A. Determinant

Bobo learned the definition of determinant $\det(A)$ of matrix A in ICPCCamp. He also knew determinant can be computed in $O(n^3)$ using Gaussian Elimination.

Bobo has an $(n-1) \times n$ matrix B he would like to find $\det(B_j)$ modulo (10^9+7) for all $j \in \{1, 2, \dots, n\}$ where B_j is the matrix after removing the j-th column from B.

Input

The input contains zero or more test cases and is terminated by end-of-file. For each test case:

The first line contains an integer n. The i-th of following n lines contains n integers $B_{i,1}, B_{i,2}, \dots, B_{i,n}$.

- $2 \le n \le 200$
- $0 \le B_{i,j} < 10^9 + 7$
- The sum of n does not exceed 2000.

Output

For each case, output n integers which denote the result.

Sample Input

```
2
2 0
3
1 2 0
6 3 1
```

Sample Output

```
0 2
2 1 999999998
```

Note

For the second sample,

$$\det(B_1) = \det \begin{bmatrix} 2 & 0 \\ 3 & 1 \end{bmatrix} = 2.$$

B. Roads

In ICPCCamp, there are n towns conveniently labeled with $1, 2, \ldots, n$ and m bidirectional roads planned to be built. The i-th road will be built between cities a_i and b_i with cost c_i. The builders in ICPCCamp will build the $(n-1)$ roads with the least total cost to connect any of two cities directly or indirectly.

Bobo, the mayor of ICPCCamp is going to remove some of the roads from the construction plan. He would like to know the minimum number roads to be removed to *strictly increases* the total cost.

Note that the total cost is considered as $+\infty$ if no valid $(n-1)$ roads exist after removing. It is also counted as "total cost strictly increases".

Input

The input contains zero or more test cases and is terminated by end-of-file. For each test case:

The first line contains two integers n and m. The i-th of the following m lines contains a_i, b_i, c_i.

- $2 \le n \le 50$
- $n-1 \le m \le n^2$
- $1 \le a_i, b_i \le n$
- $1 \le c_i \le 10^9$
- Any two cities will be connected if all m roads are built.
- The sum of n does not exceed 10^3.

Output

For each case, output an integer which denotes the result.

Sample Input

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

Sample Output

```
1
1
2
3
```

C. Intersection

Bobo has two sets of integers $A = \{a_1, a_2, \dots, a_n\}$ and $B = \{b_1, b_2, \dots, b_n\}$. He says that $x \in \text{span}(A)$ (or $\text{span}(B)$) if and only if there exists a subset of A (or B) whose exclusive-or sum equals to x.

Bobo would like to know the number of x where $x \in \text{span}(A)$ and $x \in \text{span}(B)$ hold simultaneously.

Input

The input contains zero or more test cases and is terminated by end-of-file. For each test case:

The first line contains an integer n. The second line contains n integers $a_1, a_2, \dots, a_n$. The third line contains n integers $b_1, b_2, \dots, b_n$.

- $1 \le n \le 50$
- $0 \le a_i, b_i < 2^{60}$
- The number of test cases does not exceed 5000.

Output

For each case, output an integer which denotes the result.

Sample Input

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

Sample Output

```
1
4
```

Note

For the second sample, $\text{span}(A) = \text{span}(B) = \{0, 1, 2, 3\}$.

D. Super Resolution

Bobo has an $n \times m$ picture consists of black and white pixels. He loves the picture so he would like to scale it $a \times b$ times. That is, to replace each pixel with $a \times b$ block of pixels with the same color (see the example for clarity).

Input

The input contains zero or more test cases and is terminated by end-of-file. For each test case,

The first line contains four integers n, m, a, b. The i-th of the following n lines contains a binary string of length m which denotes the i-th row of the original picture. Character "`0`" stands for a white pixel while the character "`1`" stands for black one.

- $1 \le n, m, a, b \le 10$
- The number of tests cases does not exceed 10.

Output

For each case, output $n \times a$ rows and $m \times b$ columns which denote the result.

Sample Input

```
2 2 1 1
10
11
2 2 2 2
10
11
2 2 2 3
10
11
```

Sample Output

```
10
11
1100
1100
1111
1111
111000
111000
111111
111111
```

E. Partial Sum

Bobo has a integer sequence $a_1, a_2, \ldots, a_n$ of length n. Each time, he selects two ends $0 \le l < r \le n$ and add $|\sum_{j=l+1}^{r} a_j| - C$ into a counter which is zero initially. He repeats the selection for *at most* m times.

If each end can be selected at most once (either as left or right), find out the maximum sum Bobo may have.

Input

The input contains zero or more test cases and is terminated by end-of-file. For each test case:

The first line contains three integers n, m, C. The second line contains n integers $a_1, a_2, \ldots, a_n$.

- $2 \le n \le 10^5$
- $1 \le 2m \le n + 1$
- $|a_i|, C \le 10^4$
- The sum of n does not exceed 10^6.

Output

For each test cases, output an integer which denotes the maximum.

Sample Input

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

Sample Output

```
3
4
2
0
```

F. Longest Common Subsequence

Bobo has a sequence $A = (a_1, a_2, \dots, a_n)$ of length n. He would like to know $f(0), f(1), f(2)$ and $f(3)$ where $f(k)$ denotes the number of integer sequences $X = (x_1, x_2, x_3)$ where:

- $1 \le x_1, x_2, x_3 \le m$;
- The length of longest common subsequence of A and X is exactly k.

