A - delete .

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 100 points

Problem Statement

You are given a string S consisting of lowercase English letters and `.`. Find the string obtained by removing all `.` from S.

Constraints

- S is a string of length between 1 and 100, inclusive, consisting of lowercase English letters and `.`.

Input

The input is given from Standard Input in the following format:

```
S
```

Output

Print the string obtained by removing all `.` from S.

Sample Input 1

```
.v.
```

Sample Output 1

```
v
```

Removing all `.` from `.v.` yields v, so print v.

Sample Input 2

```
chokudai
```

Sample Output 2

```
chokudai
```

There are cases where S does not contain `.`.

Sample Input 3

```
...
```

Sample Output 3

```

```

There are also cases where all characters in S are `.`.

B - 3^A

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 200 points

Problem Statement

You are given a positive integer M. Find a positive integer N and a sequence of non-negative integers $A = (A_1, A_2, \ldots, A_N)$ that satisfy all of the following conditions:

- $1 \leq N \leq 20$
- $0 \leq A_i \leq 10 \, (1 \leq i \leq N)$
- $\displaystyle \sum_{i=1}^{N} 3^{A_i} = M$

It can be proved that under the constraints, there always exists at least one such pair of N and A satisfying the conditions.

Constraints

- $1 \leq M \leq 10^5$

Input

The input is given from Standard Input in the following format:

```
M
```

Output

Print N and A satisfying the conditions in the following format:

```
N
A_1 A_2 ... A_N
```

If there are multiple valid pairs of N and A, any of them is acceptable.

Sample Input 1

```
6
```

Sample Output 1

```
2
1 1
```

For example, with $N = 2$ and $A = (1, 1)$, we have $\displaystyle \sum_{i=1}^{N} 3^{A_i} = 3 + 3 = 6$, satisfying all conditions.

Another example is $N = 4$ and $A = (0, 0, 1, 0)$, which also satisfies the conditions.

Sample Input 2

```
100
```

Sample Output 2

```
4
2 0 2 4
```

Sample Input 3

```
59048
```

Sample Output 3

```
20
0 0 1 1 2 2 3 3 4 4 5 5 6 6 7 7 8 8 9 9
```

Note the condition $1 \leq N \leq 20$.

C - Count ABC Again

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 350 points

Problem Statement

You are given a string S of length N. You are also given Q queries, which you should process in order.

The i-th query is as follows:

- Given an integer X_i and a character C_i, replace the X_i-th character of S with C_i. Then, print the number of times the string `ABC` appears as a substring in S

Here, a **substring** of S is a string obtained by deleting zero or more characters from the beginning and zero or more characters from the end of S.
For example, `ab` is a substring of `abc`, but `ac` is not a substring of `abc`.

Constraints

- $3 \leq N \leq 2 \times 10^5$
- $1 \leq Q \leq 2 \times 10^5$
- S is a string of length N consisting of uppercase English letters.
- $1 \leq X_i \leq N$
- C_i is an uppercase English letter.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \quad Q \\
S \\
X_1 \quad C_1 \\
X_2 \quad C_2 \\
\vdots \\
X_Q \quad C_Q
\end{array}
$$

Output

Print Q lines. The i-th line $(1 \leq i \leq Q)$ should contain the answer to the i-th query.

Sample Input 1

```
7 4
ABCDABC
4 B
3 A
5 C
4 G
```

Sample Output 1

```
2
1
1
0
```

After processing each query, S becomes as follows.

- After the first query: $S =$ ABCBABC. In this string, ABC appears twice as a substring.
- After the second query: $S =$ ABABABC. In this string, ABC appears once as a substring.
- After the third query: $S =$ ABABCBC. In this string, ABC appears once as a substring.
- After the fourth query: $S =$ ABAGCBC. In this string, ABC appears zero times as a substring.

Sample Input 2

```
3 3
ABC
1 A
2 B
3 C
```

Sample Output 2

```
1
1
1
```

There are cases where S does not change through processing a query.

Sample Input 3

```
15 10
BBCCBCACCBACACA
9 C
11 B
5 B
11 B
4 A
8 C
8 B
5 B
7 B
14 B
```

Sample Output 3

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

D - Buildings

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 400 points

Problem Statement

There are N buildings, Building 1, Building $2, \ldots$, Building N, arranged in a line in this order. The height of Building i $(1 \leq i \leq N)$ is H_i.

For each $i = 1, 2, \ldots, N$, find the number of integers j $(i < j \leq N)$ satisfying the following condition:

- There is no building taller than Building j between Buildings i and j.

