HAMMERWARS 2025

PROBLEM STATEMENTS

A1. Fire tower 3D (easy version)

Difficulty: Easy **Time**: 3 s **Memory**: 1024 MB

by shcal

Note: the only difference between the easy and hard versions of this problem is that in the easy version, $d \leq 100$ and $n, m, nm, k \leq 10^4$, while in the hard version, $d \leq 10^{12}$ and $n, m, nm, k \leq 10^6$.

authentic photograph of a real-life situation

It's the night before ICPC World Finals 2025 and the Purdue Waffle Wizards are playing their favorite board game: Fire Tower 3D. In retrospect, they would have solved more problems if they went to bed, but they can't rest until they know the outcome. Now that they've procured a time travel machine, all they need is for you to simulate the game!

Fire Tower 3D is played on an $n \times m$ grid (n rows, m columns). Each cell can contain fire, which can also grow upward. Each day, these steps happen in order:

1. Every cell containing fire *simultaneously* spawns a fire of height $1\mathrm{m}$ in each adjacent cell (sharing an edge and inside the grid) that does not already contain fire.
2. After that, the tower of fire in each cell grows upward by $1\mathrm{m}$.

Given the initial positions of k fires with height $1\mathrm{m}$, what's the total height in meters of all fires on the grid after d days?

Input

The first line contains 4 integers n, m, k, d ($1 \leq n, m, nm \leq 10^4$, $1 \leq k \leq nm$ and $1 \leq d \leq 100$).

The i-th of the next k lines contains two integers r_i, c_i with $1 \leq r_i \leq n$ and $1 \leq c_i \leq m$: the row and column of the i-th initial fire, respectively. All initial fires are distinct; no pair (r_i, c_i) is repeated.

Output

Output a single number: the total height in meters of all fires after d days.

Sample 1

Input

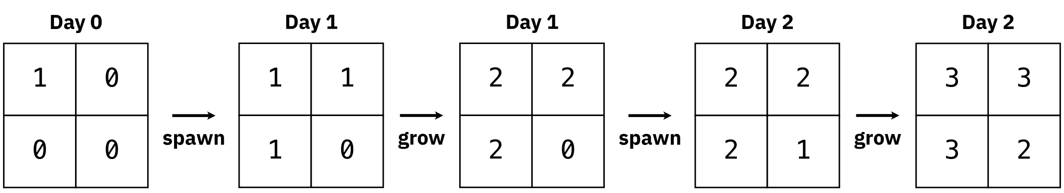
```
2 2 1 2
1 1
```

Output

```
11
```

Explanation

You can see the day-by-day process in the diagram on the next page. The number in each cell represents the height of the fire in that cell or 0 if the fire has not reached this cell yet.



After 2 days, the sum of the heights of fires is $3 + 3 + 3 + 2 = 11$, so the answer is 11.

Sample 2

Input

```
1 5 1 10
1 3
```

Output

```
53
```

Sample 3

Input

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

Output

```
83
```

A2. Fire tower 3D (hard version)

Difficulty: Easy **Time**: 3 s **Memory**: 1024 MB

by shcal

Note: the only difference between the easy and hard versions of this problem is that in the easy version, $d \leq 100$ and $n, m, nm, k \leq 10^4$, while in the hard version, $d \leq 10^{12}$ and $n, m, nm, k \leq 10^6$.

authentic photograph of a real-life situation

It's the night before ICPC World Finals 2025 and the Purdue Waffle Wizards are playing their favorite board game: Fire Tower 3D. In retrospect, they would have solved more problems if they went to bed, but they can't rest until they know the outcome. Now that they've procured a time travel machine, all they need is for you to simulate the game!

Fire Tower 3D is played on an $n \times m$ grid (n rows, m columns). Each cell can contain fire, which can also grow upward. Each day, these steps happen in order:

1. Every cell containing fire *simultaneously* spawns a fire of height $1\mathrm{m}$ in each adjacent cell (sharing an edge and inside the grid) that does not already contain fire.
2. After that, the tower of fire in each cell grows upward by $1\mathrm{m}$.

Given the initial positions of k fires with height $1\mathrm{m}$, what's the total height in meters of all fires on the grid after d days?

Input

The first line contains 4 integers n, m, k, d ($1 \leq n, m,\ nm \leq 10^6$, $1 \leq k \leq nm$ and $1 \leq d \leq 10^{12}$).

The i-th of the next k lines contains two integers r_i, c_i with $1 \leq r_i \leq n$ and $1 \leq c_i \leq m$: the row and column of the i-th initial fire, respectively. All initial fires are distinct; no pair (r_i, c_i) is repeated.

Output

Output a single number: the total height in meters of all fires after d days.

