

标题：A、Circulant Matrix | 时间限制：1 秒 | 内存限制：256M

Niuniu has just learned how to use Gaussian elimination to solve systems of linear equations.

Given n and $a[i]$, where n is a power of 2, let's consider an $n \times n$ matrix A .

The index of $A[i][j]$ and $a[i]$ are numbered from 0.

The element $A[i][j]$ satisfies $A[i][j] = a[i] \text{ xor } a[j]$,

https://en.wikipedia.org/wiki/Bitwise_operation#XOR

Let $p = 1000000007$.

Consider the equation

$$A x = b \pmod{p}$$

where A is an $n \times n$ matrix, and x and b are both $n \times 1$ row vector.

Given n , $a[i]$, $b[i]$, you need to solve the x .

For example, when $n = 4$, the equations look like

$$A[0][0] * x[0] + A[0][1] * x[1] + A[0][2] * x[2] + A[0][3] * x[3] = b[0] \pmod{p}$$

$$A[1][0] * x[0] + A[1][1] * x[1] + A[1][2] * x[2] + A[1][3] * x[3] = b[1] \pmod{p}$$

$$A[2][0] * x[0] + A[2][1] * x[1] + A[2][2] * x[2] + A[2][3] * x[3] = b[2] \pmod{p}$$

$$A[3][0] * x[0] + A[3][1] * x[1] + A[3][2] * x[2] + A[3][3] * x[3] = b[3] \pmod{p}$$

and the matrix A can be decided by the array a .

It is guaranteed that there is a unique solution x for these equations.

输入描述:

The first line contains an integer, which is n .

The second line contains n integers, which are the array a .

The third line contains n integers, which are the array b .

$$1 \leq n \leq 262144$$

$$0 \leq a[i] < p$$

$$0 \leq b[i] < p$$

输出描述:

The output should contains n lines.

The i -th(index from 0) line should contain $x[i]$.

$x[i]$ should satisfy $0 \leq x[i] < p$.

示例 1

输入

4

1 10 100 1000

1234 2143 3412 4321

输出

4

3

2

1

标题：B、Enumeration not optimization | 时间限制：1 秒 | 内存限制：256M

Maybe you have heard NOIP 2017 Day 2 Problem 2 Treasure.

You can find the problem at Luogu(<https://www.luogu.org/problemnew/show/P3959>).

The data is very weak. Many contestants accepted this problem with search solutions or wrong solutions.

Let's consider the enumeration version of this problem.

We want to find the sum of weight of all rooted spanning tree.

The weight of a rooted spanning tree is defined as follows.

$$\sum_{e=\{x,y\}} w_e \times \max(d_x, d_y)$$

Enumerate all edges in the tree.

The contribution of an edge is its length times its depth in the rooted tree.

The depth of a vertex is the number of edges from the root to the vertex.

As the answer might be very large, you only need to output the answer mod 1000000007.

输入描述:

The first line contains two integers n, m, which are the number of vertices and the number of edges.

In the following m lines, each line contains three integers x, y, z, which means there is an edge between x and y, whose length is z.

$1 \leq n \leq 10$

$1 \leq m \leq 1000$

$1 \leq x \leq n$

$1 \leq y \leq n$

$1 \leq z \leq 5000$

输出描述:

You should output one line, which contains the answer.

备注:

There might be multiple edges between two vertices.

示例 1

输入

4 5

1 2 1

1 3 3

1 4 1

2 3 4

3 4 1

输出

303

示例 2

输入

4 5

1 2 1

1 3 3

1 4 1

2 3 4

3 4 2

输出

336

标题: C、All in | 时间限制: 1 秒 | 内存限制: 256M

Niuniu likes gambling.

Team A and B will play $2n-1$ matches.

Niuniu wants to bet 2^{2n-1} that team A wins the entire series.

In other words, if A wins n or more matches, Niuniu will gain 2^{2n-1}

If A loses(B wins) n or more matches, Niuniu will lose 2^{2n-1} (gain -2^{2n-1})

However, the banker does not allow such a bet. Niuniu can only bet on individual matches.

The winning percentage of both teams is 0.5. All matches are independent of each other.

Niuniu can bet on the $(i+1)$ -th match after seeing the results of the first i games.

Your program need read n and output the bet on the first match.

Then read the result of the first match, output the bet on the second match.

Then read the result of the second match, output the bet on the third match.

....

Finally, read the result of the n -th match, and find that team A wins or loses the entire series, end the program.

