Problem Set 2 15-295 Spring 2019
Dynamic Programming

A. Robbing a Friend's House

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Sereja has decided to rob his enemies, when he's in his enemy's house, he's in a confusion as to which items to pick and which to drop, help him decide which items to pick based on the max profit based they can generate, keep in mind he can carry only a little weight and not everything.

Note: You cannot break an item, either pick the complete item, or don't pick it (0-1 property).

Input

First line contains two integers N, W

Second line contains N integers denoting the value of each item

Third line contains N integers denoting the weight of each item respectively

$0 \leq N \leq 10^3$; where N is the number of available items Sereja can rob

$1 \leq W \leq 10^4$; where W is the maximum weight Sereja can carry

$0 \leq V_i \leq 10^3$; value of the i^{th} item

$1 \leq W_i \leq 10^4$; weight of the i^{th} item

Output

Output the maximum profit Sereja can make

Example

standard input	standard output
3 50 60 100 120 10 20 30	220

B. Playing with Coins

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Given a value V, if we want to make change for V dollars, and we have infinite supply of each of C = C_1, C_2, .. , C_m valued coins, what is the minimum number of coins to make the change?

Input

First line contains two integers V, M followed by M integers in the second line

$0 \leq V \leq 10^5$; where V is the value we need to make the change of

$1 \leq M \leq 100$; where M is the number of coin denominations

$1 \leq C_i \leq 10^5$; value of the coin

Output

Output the minimum number of coins needed to make the the given value V

Example

standard input	standard output
8 3 1 4 5	2

C. Letter

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

Patrick has just finished writing a message to his sweetheart Stacey when he noticed that the message didn't look fancy. Patrick was nervous while writing the message, so some of the letters there were lowercase and some of them were uppercase.

Patrick believes that a message is *fancy* if any uppercase letter stands to the left of any lowercase one. In other words, this rule describes the strings where first go zero or more uppercase letters, and then — zero or more lowercase letters.

To make the message fancy, Patrick can erase some letter and add the same letter in the same place in the opposite case (that is, he can replace an uppercase letter with the lowercase one and vice versa). Patrick got interested in the following question: what minimum number of actions do we need to make a message fancy? Changing a letter's case in the message counts as one action. Patrick cannot perform any other actions.

Input

The only line of the input contains a non-empty string consisting of uppercase and lowercase letters. The string's length does not exceed 10^5.

Output

Print a single number — the least number of actions needed to make the message *fancy*.

Examples

input	Copy
PRuvetSTAaYA	
output	**Copy**
5	

input	Copy
OYPROSTIYAOPECHATALSYAPRIVETSTASYA	
output	**Copy**
0	

input	Copy
helloworld	
output	**Copy**
0	

D Ilya and Queries

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

Ilya the Lion wants to help all his friends with passing exams. They need to solve the following problem to pass the IT exam.

You've got string $s = s_1 s_2 \ldots s_n$ (n is the length of the string), consisting only of characters "." and "#" and m queries. Each query is described by a pair of integers l_i, r_i $(1 \le l_i < r_i \le n)$. The answer to the query l_i, r_i is the number of such integers i $(l_i \le i < r_i)$, that $s_i = s_{i+1}$.

Ilya the Lion wants to help his friends but is there anyone to help him? Help Ilya, solve the problem.

Input

The first line contains string s of length n $(2 \le n \le 10^5)$. It is guaranteed that the given string only consists of characters "." and "#".

The next line contains integer m $(1 \le m \le 10^5)$ — the number of queries. Each of the next m lines contains the description of the corresponding query. The i-th line contains integers l_i, r_i $(1 \le l_i < r_i \le n)$.

Output

Print m integers — the answers to the queries in the order in which they are given in the input.

Examples

input Copy

```
......
4
3 4
2 3
1 6
2 6
```

output Copy

```
1
1
5
4
```

input Copy

```
#..###
5
1 3
5 6
1 5
3 6
3 4
```

output Copy

```
1
1
2
2
0
```

E Consecutive Subsequence

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

You are given an integer array of length n.