Sample 1

Input

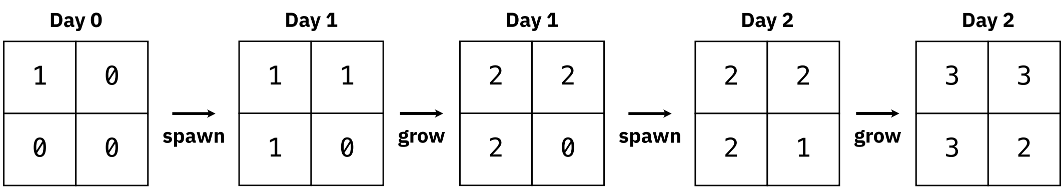
```
2 2 1 2
1 1
```

Output

```
11
```

Explanation

You can see the day-by-day process in the diagram on the next page. The number in each cell represents the height of the fire in that cell or 0 if the fire has not reached this cell yet.



After 2 days, the sum of the heights of fires is $3+3+3+2=11$, so the answer is 11.

Sample 2

Input

```
1 5 1 10
1 3
```

Output

```
53
```

Sample 3

Input

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

Output

```
83
```

B. Coloring book

Difficulty: Easy **Time**: 1 s **Memory**: 1024 MB

by munir_k

Thomas has generously provided n lines in a plane! Even better, his lines are in general position. That is, no two lines are parallel and no three lines intersect at a point.

The lines divide the plane into regions. Formally, points A and B are in the same region if and only if the segment $\overline{AB}$ doesn't intersect a line. Two regions are adjacent if the intersection of their boundaries has nonzero length. Note that this length may be infinite if both regions are unbounded.

Arvind and Zhongtang are going to play a game on the regions. First, Arvind will select a region and color it red. Then turns will proceed as follows: Zhongtang will select a region adjacent to a red region and color it blue, then Arvind will select a region adjacent to a blue region and color it red. The last person who can play wins.

You want prime betting odds. Can you predict the outcome of the game with perfect play?

Input

The first line contains an integer n ($2 \le n \le 2 \cdot 10^5$), the number of lines.

The i-th of the next n lines contains 4 integers a_i, b_i, c_i, d_i ($\max(|a_i|, |b_i|, |c_i|, |d_i|) \le 10^9$), denoting that the i-th line passes through lattice points (a_i, b_i) and (c_i, d_i) and extends forever in each direction.

Output

Output one line containing either "Arvind" or "Zhongtang" (without quotes).

Sample

Input

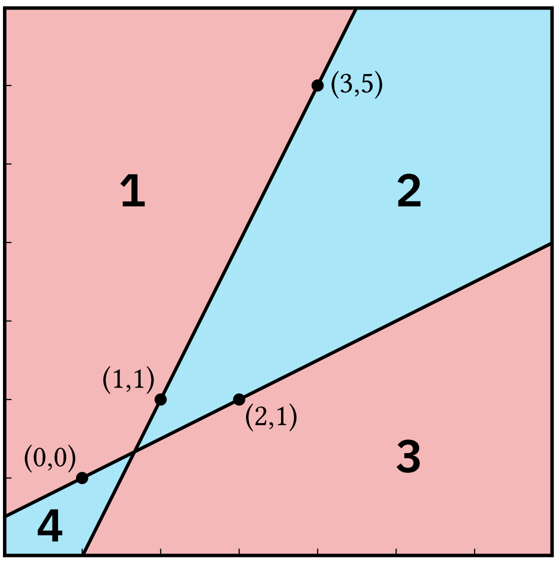
```
2
0 0 2 1
1 1 3 5
```

Output

```
Zhongtang
```

Explanation

Here's a possible game between Arvind and Zhongtang on the regions described in the sample:



Arvind selects region 1 and colors it red, then Zhongtang selects region 2, which is adjacent to region 1, colors it blue, and so on. The game ends when Zhongtang selects region 4 as all regions have been colored, so Zhongtang wins in this case.

Note that this image is cropped to fit on the page; the game is actually played on the infinite plane. (It does not make a difference here, since all regions are visible and their adjacency is the same.)

C1. Sorted subarrays (easy version)

Difficulty: Easy **Time**: 1.5 s **Memory**: 1024 MB

by shcal

Note: the only difference between the easy and hard versions of this problem is that in the easy version, $n \le 10^4$, while in the hard version, $n \le 10^5$. The hard version is easiest to complete in C++.

Given an array $a_1, ..., a_n$ of length n, find all $1 \le k \le n$ for which the sums of consecutive length k subarrays are nondecreasing.

The i-th $(1 \le i \le n - k + 1)$ consecutive length k subarray of $a_1, ..., a_n$ is $a_i, ..., a_{i+k-1}$, and its sum is $a_i + ... + a_{i+k-1}$.

Input

The first line contains an integer n $(1 \le n \le 10^4)$, the length of the array.

The second line contains n space-separated integers $a_1, ..., a_n$ $(0 \le a_i \le 10^9)$.

Output

Output a space-separated, **sorted** list of all $1 \le k \le n$ with nondecreasing subarray sums.

