A. Simple Arithmetic

Given a and b which both fit in 64-bit signed integers, find $\lfloor \frac{a}{b} \rfloor$ where $\lfloor x \rfloor$ denotes the largest integer which is not larger than x.

Input

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

Each test case contains two integers a, b.

- $-2^{63} \le a, b < 2^{63}$
- $b \neq 0$
- The number of tests cases does not exceed 10^4.

Output

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

Sample Input

```
3 2
3 -2
-9223372036854775808 1
-9223372036854775808 -1
```

Sample Output

```
1
-2
-9223372036854775808
9223372036854775808
```

B. Broken Counter

Bobo made a very long sequence of numbers $(a_1, a_2, \ldots, a_n)$ in ICPCCamp where $a_i = f(i)$ and

$$f(i) = \begin{cases} 1 & i \le 0 \\ A \cdot f(i-1) + B \cdot f(i-m) & i > 0 \end{cases}.$$

Now he wants to ask q questions where the i-th question is to compute the sum of $a_{l_i}, a_{l_i+1}, \ldots, a_{r_i}$. Unfortunately, the only tool which Bobo can utilize is an old broken 4-bit counter. While trying to answer the i-th question, Bobo will set the counter to 0, and add numbers to the counter in the order of $a_{l_i}, a_{l_i+1}, \ldots, a_{r_i}$.

As the counter is broken, adding the number a to a counter holding value x yields $[(x \oplus w_i) + a] \bmod 16$. Note that "$\oplus$" stands for bitwise exclusive or (XOR).

Bobo would like to know the final result.

Special Note: The time limit is tight so that some optimization might be necessary. Try to solve the problem as late as possible.

Input

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

The first line contains five integers n, m, A, B, q. The i-th of the following q lines contains three integers l_i, r_i and w_i.

- $1 \le n \le 10^8$
- $2 \le m \le 10^5$
- $0 \le A, B, w_i < 16$
- $1 \le q \le 13$
- $1 \le l_i \le r_i \le n$
- The number of test cases does not exceed 10.

Output

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

Sample Input

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

Sample Output

```
2
15
```

C. 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 $n \times n$ matrix B he would like to find $\det(B_{i,j})$ modulo $(10^9 + 7)$ for all $i, j \in \{1, 2, \ldots, n\}$ where $B_{i,j}$ is the matrix after removing the i-th row and j-th column from B.

It is guaranteed that the each column sum of B is a multiple of $(10^9 + 7)$.

Input

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

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

- $2 \le n \le 500$
- $0 \le B_{i,j} < 10^9 + 7$
- The sum of n does not exceed 5000.

Output

For each case, output n rows where the i-th row contains n integers $\det(B_{i,1}), \det(B_{i,2}), \ldots, \det(B_{i,n})$ modulo $(10^9 + 7)$.

Sample Input

```
2
0 1
0 1000000006
```

Sample Output

```
1000000006 0
1 0
```

D. Dynamic Graph

Bobo has a directed acyclic graph (DAG) with n nodes and m edges whose nodes is conveniently labeled with $1, 2, \ldots, n$. All nodes are white initially.

Bobo performs q operations subsequently. The i-th operation is to change the node v_i from white to black or vice versa.

After each operation, he would like to know the number of pairs of nodes (u, v) that u, v are both white and there exists a path from u to v passing only white nodes.

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 and q.

The i-th of the following m lines contains two integers a_i, b_i.

The i-th of the last q lines contains an integer v_i.

- $2 \le n \le 300$
- $1 \le m \le \frac{n(n-1)}{2}$
- $1 \le q \le 300$
- $1 \le a_i < b_i \le n$
- $1 \le v_i \le n$
- The number of tests cases does not exceed 10.

Output

For each operation, output an integer which denotes the number of pairs.

Sample Input

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

Sample Output

```
1
3
```

E. Longest Increasing Subsequence

Bobo learned how to compute Longest Increasing Subsequence (LIS) in $O(n \log n)$ in ICPCCamp.

For those who did not attend ICPCCamp as Bobo, recall $\text{LIS}(a_1, a_2, \dots, a_n)$ is defined as $f[1]^2 \oplus f[2]^2 \oplus \dots \oplus f[n]^2$ where $\oplus$ denotes the exclusive-or (XOR) and f is calculated as follows.

```
for i in [1, 2, ..., n]
  f[i] = 1
  for j in [1, 2, ..., i - 1]
    if a[j] < a[i] then
      f[i] = max(f[i], f[j] + 1)
```

Given sequence $A = (a_1, a_2, \dots, a_n)$, Bobo would like to find $\text{LIS}(B_1), \text{LIS}(B_2), \dots, \text{LIS}(B_n)$ where B_i is the sequence after removing the i-th element from A.

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 $a_1, a_2, \dots, a_n$.

- $2 \le n \le 5000$
- $1 \le a_i \le n$
- The number of test cases does not exceed 10.

