nvodsk winters aren't exactly hot. That increased the popularity of the public transport dramatically. The route of bus 62 has exactly n stops (stop 1 goes first on its way and stop n goes last). The stops are positioned on a straight line and their coordinates are $0 = x_1 < x_2 < \dots < x_n$.

Each day exactly m people use bus 62. For each person we know the number of the stop where he gets on the bus and the number of the stop where he gets off the bus. A ticket from stop a to stop b ($a < b$) costs $x_b - x_a$ rubles. However, the conductor can choose no more than one segment NOT TO SELL a ticket for. We mean that conductor should choose C and D (C <= D) and sell a ticket for the segments $[A, C]$ and $[D, B]$, or not sell the ticket at all. The conductor and the passenger divide the saved money between themselves equally. The conductor's "untaxed income" is sometimes interrupted by inspections that take place as the bus drives on some segment of the route located between two consecutive stops. The inspector fines the conductor by c rubles for each passenger who doesn't have the ticket for this route's segment.

You know the coordinated of all stops x_i; the numbers of stops where the i-th passenger gets on and off, a_i and b_i ($a_i < b_i$); the fine c; and also p_i — the probability of inspection on segment between the i-th and the $i + 1$-th stop. The conductor asked you to help him make a plan of selling tickets that maximizes the mathematical expectation of his profit.

Input

The first line contains three integers n, m and c ($2 \le n \le 150\,000$, $1 \le m \le 300\,000$, $1 \le c \le 10\,000$).

The next line contains n integers x_i ($0 \le x_i \le 10^9$, $x_1 = 0$, $x_i < x_{i+1}$) — the coordinates of the stops on the bus's route.

The third line contains $n - 1$ integer p_i ($0 \le p_i \le 100$) — the probability of inspection in percents on the segment between stop i and stop $i + 1$.

Then follow m lines that describe the bus's passengers. Each line contains exactly two integers a_i and b_i ($1 \le a_i < b_i \le n$) — the numbers of stops where the i-th passenger gets on and off.

Output

Print the single real number — the maximum expectation of the conductor's profit. Your answer will be considered correct if its absolute or relative error does not exceed 10^{-6}.

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 $\frac{|a-b|}{\max(1,b)} \le 10^{-6}$.

Examples

input

```
3 3 10
0 10 100
100 0
1 2
2 3
1 3
```

output

```
90.000000000
```

input

```
10 8 187
0 10 30 70 150 310 630 1270 2550 51100
13 87 65 0 100 44 67 3 4
1 10
2 9
3 8
1 5
6 10
2 7
4 10
4 5
```

output

```
76859.990000000
```

Note

A comment to the first sample:

The first and third passengers get tickets from stop 1 to stop 2. The second passenger doesn't get a ticket. There always is inspection on the segment 1-2 but both passengers have the ticket for it. There never is an inspection on the segment 2-3, that's why the second passenger gets away with the cheating. Our total profit is $(0 + 90 / 2 + 90 / 2) = 90$.

D. Mission Impassable

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Market stalls now have the long-awaited game The Colder Scrools V: Nvodsk. The game turned out to be difficult as hell and most students can't complete the last quest ("We don't go to Nvodsk..."). That threatened winter exams. The rector already started to wonder whether he should postpone the winter exams till April (in fact, he wanted to complete the quest himself). But all of a sudden a stranger appeared at the door of his office. "Good afternoon. My name is Chuck and I solve any problems" — he said.

And here they are sitting side by side but still they can't complete the mission. The thing is, to kill the final boss one should prove one's perfect skills in the art of managing letters. One should be a real magician to do that. And can you imagine what happens when magicians start competing...

But let's put it more formally: you are given a string and a set of integers a_i. You are allowed to choose any substring that is a palindrome and delete it. At that we receive some number of points equal to a_k, where k is the length of the deleted palindrome. For some k, $a_k =$ -1, which means that deleting palindrome strings of such length is **forbidden**. After a substring is deleted, the remaining part "shifts together", that is, at no moment of time the string has gaps. The process is repeated while the string has at least one palindrome substring that can be deleted. All gained points are summed up.