Sample 1

Input

```
5
1 9 6 9 8
```

Output

```
2 4 5
```

Explanation

The subarrays of length 2 are $[1, 9], [9, 6], [6, 9], [9, 8]$. Respectively, their sums are $1 + 9 = 10, 9 + 6 = 15, 6 + 9 = 15, 9 + 8 = 17$. Since $[10, 15, 15, 17]$ is nondecreasing, $k = 2$ has the property that the consecutive length k subarrays of a have nondecreasing sums, so it is in the output list.

Similarly, one can see that $k = 4$ and $k = 5$ are valid choices and that no others exist.

Sample 2

Input

```
20
21 13 34 1 21 7 36 24 44 31 19 46 42 45 35
21 46 16 43 45
```

Output

```
12 14 16 18 19 20
```

Explanation

The second line is wrapped here, but not in the actual input.

C2. Sorted subarrays (hard version)

Difficulty: Medium **Time**: 1.5 s **Memory**: 1024 MB

by shcal

Note: the only difference between the easy and hard versions of this problem is that in the easy version, $n \le 10^4$, while in the hard version, $n \le 10^5$. The hard version is easiest to complete in C++.

Given an array $a_1, ..., a_n$ of length n, find all $1 \le k \le n$ for which the sums of consecutive length k subarrays are nondecreasing.

The i-th $(1 \le i \le n - k + 1)$ consecutive length k subarray of $a_1, ..., a_n$ is $a_i, ..., a_{i+k-1}$, and its sum is $a_i + ... + a_{i+k-1}$.

Input

The first line contains an integer n $(1 \le n \le 10^5)$, the length of the array.

The second line contains n space-separated integers $a_1, ..., a_n$ $(0 \le a_i \le 10^9)$.

Output

Output a space-separated, **sorted** list of all $1 \le k \le n$ with nondecreasing subarray sums.

Sample 1

Input

```
5
1 9 6 9 8
```

Output

```
2 4 5
```

Explanation

The subarrays of length 2 are $[1, 9], [9, 6], [6, 9], [9, 8]$. Respectively, their sums are $1 + 9 = 10, 9 + 6 = 15, 6 + 9 = 15, 9 + 8 = 17$. Since $[10, 15, 15, 17]$ is nondecreasing, $k = 2$ has the property that the consecutive length k subarrays of a have nondecreasing sums, so it is in the output list.

Similarly, one can see that $k = 4$ and $k = 5$ are valid choices and that no others exist.

Sample 2

Input

```
20
21 13 34 1 21 7 36 24 44 31 19 46 42 45 35
21 46 16 43 45
```

Output

```
12 14 16 18 19 20
```

Explanation

The second line is wrapped here, but not in the actual input.

D1. Networking (easy version)

Difficulty: Easy **Time**: 3 s **Memory**: 1024 MB

by canin3

Note: The only difference between the easy and hard versions is that in the easy version, $n \leq 10^4$ while in the hard version, $n \leq 10^6$.

I don't know much about networking, but let's say Purdue's PAL network consists of n magical gizmos numbered $1, 2, ..., n$. Since IT is on a budget, their network topology is a weighted tree with $n - 1$ edges (u_i, v_i, w_i) for $1 \leq i \leq n - 1$. The i-th edge is between gizmos u_i and v_i, and takes w_i milliseconds to traverse.

After HammerWars registration crashed their network, they decided to add m extra bidirectional "shortcut" edges (u_i, v_i, w_i) for $n \leq i \leq n + m - 1$.

You have devices connected to the first k gizmos $1, 2, ..., k$ for some $2 \leq k \leq n$. Using all (tree and shortcut) edges, how long does it take to travel between every pair of your devices i, j for $1 \leq i < j \leq k$?

Input

The first line contains 3 integers n, m, k $(3 \leq n \leq 10^4, 1 \leq m, k \leq 100)$.

The i-th of the next $n - 1 + m$ lines contains integers u_i, v_i, w_i $(1 \leq u_i, v_i \leq n, 1 \leq w_i \leq 10^9)$, denoting that the i-th edge connects vertices u_i and v_i.

It is guaranteed that the first $n - 1$ edges form a tree. It is also guaranteed that there are no self-edges $(u_i \neq v_i)$ and all edges are between different pairs of vertices (each possible pair of u_i, v_i appears at most once in the edge list).

Output

Print k lines.

On the i-th line, output k space-separated integers $a_1, ..., a_k$, where a_j is the time it takes to travel from gizmo i to gizmo j (or 0 when $i = j$).