You have to choose some subsequence of this array of maximum length such that this subsequence forms a increasing sequence of consecutive integers. In other words the required sequence should be equal to $[x, x+1, \ldots, x+k-1]$ for some value x and length k.

Subsequence of an array can be obtained by erasing some (possibly zero) elements from the array. You can erase any elements, not necessarily going successively. The remaining elements preserve their order. For example, for the array $[5, 3, 1, 2, 4]$ the following arrays are subsequences: $[3]$, $[5, 3, 1, 2, 4]$, $[5, 1, 4]$, but the array $[1, 3]$ is not.

Input

The first line of the input containing integer number n ($1 \le n \le 2 \cdot 10^5$) — the length of the array. The second line of the input containing n integer numbers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^9$) — the array itself.

Output

On the first line print k — the maximum length of the subsequence of the given array that forms an increasing sequence of consecutive integers.

On the second line print the sequence of the indices of the **any** maximum length subsequence of the given array that forms an increasing sequence of consecutive integers.

Examples

input	Copy
7 3 3 4 7 5 6 8	

output	Copy
4 2 3 5 6	

input Copy

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

output Copy

```
2
1 4
```

input Copy

```
4
10 9 8 7
```

output Copy

```
1
1
```

input Copy

```
9
6 7 8 3 4 5 9 10 11
```

output Copy

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

Note

All valid answers for the first example (as sequences of indices):

- $[1, 3, 5, 6]$
- $[2, 3, 5, 6]$

All valid answers for the second example:

- $[1, 4]$
- $[2, 5]$
- $[3, 6]$

All valid answers for the third example:

- $[1]$
- $[2]$
- $[3]$
- $[4]$

All valid answers for the fourth example:

- $[1, 2, 3, 7, 8, 9]$

F. Vasya and Array

time limit per test: 2 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

Vasya has got an array consisting of n integers, and two integers k and len in addition. All numbers in the array are either between 1 and k (inclusive), or equal to -1. The array is good if there is no segment of len consecutive **equal** numbers.

Vasya will replace each -1 with some number from 1 to k (inclusive) in such a way that the resulting array is good. Tell him the number of ways to do this replacement. Since the answer may be large, print it modulo 998244353.

Input

The first line contains three integers n, k and len ($1 \le n \le 10^5, 1 \le k \le 100, 1 \le len \le n$).

The second line contains n numbers — the array. Each number is either -1 or between 1 and k (inclusive).

Output

Print one integer — the number of ways to replace each -1 with some number from 1 to k (inclusive) so the array is good. The answer may be large, so print it modulo 998244353.

Examples

input Copy

```
5 2 3
1 -1 1 -1 2
```

output Copy

```
2
```

input Copy

```
6 3 2
1 1 -1 -1 -1 -1
```

output Copy

```
0
```

input Copy

```
10 42 7
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1
```

output Copy

```
645711643
```

Note

Possible answers in the first test:

1. $[1, 2, 1, 1, 2]$;
2. $[1, 2, 1, 2, 2]$.

There is no way to make the array good in the second test, since first two elements are equal.

There are too many answers in the third test, so we won't describe any of them.

G Ticket Draw

The concert promoters of the Bon Jovi Tour 2013 have decided to sell tickets for the concerts in lotteries. The rules are quite simple. For every concert, fans can apply online for tickets. In response they receive unique reservation numbers. It is important that for each concert the numbers distributed online are consecutive nonnegative integers starting with 0. Unfortunately, as the organizers tried to draw reservation numbers randomly, they discovered that the pseudo random generator they used is extremely slow. To minimize the number of calls to the generator, they invented a peculiar but fair enough way to distribute tickets.

As soon as the reservation for a concert is finished, the organizers ascertain the number of submissions M and draw one random integer Z from $\{0, \ldots, M-1\}$ (remember, fans get integers from 0 to $M-1$). Integer Z is the only object the organizers have to draw randomly! Finally, to complete the selection rules the organizers determine an integer $r > 0$ which has a direct impact on the number of selected tickets.