Constraints

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq H_i \leq N$
- $H_i \neq H_j$ $(i \neq j)$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$N$$
$$H_1 \quad H_2 \quad \ldots \quad H_N$$

Output

For each $i = 1, 2, \ldots, N$, let c_i be the number of j satisfying the condition. Print $c_1, c_2, \ldots, c_N$ in order, separated by spaces.

Sample Input 1

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

Sample Output 1

```
3 2 2 1 0
```

For $i = 1$, the integers j satisfying the condition are $2, 3$, and 5: there are three. (Between Buildings 1 and 4, there is a building taller than Building 4, which is Buil
so $j = 4$ does not satisfy the condition.) Therefore, the first number in the output is 3.

Sample Input 2

```
4
1 2 3 4
```

Sample Output 2

```
3 2 1 0
```

Sample Input 3

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

Sample Output 3

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

E - K-th Largest Connected Components

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 475 points

Problem Statement

There is an undirected graph with N vertices and 0 edges. The vertices are numbered 1 to N.

You are given Q queries to process in order. Each query is of one of the following two types:

- Type 1: Given in the format `1 u v`. Add an edge between vertices u and v.
- Type 2: Given in the format `2 v k`. Print the k-th largest vertex number among the vertices connected to vertex v. If there are fewer than k vertices connect print −1.

Constraints

- $1 \leq N, Q \leq 2 \times 10^5$
- In a Type 1 query, $1 \leq u < v \leq N$.
- In a Type 2 query, $1 \leq v \leq N, 1 \leq k \leq 10$.
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \quad Q \\
\mathrm{query}_1 \\
\mathrm{query}_2 \\
\vdots \\
\mathrm{query}_Q
\end{array}
$$

Here, query_i is the i-th query and is given in one of the following formats:

$$1 \quad u \quad v$$

$$2 \quad v \quad k$$

Output

Let q be the number of Type 2 queries. Print q lines. The i-th line should contain the answer to the i-th Type 2 query.

Sample Input 1

```
4 10
1 1 2
2 1 1
2 1 2
2 1 3
1 1 3
1 2 3
1 3 4
2 1 1
2 1 3
2 1 5
```

Sample Output 1

```
2
1
-1
4
2
-1
```

- In the first query, an edge is added between vertices 1 and 2.
- In the second query, two vertices are connected to vertex 1: 1 and 2. Among them, the 1-st largest vertex number is 2, which should be printed.
- In the third query, two vertices are connected to vertex 1: 1 and 2. Among them, the 2-nd largest vertex number is 1, which should be printed.
- In the fourth query, two vertices are connected to vertex 1: 1 and 2, which is fewer than 3, so print -1.
- In the fifth query, an edge is added between vertices 1 and 3.
- In the sixth query, an edge is added between vertices 2 and 3.
- In the seventh query, an edge is added between vertices 3 and 4.
- In the eighth query, four vertices are connected to vertex 1: $1, 2, 3, 4$. Among them, the 1-st largest vertex number is 4, which should be printed.
- In the ninth query, four vertices are connected to vertex 1: $1, 2, 3, 4$. Among them, the 3-rd largest vertex number is 2, which should be printed.
- In the tenth query, four vertices are connected to vertex 1: $1, 2, 3, 4$, which is fewer than 5, so print -1.

Sample Input 2

```
6 20
1 3 4
1 3 5
2 1 1
2 3 1
1 1 5
2 6 9
2 1 3
2 6 1
1 4 6
2 2 1
2 6 2
2 4 7
1 1 4
2 6 2
2 3 4
1 2 5
2 4 1
1 1 6
2 3 3
2 1 3
```

Sample Output 2

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

F - Teleporting Takahashi 2

Time Limit: 3 sec / Memory Limit: 1024 MB

Score : 525 points

Problem Statement

There is a simple directed graph G with N vertices and $N + M$ edges. The vertices are numbered 1 to N, and the edges are numbered 1 to $N + M$.

Edge $i\,(1 \leq i \leq N)$ goes from vertex i to vertex $i + 1$. (Here, vertex $N + 1$ is considered as vertex 1.)
Edge $N + i\,(1 \leq i \leq M)$ goes from vertex X_i to vertex Y_i.

Takahashi is at vertex 1. At each vertex, he can move to any vertex to which there is an outgoing edge from the current vertex.

Compute the number of ways he can move exactly K times.

That is, find the number of integer sequences $(v_0, v_1, \dots, v_K)$ of length $K + 1$ satisfying all of the following three conditions:

- $1 \leq v_i \leq N$ for $i = 0, 1, \dots, K$.
- $v_0 = 1$.
- There is a directed edge from vertex v_{i-1} to vertex v_i for $i = 1, 2, \dots, K$.

Since this number can be very large, print it modulo 998244353.