Sample

Input

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

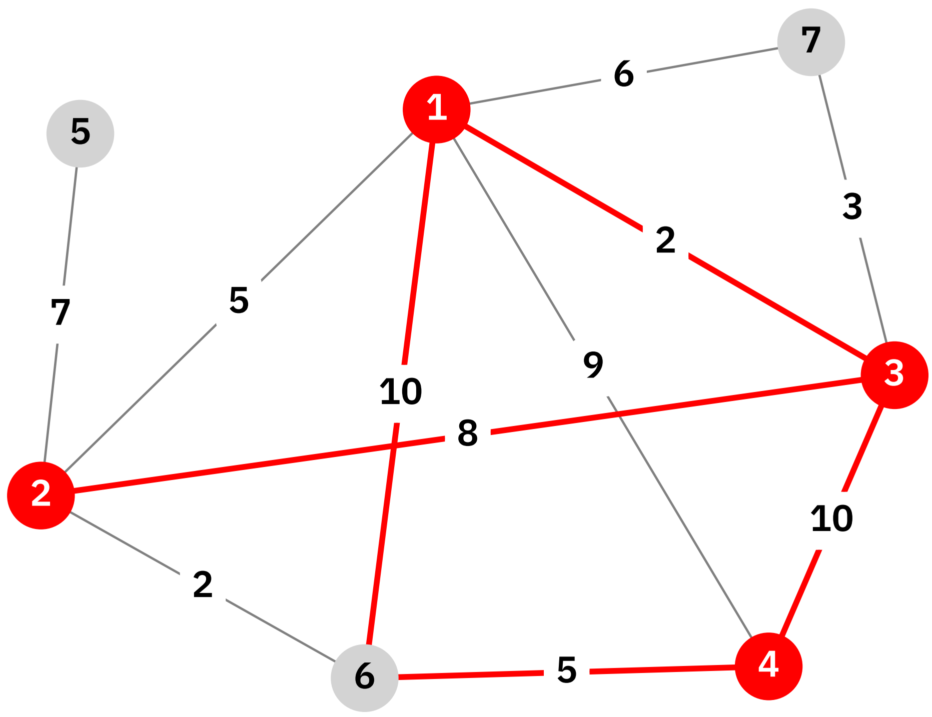
Output

```
0 5 2 9
5 0 7 7
2 7 0 10
9 7 10 0
```

Explanation

The next page contains a diagram of the graph for the sample. The first k gizmos and shortcut edges are highlighted in red.

For example, the 2nd element of the 3rd line is 7 since the shortest path between nodes 2 and 3 takes the shortcut edge with weight 2 from $3 \to 1$, then the tree edge with weight 5 from $1 \to 2$.


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

D2. Networking (hard version)

Difficulty: Medium **Time**: 3 s **Memory**: 1024 MB

by canin3

Note: The only difference between the easy and hard versions is that in the easy version, $n \le 10^4$ while in the hard version, $n \le 10^6$.

I don't know much about networking, but let's say Purdue's PAL network consists of n magical gizmos numbered $1, 2, ..., n$. Since IT is on a budget, their network topology is a weighted tree with $n-1$ edges (u_i, v_i, w_i) for $1 \le i \le n-1$. The i-th edge is between gizmos u_i and v_i, and takes w_i milliseconds to traverse.

After HammerWars registration crashed their network, they decided to add m extra bidirectional "shortcut" edges (u_i, v_i, w_i) for $n \le i \le n+m-1$.

You have devices connected to the first k gizmos $1, 2, ..., k$ for some $2 \le k \le n$. Using all (tree and shortcut) edges, how long does it take to travel between every pair of your devices i, j for $1 \le i < j \le k$?

Input

The first line contains 3 integers n, m, k $(3 \le n \le 10^6, 1 \le m, k \le 100)$.

The i-th of the next $n-1+m$ lines contains integers u_i, v_i, w_i $(1 \le u_i, v_i \le n, 1 \le w_i \le 10^9)$, denoting that the i-th edge connects vertices u_i and v_i.

It is guaranteed that the first $n-1$ edges form a tree. It is also guaranteed that there are no self-edges $(u_i \ne v_i)$ and all edges are between different pairs of vertices (each possible pair of u_i, v_i appears at most once in the edge list).

Output

Print k lines.

On the i-th line, output k space-separated integers $a_1, ..., a_k$, where a_j is the time it takes to travel from gizmo i to gizmo j (or 0 when $i = j$).