Determine what maximum number of points can be earned.

"Oh" — said Chuck, raising from the chair, — "I used to love deleting palindromes, just like you, but one day I took an arrow in the Knee".

Input

The first line contains an integer l ($1 \le l \le 150$) — the length of the string.

The second line contains exactly l integers a_k ($-1 \le a_k \le 10^5$) — the points a player gains for deleting.

The third line contains exactly l lowercase Latin letters — the original string from which a player can delete palindromes. The line contains no other characters apart from the newline character at the end of the string.

Output

Print a single number — the maximum number of points one can gain if he plays on the given string.

Examples

input

```
7
-1 -1 -1 -1 -1 -1 -1
abacaba
```

output

```
0
```

input

```
7
1 -1 -1 -1 -1 -1 -1
abacaba
```

output

```
7
```

input

```
7
1 5 -1 -1 -1 -1 10
abacaba
```

output

```
16
```

Note

In the first sample we cannot delete any substring, so the best result is 0. In the second sample we are allowed to delete only those palindromes whose length equals 1, thus, if we delete the whole string, we get 7 points. In the third sample the optimal strategy is: first we delete character c, then string aa, then bb, and the last one aa. At that we get $1 + 3 * 5 = 16$ points.

E. Freezing with Style

time limit per test: 7 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

This winter is so... well, you've got the idea :-) The Nvodsk road system can be represented as n junctions connected with $n - 1$ bidirectional roads so that there is a path between any two junctions. The organizers of some event want to choose a place to accommodate the participants (junction v), and the place to set up the contests (junction u). Besides, at the one hand, they want the participants to walk about the city and see the neighbourhood (that's why the distance between v and u should be no less than l). On the other hand, they don't want the participants to freeze (so the distance between v and u should be no more than r). Besides, for every street we know its beauty — some integer from 0 to 10^9. Your task is to choose the path that fits in the length limits and has the largest average beauty. We shall define the average beauty as a median of sequence of the beauties of all roads along the path.

We can put it more formally like that: let there be a path with the length k. Let a_i be a non-decreasing sequence that contains exactly k elements. Each number occurs there exactly the number of times a road with such beauty occurs along on path. We will represent the path median as number $a_{\lfloor k/2 \rfloor}$, assuming that **indexation starting from zero** is used. $\lfloor x \rfloor$ — is number x, rounded down to the nearest integer.

For example, if $a = \{0, 5, 12\}$, then the median equals to 5, and if $a = \{0, 5, 7, 12\}$, then the median is number 7.

It is guaranteed that there will be at least one path with the suitable quantity of roads.

Input

The first line contains three integers n, l, r ($1 \le l \le r < n \le 10^5$).

Next $n - 1$ lines contain descriptions of roads of the Nvodsk, each line contains three integers a_i, b_i, c_i ($1 \le a_i, b_i \le n$, $0 \le c_i \le 10^9$, $a_i \ne b_i$) — junctions a_i and b_i are connected with a street whose beauty equals c_i.

Output

Print two integers — numbers of the junctions, where to accommodate the participants and set up the contests, correspondingly. If there are multiple optimal variants, print any of them.

Examples

input

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

output

```
4 1
```

input

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

output

```
6 3
```

input

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

output

```
4 3
```

input

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

```
4 5 8
5 8 2
3 6 3
2 7 4
```

output

```
5 1
```

Note

In the first sample all roads have the same beauty. That means that all paths of the positive length have the same median. Thus, any path with length from 3 to 4, inclusive will be valid for us.

In the second sample the city looks like that: 1 - 2 - 3 - 4 - 5 - 6. Two last roads are more valuable and we should choose any path that contains both of them and has the suitable length. It is either the path between 2 and 6 or the path between 3 and 6.