牛客网暑期ACM多校训练营（第九场）

一. 编程题

1.
Niuniu has recently 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 x 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] xor a[j],
https://en.wikipedia.org/wiki/Bitwise_operation#XOR

Let p = 1000000007.
Consider the equation
A x = b (mod p)
where A is an n x n matrix, and x and b are both n x 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] (mod p)
A[1][0]*x[0] + A[1][1]*x[1] + A[1][2]*x[2] + A[1][3]*x[3] = b[1] (mod p)
A[2][0]*x[0] + A[2][1]*x[1] + A[2][2]*x[2] + A[2][3]*x[3] = b[2] (mod p)
A[3][0]*x[0] + A[3][1]*x[1] + A[3][2]*x[2] + A[3][3]*x[3] = b[3] (mod 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 <= n <= 262144

p = 1000000007

0 <= a[i] < p
0 <= 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 <= x[i] < p.
示例1:
输入
4
1 10 100 1000
1234 2143 3412 4321

输出
4
3
2
1

正确答案：

2.
Niuniu is (p)reviewing NOIP 2017 problems.
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).
He found that the official 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 weight 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 weight is z.

1 <= n <= 12
1 <= m <= 1000
1 <= x <= n
1 <= y <= n
1 <= z <= 5000
输出描述：
You should ouput 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

正确答案：

3.
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 (......)-th match, and team A wins or loses the entire series, the program ends.

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 <= n <= 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.

正确答案：

4.

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 <= j <= 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 <= k <= 7

2k+1 <= n <= 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

正确答案：

5.

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 <= n <= 1000
1 <= m <= 1000
0 <= p[i] <= 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 mod 1000000007 = 59
You should output 750000020.

正确答案：

6.
Niuniu is practicing typing.

Given n words, Niuniu want to input one of these. He wants to input as few characters as possible, to make at least one of the n words appears in the text.
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.

He may backspace when there is no characters left, and nothing will happen.

$1 <= n <= 4$
The total length of n words <= 100000
The length of the operation sequence <= 100000
输出描述：
You should output l+1 integers.
The i-th(index from 0) is the minimum characters to achieve the goal.
示例1:
输入
2
a
bab
baa-

输出
1
1
0
0
0

正确答案：

7.
Niuniu has recently learned LCS(Longest Common Subsequence) of two sequences.
Now he wants to know the LCS of four sequences.
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 $\{d_i\}$.

$1 <= n <= 10000$
$1 <= a_i <= n$
Any number appears in $\{a_i\}$ at most 2 times.
$1 <= b_i <= n$
Any number appears in $\{b_i\}$ at most 2 times.
$1 <= c_i <= n$
Any number appears in $\{c_i\}$ at most 2 times.
$1 <= d_i <= n$
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

正确答案：

8.
Niuniu has learned prefix sum and he found an interesting about prefix sum.

Let's consider (k+1) arrays a[i] (0 <= i <= k)
The index of a[i] starts from 1.
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 >= 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 <= n <= 100000
1 <= m <= 100000
1 <= k <= 40
1 <= x <= n
0 <= y < 1000000007
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

正确答案：

9.
Niuniu wants to fill an n x 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
$\forall i(0 \le i < n), \forall j(0 \le j < m), A_{i,j} = B_{(i+x) \bmod n,(j+y) \bmod m}$
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.

$1 <= n <= 10^9$
$1 <= m <= 10^9$
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
示例3:
输入
998244353 998244353

输出
295980207

正确答案：

10.
Niuniu likes his maze (Mei Zi) and he wants to collect more mazes.

Given n, m, a maze is a n x 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 may be vertical walls between (i, j) and (i, j + 1).
There may be vertical walls between (i, j) and (i, j + 1).

Two array v and h are given.
The size of v is n x (m - 1).
If v[i][j] is 1, then there is a wall between (i, j) and (i, j + 1).
If v[i][j] is 0, then there are two possibilities. There may or may not be a wall between (i, j) and (i, j + 1).

The size of h is (n-1) x m.
If h[i][j] is 1, then there is a wall between (i, j) and (i + 1, j).
If h[i][j] is 0, then there are two possibilities. There may or may not be a wall between (i, j) and (i + 1, j).

The beauty is defined as follows:
Calculate the size of each connected component, the sum of their square is the beauty.

You need output the sum of beauty of all possibilities.

(Obviously, the number of possibilities are 2^{the number of 0 in v and h})

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 integer n, m.
In following n lines, each line contains m-1 integers, which are the array v.
In following n-1 lines, each line contains m integers, which are the array h.

1 <= n <= 7
1 <= m <= 7
0 <= v[i][j] <= 1
0 <= h[i][j] <= 1
输出描述：
For each test case, output the answer in one line.
示例1:
输入
2 2
0
0
0 0

输出
164
说明
If there is 0 or 1 wall, the beauty is 4 * 4 = 16
If there are 2 walls and they form shape L, the beauty is 3 * 3 + 1 * 1 = 10
If there are 2 walls and they form shape I, the beauty is 2 * 2 + 2 * 2 = 8
If there are 3 walls, the beauty is 2 * 2 + 1 * 1 + 1 * 1 = 6
If there are 4 walls, the beauty is 1 * 1 + 1 * 1 + 1 * 1 + 1 * 1 = 4
The answer should be 16 * 5 + 10 * 4 + 8 * 2 + 6 * 4 + 4 * 1 = 164
示例2:
输入
2 2
0
1
0 1

输出
26
说明
The two existing wall forms shape L.
If there is no more wall, the beauty is 3 * 3 + 1 * 1 = 10
If there is 1 more wall, the beauty is 2 * 2 + 1 * 1 + 1 * 1 = 6
If there is 2 more walls, the beauty is 1 * 1 + 1 * 1 + 1 * 1 + 1 * 1 = 4
The answer should be 10 * 1 + 6 * 2 + 4 * 1 = 26

正确答案：