Sample

Input

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

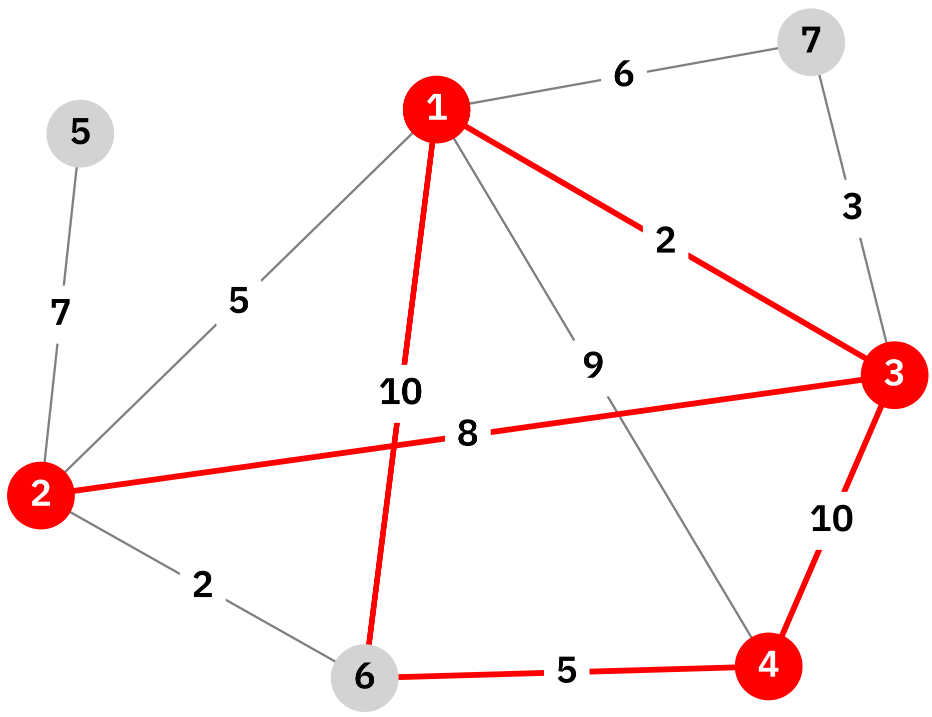
Output

```
0 5 2 9
5 0 7 7
2 7 0 10
9 7 10 0
```

Explanation

The next page contains a diagram of the graph for the sample. The first k gizmos and shortcut edges are highlighted in red.

For example, the 2nd element of the 3rd line is 7 since the shortest path between nodes 2 and 3 takes the shortcut edge with weight 2 from $3 \to 1$, then the tree edge with weight 5 from $1 \to 2$.


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

E. Research

Difficulty: Medium **Time**: 3 s **Memory**: 1024 MB

by zhtluo

A poor PhD student is submitting his paper to the ICLR (International Conference of Lottery Research). His paper will be reviewed by 3 different reviewers randomly picked out of n reviewers relevant to his field, and will be published only if all of them recommend acceptance.

To put his theory to the test, his reviewers will buy lottery tickets based on his results and either accept or reject his paper based on whether or not they make a profit. Since each reviewer either achieves financial freedom or goes completely broke after buying lottery tickets, they won't buy any more tickets after reviewing his paper once and will keep the same decision if they get picked again.

According to his paper, there is a p/q chance that any reviewer achieves financial freedom after buying lottery tickets based on his paper. If the student is willing to submit k times until his paper is published, what is the chance his paper is accepted?

Input

The first line contains 4 integers n, k, p, q ($3 \leq n \leq 10^6$, $1 \leq k, p, q \leq 10^9$, $p \leq q$).

Output

It can be shown that the answer can be represented as an irreducible fraction a/b for integers a, b ($b \neq 0$). Output the unique integer x ($0 \leq x < 10^9 + 7$) for which $bx = a \pmod{10^9 + 7}$.

Sample 1

Input

```
3 100 1 2
```

Output

```
125000001
```

Explanation

Here, there are 3 reviewers and you submit 100 times. On the first submission, all 3 reviewers are chosen. Since there is a half probability ($p = 1$ and $q = 2$, so $p/q = 1/2$) of each reviewer achieving financial freedom, the chance that all 3 of them achieve financial freedom and the paper gets accepted is $1/8$.

On the next 99 submissions (if needed), each of the 3 reviewers have already tested his theory, so this does not improve his chances. Hence, the answer is $1/8$, which is 125000001 mod $10^9 + 7$.

Sample 2

Input

```
10 10 9 10
```

Output

```
60520013
```

Explanation

After 10 submissions, the student gets his paper accepted

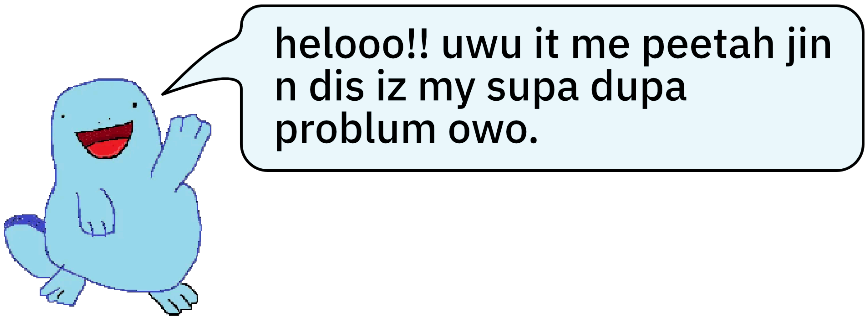
$$\frac{14088956145970285667291 71}{14155776000000000000000 00} \approx 99.5\%$$

of the time, which is 60520013 mod $10^9 + 7$.

F. Peter's polygon problem

Difficulty: Medium **Time**: 3 s **Memory**: 1024 MB

by shcal



givven array a wit n pozitive intejers, u must find some l, r $(1 \leq l, l + 2 \leq r \leq n)$ such dat $a_l, \ldots, a_r$ can b da side langths of some non-degenerat pollygon (dat meens polygon haz nonzero areaa owo!!). also u need 2 minimiz $r - l$. can u do it??!?!