Constraints

- $2 \leq N \leq 2 \times 10^5$
- $0 \leq M \leq 50$
- $1 \leq K \leq 2 \times 10^5$
- $1 \leq X_i, Y_i \leq N, X_i \neq Y_i$
- All of the $N + M$ directed edges are distinct.
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \quad M \quad K \\
X_1 \quad Y_1 \\
X_2 \quad Y_2 \\
\vdots \\
X_M \quad Y_M
\end{array}
$$

Output

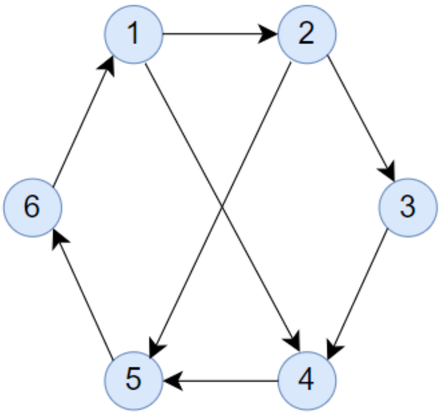
Print the count modulo 998244353.

Sample Input 1

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

Sample Output 1

```
5
```



The above figure represents the graph G. There are five ways for Takahashi to move:

- Vertex 1 $\to$ Vertex 2 $\to$ Vertex 3 $\to$ Vertex 4 $\to$ Vertex 5 $\to$ Vertex 6
- Vertex 1 $\to$ Vertex 2 $\to$ Vertex 5 $\to$ Vertex 6 $\to$ Vertex 1 $\to$ Vertex 2
- Vertex 1 $\to$ Vertex 2 $\to$ Vertex 5 $\to$ Vertex 6 $\to$ Vertex 1 $\to$ Vertex 4
- Vertex 1 $\to$ Vertex 4 $\to$ Vertex 5 $\to$ Vertex 6 $\to$ Vertex 1 $\to$ Vertex 2
- Vertex 1 $\to$ Vertex 4 $\to$ Vertex 5 $\to$ Vertex 6 $\to$ Vertex 1 $\to$ Vertex 4

Sample Input 2

```
10 0 200000
```

Sample Output 2

```
1
```

Sample Input 3

```
199 10 1326
122 39
142 49
164 119
197 127
188 145
69 80
6 120
24 160
18 154
185 27
```

Sample Output 3

```
451022766
```

G - Ax + By < C

Time Limit: 3 sec / Memory Limit: 1024 MB

Score : 625 points

Problem Statement

You are given three length-N sequences of positive integers: $A = (A_1, A_2, \ldots, A_N)$, $B = (B_1, B_2, \ldots, B_N)$, and $C = (C_1, C_2, \ldots, C_N)$.

Find the number of pairs of positive integers (x, y) that satisfy the following condition:

- $A_i \times x + B_i \times y < C_i$ for all $1 \leq i \leq N$.

It can be proved that the number of such pairs of positive integers satisfying the condition is finite.

You are given T test cases, each of which should be solved.

Constraints

- $1 \leq T \leq 2 \times 10^5$
- $1 \leq N \leq 2 \times 10^5$
- $1 \leq A_i, B_i, C_i \leq 10^9$
- The sum of N over all test cases is at most 2×10^5.
- All input values are integers.

Input

The input is given from Standard Input in the following format. Here, case_i refers to the i-th test case.

$$\begin{array}{l} T \\ \mathrm{case}_1 \\ \mathrm{case}_2 \\ \vdots \\ \mathrm{case}_T \end{array}$$

Each test case is given in the following format:

$$\begin{array}{l} N \\ A_1 \quad B_1 \quad C_1 \\ A_2 \quad B_2 \quad C_2 \\ \vdots \\ A_N \quad B_N \quad C_N \end{array}$$

Output

Print T lines. The i-th line $(1 \leq i \leq T)$ should contain the answer for case_i.

Sample Input 1

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

Sample Output 1

```
2
0
```

In the first test case, there are two valid pairs of integers: $(x, y) = (1, 1), (2, 1)$. Thus, the first line should contain 2.

In the second test case, there are no valid pairs of integers. Thus, the second line should contain 0.

Sample Input 2

```
3
7
138 16011 918976
5478 7748 499926
5234 17727 748589
1157 10511 643136
31200 3005 721285
28839 14469 798851
1933 5378 864127
9
17775 1665 386430
37001 863 922418
9756 4182 746671
12379 9106 807578
3984 4049 640539
25333 9869 780810
20372 7000 688738
16107 11974 827227
10779 10531 770510
5
4916 14132 460944
11856 45422 610561
56014 18216 825793
10363 6220 945356
37418 33866 851593
```

Sample Output 2

```
660
995
140
```