Output

For each case, output n integers which denote $\text{LIS}(B_1), \text{LIS}(B_2), \dots, \text{LIS}(B_n)$.

Sample Input

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

Sample Output

```
5 13 0 8 0
```

F. Simple Algebra

Given function $f(x, y) = ax^2 + bxy + cy^2$, check if $f(x, y) \geq 0$ holds for all $x, y \in \mathbb{R}$.

Input

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

Each test case contains three integers a, b, c.

- $-10 \leq a, b, c \leq 10$
- The number of tests cases does not exceed 10^4.

Output

For each case, output "`Yes`" if $f(x, y) \geq 0$ always holds. Otherwise, output "`No`".

Sample Input

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

Sample Output

```
Yes
No
Yes
```

G. 2017

Given a, b, c, d, find out the number of pairs of integers (x, y) where $a \le x \le b, c \le y \le d$ and $x \cdot y$ is multiple of 2017.

Input

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

Each test case contains four integers a, b, c, d.

- $1 \le a \le b \le 10^9, 1 \le c \le d \le 10^9$
- The number of tests cases does not exceed 10^4.

Output

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

Sample Input

```
1 2017 1 2016
1 1000000000 1 1000000000
```

Sample Output

```
2016
991324197233775
```

H. Roads

In ICPCCamp there were n towns conveniently numbered with $1, 2, \ldots, n$ connected with m roads.

Bobo would like to know the number of ways to keep only $(n - 1)$ roads so that the towns remain connected.

Note that the towns are connected if and only any two cities reach each other.

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 two integers a_i and b_i which denotes a road between cities a_i and b_i.

- $1 \leq n \leq 10^5$
- $n < m < n + 100$
- $1 \leq a_i, b_i \leq n$
- The towns are connected with m roads.
- The number of test cases does not exceed 10.

Output

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

Sample Input

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

Sample Output

```
3
9
```

I. Strange Prime

Bobo finds a strange prime $P = 10^{10} + 19$ in ICPCCamp, and he decides to write n integers $x_1, x_2, \ldots, x_n$ whose sum is a multiple of P, while x_i should satisfy $0 \le x_i < P - a_i$ for given $a_1, a_2, \ldots, a_n$.

Bobo would like to know the number of different ways to write $x_1, x_2, \ldots, x_n$ 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 an integer n. The second line contains n integers $a_1, a_2, \ldots, a_n$.

- $1 \le n \le 10^5$
- $0 \le a_i \le 10^5$
- The sum of n does not exceed 10^6.

Output

For each case, output an integer which denotes the number of different ways.

Sample Input

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

Sample Output

```
999999956
2756
```

J. Skewness

Bobo has a matrix A with n rows and n columns.

For submatrix with upper-left corner (x_1, y_1) and lower-right corner (x_2, y_2) $(1 \le x_1 \le x_2 \le n, 1 \le y_1 \le y_2 \le n)$, he defined its *skewness* $S(x_1, y_1, x_2, y_2) = \left(\sum_{i=x_1}^{x_2} \sum_{j=y_1}^{y_2} A_{i,j}\right)^3$.

Bobo would like to know the sum of *skewness* of all submatrices 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 an integer n.

The i-th of following n lines contains n integers $A_{i,1}, A_{i,2}, \dots, A_{i,n}$.

- $1 \le n \le 1000$
- $0 \le A_{i,j} \le 10^9$
- The number of test cases does not exceed 10.

Output

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

Sample Input

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

Sample Output

```
14
448
```

K. 2017 Revenge

Bobo has n integers $a_1, a_2, \ldots, a_n$. He would like to choose some of the integers and calculate their product (the product of the empty set is defined as 1).

Bobo would like to know the number of products whose remainder divided by 2017 is r. As the exact number is too large, he only asks for the number modulo 2.

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, r.

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

- $1 \le n \le 2 \times 10^6$
- $1 \le r, a_1, a_2, \ldots, a_n < 2017$
- The sum of n does not exceed 2×10^6.

Output

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

Sample Input

```
3 6
2 3 4
4 1
1 1 2016 2016
```

Sample Output

```
1
0
```

L. Nice Trick

Given n integers $a_1, a_2, \ldots, a_n$, Bobo knows how to compute the *sum of triples*

$$S_3 = \sum_{1 \le i < j < k \le n} a_i a_j a_k.$$

It follows that

$$S_3 = \frac{(\sum_{1 \le i \le n} a_i)^3 - 3(\sum_{1 \le i \le n} a_i^2)(\sum_{1 \le i \le n} a_i) + 2(\sum_{1 \le i \le n} a_i^3)}{6}.$$

Bobo would like to compute the *sum of quadrangles*

$$\left(\sum_{1 \le i < j < k < l \le n} a_i a_j a_k a_l \right) \bmod (10^9 + 7).$$

Input

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

The first line contains an integer n.

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

- $1 \le n \le 10^5$
- $0 \le a_i \le 10^9$
- The number of tests cases does not exceed 10.

Output

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

Sample Input

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

Sample Output

```
0
24
274
```