The answer can be uniquely determined, and this is not a interactive problem.

As the result might be very large, you should output the result mod 1000000007.

输入描述:

The first line contains an integer, which is n .

The second line contains the results of contests, which are 0s and 1s.

0 means team A wins, and 1 means team A loses(team B wins)

It is guaranteed this is a legal process.

Once the number of 0 or 1 reaches n , there will be no more input.

$1 \leq n \leq 100000$

输出描述:

You should output the bet for each contest in a new line.

The number of output is the same as the second line of input.

示例 1

输入

2

0 1 1

输出

4

4

8

说明

In fact, for $n = 2$, the bets are always 4 4 8, no matter what the outcome of the matches.

If team A (or B) wins the first two matches, Niuniu will gain $4 + 4 = 8$ (or lose $4 + 4 = 8$).

If team A and B both win one of the first two matches, Niuniu gains $4 - 4 = 0$, and will bet 8 on the third match.

In all situations, Niuniu will gain 8 if team A wins the entire series, and will lose 8 if team A loses the entire series.

示例 2

输入

3

1 1 0 0 1

输出

12

12

8

16

32

说明

The sum of all bets (with sign) is -2^{2n-1} or 2^{2n-1}

$$(-12) + (-12) + (8) + (16) + (-32) = -32 = -2^5$$

示例 3

输入

3

0 1 0 0

输出

12

12

16

16

说明

$$(12) + (-12) + (16) + (16) = 32 = 2^5$$

You can make the third bet, based on the first two result.

标题: D、The number of circuits | 时间限制: 1 秒 | 内存限制: 32M

Niuniu likes traveling. Now he will travel on a special graph.

Given k and n , The directed graph contains n vertices, which are numbered from 0 to $n - 1$.

For the vertex i , and for $1 \leq j \leq k$, there is a directed edge from vertex i to vertex $((i + j) \% n)$.

We want to know the number of (directed) cycles, that pass each directed edge exactly once.

As the answer might be very large, you only need to output the answer mod 1000000007.

输入描述:

The first and only line contains two integers, which are k and n .

$1 \leq k \leq 5$

$2k+1 \leq n \leq 10^9$

输出描述:

The first and only line contains the answer.

示例 1

输入

2 5

输出

11

说明

The answer is not 22.

0 1 2 3 4 0 2 4 1 3 0

is the same as

0 2 4 1 3 0 1 2 3 4 0

标题: E、Music Game | 时间限制: 1 秒 | 内存限制: 256M

Niuniu likes to play OSU!

We simplify the game OSU to the following problem.

Given n and m , there are n clicks. Each click may success or fail.

For a continuous success sequence with length X , the player can score X^m .

The probability that the i -th click success is $p[i]/100$.

We want to know the expectation of score.

As the result might be very large (and not integral), you only need to output the result mod 1000000007.

输入描述:

The first line contains two integers, which are n and m .

The second line contains n integers. The i -th integer is $p[i]$.

$1 \leq n \leq 1000$

$1 \leq m \leq 1000$

$0 \leq p[i] \leq 100$

输出描述:

You should output an integer, which is the answer.

备注:

If you don't know how to output a fraction mod 1000000007,

You may have a look at https://en.wikipedia.org/wiki/Modular_multiplicative_inverse

示例 1

输入

3 4

50 50 50

输出

750000020

说明

000 0

001 1

010 1

011 16

100 1

101 2

110 16

111 81

The exact answer is $(0 + 1 + 1 + 16 + 1 + 2 + 16 + 81) / 8 = 59/4$.

As $750000020 * 4 \bmod 1000000007 = 59$

You should output 750000020.

标题：F、Typing practice | 时间限制：1 秒 | 内存限制：256M

Niuniu is practicing typing.

Given n ($1 \leq n \leq 4$) words, Niuniu want to input one of these.

Given an operation sequence, Niuniu want to know the answer after every operation.

An operation might input a character or delete the last character.

输入描述:

The first line contains one integer n .

In the following n lines, each line contains a word.

The last line contains the operation sequence.

The sequence only contains lower case letter.

'-' means backspace, and will delete the last character he typed.

输出描述:

You should output $l+1$ integers.

示例 1

输入

2

a

bab

baa-

输出

1

1

0

0

0

标题: G、Longest Common Subsequence | 时间限制: 1 秒 | 内存限制: 256M

Longest Common Subsequence

Given 4 sequences $\{a_i\}$, $\{b_i\}$, $\{c_i\}$, $\{d_i\}$.