Inpumbuxiageucxih

da furst line contans un intger n $(3 \leq n \leq 10^5)$.

da sekond line contans n pozitive intagers $a_1, \ldots, a_n$ $(0 < a_i \leq 10^{12}$, zero is bannd >:3).

o, i almos fogor! itz guranted dat der iz att leest 1 pear l, r satissfyeing teh afermontined reqwiremen!1

stanwar oput

prrint 1 line wit 2 intgers an spaec in midle l, r $(1 \leq l + 2 \leq r \leq n)$. if dere r multipl possibillitees of l, r dat minimiz $r - l$, spit out da one wit ze teeniest l owo.

swapmel on

Input

```
5
4 2 9 1 4
```

Output

```
1 5
```

Explanation

sowwy i 2 lazye to explien dis, butt metink it vewy tribial.

snnapnle tooh

Input

```
10
69 42 1 2 1 46 1 2 1 10
```

Output

```
1 6
```

G. Positivity

Difficulty: Hard **Time**: 1 s **Memory**: 1024 MB

by munir_k

Egor has an array $a_1, a_2, \dots, a_n$ of integers and a permutation $p_1, p_2, \dots, p_n$ with $i \neq p_i$ for all i.

In the face of any setback, Egor focuses on staying positive! Or, in this case, not negative. He wants to make $a_i + a_{p_i}$ nonnegative for all i after applying at most $\lfloor \frac{n}{2} \rfloor$ operations. In one operation, he will choose x and negate both a_x and a_{p_x}.

Since we all love Egor and he has retired from competitive programming, you must help Egor choose the operation sequence!

Input

The first line contains a single integer n $(2 \leq n \leq 2 \cdot 10^5)$.

The second line contains n integers $a_1, a_2, \dots, a_n$ $(|a_i| \leq 10^9)$.

The third line contains n integers $p_1, p_2, \dots, p_n$, a permutation of $1, \dots, n$ with $i \neq p_i$ for all i.

Output

On the first line, print the number of operations k $(0 \leq k \leq \lfloor \frac{n}{2} \rfloor)$.

On the second line, print the chosen indices $x_1, \dots, x_k$ separated by a space, where x_i is the index selected in the i-th operation.

If there are multiple sequences of at most $\lfloor \frac{n}{2} \rfloor$ operations which make $a_i + a_{p_i}$ nonnegative for all i, you may output any one of them.

Sample 1

Input

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

Output

```
2
1 3
```

Explanation

In this case, we negate $a_1, a_{p_1} = a_2, a_3$, and $a_{p_3} = a_4$, resulting in the array $-3, 4, -5, 6$. In this array:

- $a_1 + a_{p_1} = a_1 + a_2 = -3 + 4 = 1 \geq 0$
- $a_2 + a_{p_2} = a_2 + a_1 = 4 - 3 = 1 \geq 0$
- $a_3 + a_{p_3} = a_3 + a_4 = -5 + 6 = 1 \geq 0$
- $a_4 + a_{p_4} = a_4 + a_3 = 6 - 5 = 1 \geq 0$

so the requirement $a_i + a_{p_i} \geq 0$ for all i is satisfied.

Of course, this is only one solution and there are others which work for this case.

Sample 2

Input

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

Output

```
3
1 3 5
```

H. Equivalence classes

Difficulty: Hard **Time**: 2 s **Memory**: 1024 MB

by canin3

Wilbert is practicing for a medal at next year's ICPC World Finals. He knows ICPC problems are usually heavier on implementation and he is already incredible at math, so he should avoid pure math problems. Yet he is suddenly struck by a combinatorial curiosity...

He has a set of n elements $1, ..., n$ which are equivalent to each other at the beginning. Then, k operations occur. In each operation, a subset of the n elements is picked. Then the elements inside the set become different from the elements outside the set. Note that the selected set may be empty.

Since equivalence is bidirectional and transitive, it can be seen that after k operations, the elements can be uniquely partitioned into *equivalence classes*: maximal subsets in which each element is equivalent to every other, i.e. either both chosen or not chosen by each operation.

For each $1 \leq i \leq n$, how many sequences of operations are there that result in exactly i equivalence classes? Two sequences of operations are different if the subset chosen in the i-th operation is different for some i.

Input

The first line contains 2 integers n, k ($1 \leq n \leq 10^4$ and $1 \leq k \leq \min(10^{18}, 2^n)$), the number of elements and operations respectively.

Output

Output n lines.

On the i-th line, print one integer: the number of ways to end up with i equivalence classes, mod 998244353.

Sample 1

Input

```
2 2
```

Output

```
4
12
```

Explanation

When $i = 1$: for both sets, we may choose the empty set or all elements ($2 \cdot 2 = 4$ ways in total). At the end, there's only one equivalence class consisting of all elements.

Since there are $(2^n)^k = 2^4 = 16$ ways to perform all operations and there can only be 1 or 2 equivalence classes when $n = 2$, the number of ways to get $i = 2$ equivalence classes is $16 - 4 = 12$.