Now, using Z and r, tickets are selected deterministically as follows. For the reservation numbers $0, \ldots, M-1$ and the number Z, their decimal representations of length n are considered, where n is the length of the representation of $M-1$ without leading zeros. Thus, the decimal representations of the remaining numbers are padded on the left with leading zeros, if needed. If $z_1 \ldots z_n$ denotes such a representation for Z, then the holder of a number $a_1 \ldots a_n$ gets the ticket if and only if the strings $z_1 \ldots z_n$ and $a_1 \ldots a_n$ have a common contiguous substring of length r or more which starts at the same position. Speaking formally, he or she gets the ticket if there exists an index i, with $1 \leq i \leq n-r+1$, such that $z_i \ldots z_{i+r-1} = a_i \ldots a_{i+r-1}$. For example, if $Z = 56743$ and $r = 3$ then a fan with a reservation number 06740 gets a ticket, but a fan having 56143 does not.

Your task is to help the organizers to estimate, for given numbers M, Z and r, the exact number of tickets selected in such a way.

Input

The first line contains the number of concerts C, with $1 \leq C \leq 5000$. Then follow C lines, each containing three integers M, Z, and r, with $0 < M \leq 10^{18}$, $0 \leq Z \leq M-1$ and $r \geq 1$. You may safely assume that r is smaller or equal to the length of the decimal representation of $M-1$.

Output

For each concert, print one line containing the number of tickets selected during the ticket draw.

Sample Input

```
8
89 32 1
67 49 1
67 45 2
1000 23 1
1000 401 2
1000 54 2
3571 2 3
3571 976 3
```

Sample Output

```
18
15
1
271
19
19
13
12
```

Problem H
Fastest Speedrun

The classic video game "Prince of Python" comprises n levels, numbered from 1 to n. You are going to speedrun this game by finishing all of the levels as fast as possible, and you can beat them in any order that you want.

You enter each level equipped with one of $n + 1$ magical items. In the beginning you only have item 0 in your inventory. Once you beat a level, you get to keep the item numbered the same as that level. For example, on finishing level 5, you obtain a mighty *Gauntlet of 5 Fingers* you may equip thereafter instead of the less-acclaimed *Sword of 0 Damage* you always start out with.

Beating a level can take different amounts of time depending on which item you take into the level with you. Higher-numbered items are more powerful, so if playing by the rules it is always at least as fast to finish the level with a higher-numbered item as with a lower-numbered item.

However, each level also has a shortcut left in by the developers. The shortcut for a level can be accessed by applying a specific item in an unconventional way. By doing so you can finish the level as fast as, or even faster than, if you had used any of the other items.

How long will it take you to beat all of the levels of the game?

Input

The input consists of:

- One line containing an integer n ($1 \leq n \leq 2\,500$), the number of levels.
- n lines, describing the levels.

  The ith such line starts with two integers x_i and s_i ($0 \leq x_i \leq n$, $1 \leq s_i \leq 10^9$), the shortcut item for level i and the completion time for level i when using the shortcut.

  The remainder of the line has $n + 1$ integers $a_{i,0}, \ldots, a_{i,n}$ ($10^9 \geq a_{i,0} \geq a_{i,1} \geq \ldots \geq a_{i,n} \geq s_i$), where $a_{i,j}$ is the completion time for level i when playing by the rules using item j.

Output

Output the minimum time it takes to beat, in any order, all of the levels in the game.

Sample Input 1

```
3
1 1 40 30 20 10
3 1 95 95 95 10
2 1 95 50 30 20
```

Sample Output 1

```
91
```

Sample Input 2

```
4
4 4 5 5 5 5 5
4 4 5 5 5 5 5
4 4 5 5 5 5 5
4 4 5 5 5 5 5
```

Sample Output 2

```
17
```