Please find the longest common subsequence.

输入描述:

The first line contains one integer, which is n .

The second line contains n integers, which is the sequence $\{a_i\}$.

The third line contains n integers, which is the sequence $\{b_i\}$.

The fourth line contains n integers, which is the sequence $\{c_i\}$.

The fifth line contains n integers, which is the sequence $\{d_i\}$.

$1 \leq n \leq 10000$

$1 \leq a_i \leq n$

Any number appears in $\{a_i\}$ at most 2 times.

$1 \leq b_i \leq n$

Any number appears in $\{b_i\}$ at most 2 times.

$1 \leq c_i \leq n$

Any number appears in $\{c_i\}$ at most 2 times.

$1 \leq d_i \leq n$

输出描述:

You should output one integer, which is the length of the LCS.

示例 1

输入

5

1 2 1 2 3

1 2 3 1 2

3 1 2 1 2

1 2 1 2 1

输出

4

标题: H、Prefix Sum | 时间限制: 3 秒 | 内存限制: 256M

Niuniu has learned prefix sum and he found an interesting about prefix sum.

Let's consider $(k+1)$ arrays $a[i]$ ($0 \leq i \leq k$)

$a[i]$ is always the prefix sum of $a[i-1]$.

"always" means $a[i]$ will change when $a[i-1]$ changes.

"prefix sum" means $a[i][1] = a[i-1][1]$ and $a[i][j] = a[i][j-1] + a[i-1][j]$ ($j \geq 2$)

There are two kinds of operations, which are modify and query.

For a modify operation, two integers x, y are given, and it means $a[0][x] += y$.

For a query operation, one integer x is given, and it means querying $a[k][x]$.

As the result might be very large, you should output the result mod 1000000007.

输入描述:

The first line contains three integers, n, m, k .

n is the length of each array.

m is the number of operations.

k is the number of prefix sum.

In the following m lines, each line contains an operation.

If the first number is 0, then this is a change operation.

There will be two integers x, y after 0, which means $a[0][x] += y$;

If the first number is 1, then this is a query operation.

There will be one integer x after 1, which means querying $a[k][x]$.

$1 \leq n \leq 100000$

$1 \leq m \leq 100000$

$1 \leq k \leq 40$

输出描述:

For each query, you should output an integer, which is the result.

示例 1

输入

4 11 3

0 1 1

0 3 1

1 1

1 2

1 3

1 4

0 3 1

1 1

1 2

1 3

1 4

输出

1

3

7

13

1

3

8

16

说明

For the first 4 queries, the $(k+1)$ arrays are

1 0 1 0

1 1 2 2

1 2 4 6

1 3 7 13

For the last 4 queries, the $(k+1)$ arrays are

1 0 2 0

1 1 3 3

1 2 5 8

1 3 8 16

标题: I、Floorfiller | 时间限制: 1 秒 | 内存限制: 256M

Niuniu wants to fill an $n \times m$ sheet with 0s and 1s.

Niuniu wants the xor sum for each row and each column is 0.

In other words there is a even number of 1 in each row and each column.

Two sheets are considered the same, if they are identical after cyclic shift (vertical or horizontal).

Formally, for two sheets A and B, if we can find x and y such that

$$\forall i(0 \leq i < n), \forall j(0 \leq j < m), A_{i,j} = B_{(i+x) \bmod n, (j+y) \bmod m}$$

we will consider A and B are the same sheet.

As the result might be very large, he wants to know the result modulo 998244353.

输入描述:

The first line contains two integers, which are n and m.

$$1 \leq n \leq 10^9$$

$$1 \leq m \leq 10^9$$

输出描述:

You should output one integer, which is the answer modulo 998244353.

示例 1

输入

4 4

输出

48

示例 2

输入

4 6

输出

1448

标题: J、Maze | 时间限制: 1 秒 | 内存限制: 32M

Niuniu likes his maze (Mei Zi) and he wants to collect more mazes.

Given n, m , a maze is a $n \times m$ grid. The rows and columns are numbered from 0.

Let's call the cell on the i -th row and the j -th column as (i, j) .

There might vertical walls between (i, j) and $(i, j + 1)$.

There are vertical walls between (i, j) and $(i, j + 1)$.

输入描述:

For each test case, there is only one line, with two integers $n, m (1 \leq n, m \leq 10^9)$.

输出描述:

For each test case, output the answer in one line.

示例 1

输入

2 2

输出

164