Sample 2

Input

```
5 5
```

Output

```
32
14880
744000
8630400
24165120
```

I. Complete and complement

Difficulty: Hard **Time**: 2 s **Memory**: 1024 MB

by bucketpotato

You are given an undirected **disconnected** graph G_1 with n vertices. You can perform the following two operations on G_1:

1. Pick distinct vertices x, y, z, such that the edges (x, y) and (y, z) are in the graph, but (x, z) is not in the graph. Then, add the edge (x, z) to the graph.
2. For all pairs of distinct vertices x, y: if the edge (x, y) is in the graph, then remove it from the graph. Otherwise, add it to the graph.

Now you are given a second graph G_2 with the same vertex set. Transform G_1 into G_2 using at most n^2 operations or say it's impossible.

Input

Input consists of multiple tests. The first line contains t $(1 \le t \le 250)$, the number of tests.

The first line of each test contains n $(2 \le n \le 10^3)$, the number of vertices in G_1 and G_2 .

The next n lines each contain strings of length n, consisting of "0"s and "1"s, which describe G_1. The j-th character of the i-th string (denoted $s_{i,j}$) is "1" if there is an edge between vertex i and vertex j, and is "0" otherwise.

The next n lines each contain strings of length n, describing G_2 in the same format as G_1.

It is guaranteed that for both graphs, and for all $1 \le i, j \le n$, that $s_{i,j} = s_{j,i}$ and $s_{i,i} = 0$.

It is guaranteed that G_1 is disconnected. It is **NOT** guaranteed that G_2 is disconnected.

It is guaranteed that the sum of n across all tests does not exceed 10^3.

Output

For each test, if the goal is impossible, output -1.

Otherwise, on the first line, output k $(0 \le k \le n^2)$, the number of operations you perform.

On the following k lines: first output an integer $1 \le \text{type} \le 2$ denoting the type of move you wish to perform.

If $\text{type} = 1$, then you should also output 3 distinct integers $1 \le x, y, z \le n$ such that edges (x, y) and (y, z) exist in the graph, but (x, z) does not. **Note that the order in which you output these numbers matters.**

If there are multiple possible answers using at most n^2 moves, you can output any. You do not need to minimize the number of moves. It can be shown that if there exists a way to transform G_1 into G_2, there is a way using at most n^2 moves.

Sample 1

Input

```
2
2
00
00
01
10
2
00
00
00
00
```

Output

```
1
2

2
2
2
```

Explanation

We start with a graph with 2 vertices and no edges.

- In the first test, G_2 is the complement of G_1, so we just need to apply operation 2 once.
- In the second test, G_2 is already equal to G_1, so we don't need to apply any operations. But we don't need to minimize the number of operations, so it's fine if we perform operation 2 twice.

Sample 2

Input

```
2
4
0100
1000
0001
0010
0111
1011
1101
1110
4
0100
1000
0001
0010
0100
1010
0101
0010
```

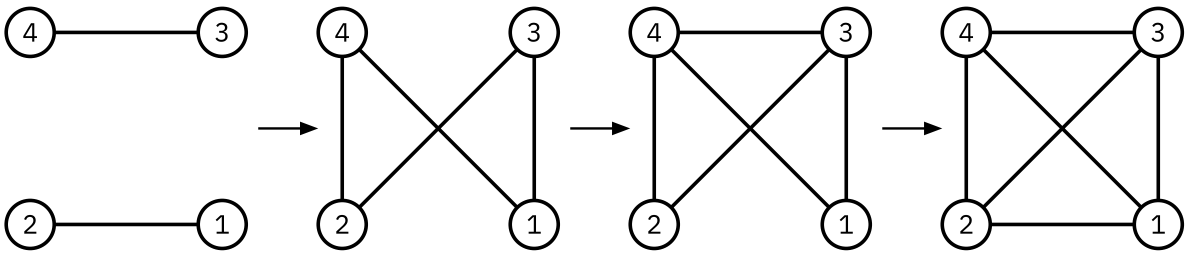
Output

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

-1
```

Explanation

In the first test, the sequence of operations performed looks like this:



In the second test, we can show there is no answer.

J. Immense integers

Difficulty: Demon **Time**: 2 s **Memory**: 1024 MB

by shcal and canin3

You're given two real numbers $0 < x, y < 1$. In one operation, you can halve the distance between x and 1 or between x and 0. Formally, for each operation, you may do one of the following:

- $x := (1 + x)/2$
- $x := x/2$

You are also given a tolerance $0 < t < 1$. What is the minimum number of operations (possibly none) needed so that $|x - y| \leq t$?

Input

The input consists of 3 binary strings x, y, t separated by a newline.

All numbers are fractions strictly between 0 and 1, specified in base-2 with at most 10^6 digits. That is, each number is given as a binary string $s_1 ... s_n$ $(s_i = 0$ or $1)$ for some $n \geq 1$ and corresponds to $\sum_{i=1}^{n} s_i 2^{-i}$.

Output

Output a single integer: the number of operations required to satisfy the given tolerance, which may be 0.

Sample 1

Input