Note:

- u is a subsequence of v if and only if u can be obtained by removing some of the entries from v (possibly none).
- u is common subsequence of v and w if and only if u is subsequence of v and w.

Input

The input contains zero or more test cases and is terminated by end-of-file. For each case,

The first line contains two integers n, m. The second line contains n integers $a_1, a_2, \dots, a_n$.

- $1 \le n \le 200$
- $1 \le m, a_1, a_2, \dots, a_n \le 10^6$
- The number of tests cases does not exceed 10.

Output

For each case, output four integers which denote $f(0), f(1), f(2), f(3)$.

Sample Input

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

Sample Output

```
1 14 11 1
0 1 17 9
```

Note

For the second sample, $X = (3, 3, 3)$ is the only sequence that the length of longest common subsequence of A and X is 1. Thus, $f(1) = 1$.

G. Parentheses

Bobo has a very long sequence divided into n consecutive groups. The i-th group consists of l_i copies of character c_i where c_i is either "`(`" or "`)`".

As the sequence may not be valid parentheses sequence, Bobo can change a character in the i-th group from "`(`" to "`)`" (and vice versa) with cost d_i. He would like to know the minimum cost to transform the sequence into a valid one.

Note:

- An empty string is valid.
- If S is valid, (S) is valid.
- If U, V are valid, UV is valid.

Input

The input contains zero or more test cases and is terminated by end-of-file. For each test case:

The first line contains an integer n. The i-th of the following n lines contains l_i, c_i, d_i.

- $1 \le n \le 10^5$
- $1 \le l_1 + l_2 + \cdots + l_n \le 10^9$
- $l_1 + l_2 + \cdots + l_n$ is even.
- $1 \le d_i \le 10^9$
- The sum of n does not exceed 10^6.

Output

For each case, output an integer which denotes the result.

Sample Input

```
4
1 ( 1
1 ( 2
1 ( 3
1 ) 4
2
500000000 ) 1000000000
500000000 ( 1000000000
```

Sample Output

```
2
500000000000000000
```

Note

For the first sample, Bobo should change only the character in the second group.

For the second sample, Bobo should change half of characters in both groups.

H. Highway

In ICPCCamp there were n towns conveniently numbered with $1, 2, \ldots, n$ connected with $(n-1)$ roads. The i-th road connecting towns a_i and b_i has length c_i. It is guaranteed that any two cities reach each other using only roads.

Bobo would like to build $(n-1)$ highways so that any two towns reach each using *only highways*. Building a highway between towns x and y costs him $\delta(x, y)$ cents, where $\delta(x, y)$ is the length of the shortest path between towns x and y using roads.

As Bobo is rich, he would like to find the most expensive way to build the $(n-1)$ highways.

Input

The input contains zero or more test cases and is terminated by end-of-file. For each test case:

The first line contains an integer n. The i-th of the following $(n-1)$ lines contains three integers a_i, b_i and c_i.

- $1 \leq n \leq 10^5$
- $1 \leq a_i, b_i \leq n$
- $1 \leq c_i \leq 10^8$
- The number of test cases does not exceed 10.

Output

For each test case, output an integer which denotes the result.

Sample Input

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

Sample Output

```
19
15
```

I. Strange Optimization

Bobo is facing a strange optimization problem. Given n, m, he is going to find a real number α such that $f(\frac{1}{2} + \alpha)$ is maximized, where $f(t) = \min_{i,j \in \mathbb{Z}} |\frac{i}{n} - \frac{j}{m} + t|$. Help him!

Note: It can be proved that the result is always rational.

Input

The input contains zero or more test cases and is terminated by end-of-file.

Each test case contains two integers n, m.

- $1 \le n, m \le 10^9$
- The number of tests cases does not exceed 10^4.

Output

For each case, output a fraction p/q which denotes the result.

Sample Input

```
1 1
1 2
```

Sample Output

```
1/2
1/4
```

Note

For the first sample, $\alpha = 0$ maximizes the function.

J. Similar Subsequence

For given sequence $A = (a_1, a_2, \dots, a_n)$, a sequence $S = (s_1, s_2, \dots, s_n)$ has *shape* A if and only if:

- $s_i = \min\{s_i, s_{i+1}, \dots, s_n\}$ for all $a_i = 0$;
- $s_i = \max\{s_i, s_{i+1}, \dots, s_n\}$ for all $a_i = 1$.

Given sequence $B = (b_1, b_2, \dots, b_m)$, Bobo would like to know the number of subsequences of length n which have *shape* A modulo $(10^9 + 7)$.

Input

The input contains zero or more test cases and is terminated by end-of-file. For each test case:

The first line contains two integers n and m.

The second line contains n integers $a_1, a_2, \dots, a_n$.

The thrid line contains m integers $b_1, b_2, \dots, b_m$.

- The number of test cases does not exceed 10.
- $1 \le n \le 20$
- $1 \le m \le 500$
- $0 \le a_i \le 1$
- $1 \le b_i \le m$
- $b_1, b_2, \dots, b_m$ are distinct.

Output

For each case, output an integer which denotes the number of subsequences modulo $(10^9 + 7)$.

Sample Input

```
2 3
0 0
1 2 3
3 5
1 0 1
4 1 3 2 5
```

Sample Output

```
3
2
```

Note

For the first sample, all three subsequences of length 2 are of shape A.