```
01
00101
00010
```

Output

```
1
```

Explanation

Here, $x = 0 \cdot 2^{-1} + 1 \cdot 2^{-2} = 1/4, y = 5/32$, and $t = 1/16$.

Before any operations, $|x - y| = |1/4 - 5/32| = 3/32 > 1/16 = t$, so the constraint is not already satisfied and we must perform at least one operation.

After one operation with $x := x/2$, $x = 1/8$ and $|x - y| = |1/8 - 5/32| = 1/32 \leq 1/16 = t$. Hence the answer is 1.

Sample 2

Input

```
0001101110
0011111110
00000101
```

Output

```
3
```

Sample 3

Input

```
0100010110110001011011101
0101101011101101101010011
0000000000000000000000011
```

Output

```
17
```

K. Sadism

Difficulty: Demon **Time**: 2 s **Memory**: 1024 MB

by canin3

Oh dear! Thomas accidentally overcomplicated the solution to a Div 2 E. Luckily, he solved a more general case and can now force you to endure his suffering!

You're given an array $a_1, ..., a_n$ of length n. The MEX of the subarray $a_i, ..., a_j$ is defined as the smallest nonnegative integer $x \geq 0$ which does not appear in the subarray (i.e. no $i \leq l \leq j$ satisfies $a_l = x$). For each possible MEX $0 \leq x \leq n$, calculate the number of nonempty subarrays $a_i, ..., a_j$ $(1 \leq i \leq j \leq n)$ with that MEX.

Input

The first line contains an integer n $(1 \leq n \leq 10^6)$, the length of the array.

The second line contains n integers $a_1, ..., a_n$ $(0 \leq a_i < n)$.

Output

Output $n + 1$ integers separated by a space on one line, where the i-th integer is the number of nonempty subarrays with MEX $i - 1$ (i.e. starting at MEX 0).

Sample 1

Input

```
4
1 0 0 2
```

Output

```
2 5 2 1 0
```

Explanation

For example, the subarray $a_1, ..., a_4$ is the only subarray with MEX 3, since it contains $0, 1, 2$, but not 3. Therefore the 4th element of the output sequence is 1.

Sample 2

Input

```
7
1 5 4 2 6 0 3
```

Output

```
16 10 0 1 0 0 0 1
```

L. Birthday bash

Difficulty: Demon **Time**: 5 s **Memory**: 1024 MB

by robertyl

Today is Alice's birthday! She invited you and her $n - 2$ other friends to her birthday bash.

The birthday cake is a convex polygon with n sides. After they sing Happy Birthday, the cake is cut as follows: a point P is chosen uniformly at random in the interior of the cake and n cuts are made from P to the corners of the cake, creating n triangular slices. Since you and Alice are nice, you let your $n - 2$ friends first take a slice, then you take a slice, and Alice gets the last one.

If everyone greedily takes cake slices by area, what is the expected area of your slice?

Input

The first line contains a single integer n $(3 \leq n \leq 200)$, the number of sides.

The i-th of the following n lines contains two integers x_i and y_i $(-10^3 \leq x_i, y_i \leq 10^3)$, the coordinates of the i-th vertex of the polygon.

The vertices are given in counterclockwise order. The polygon is convex: all internal angles of the polygon are strictly smaller than π.

Output

Print one real number: the expected area of your slice of cake.

Your answer will be considered correct if its absolute or relative error does not exceed 10^{-6}.

Sample 1

Input

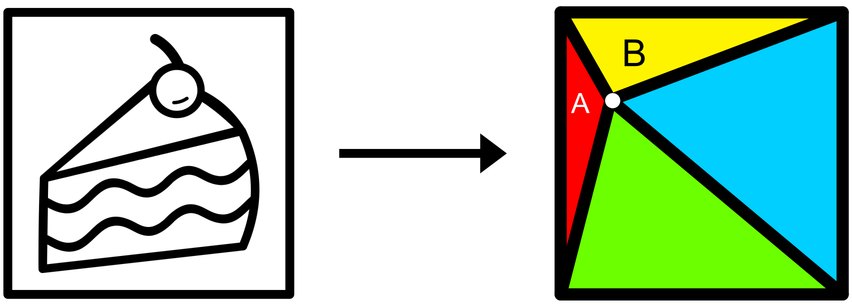
```
4
0 0
1 0
1 1
0 1
```

Output

```
0.166666637
```

Explanation

In the sample, we have a 1×1 square. For instance, if the white dot below is chosen as the point, Alice gets the smallest slice A and you receive B. It can be shown that over all points inside the square, the average area of the second-smallest slice is $1/6$.



There is another sample on the next page.

Sample 2

Input

```
10
-43 33
-35 -39
32 -41
46 -12
50 30
50 34
49 38
42 49
37 49
-6 44
```

Output

```
95.718821997
```