Problem A
ATCG

Time limit: 1 second

Memory limit: 2048 megabytes

Problem Description

DNA is a polymer found in many organisms' cells, including humans. It composes of two strands, and the two separate strands coil together to form a double helix.

The structure of each DNA strand can be seen as an array of nucleotides. According to the type of nucleobases each nucleotide contains, we can classify the nucleotides into four categories: `A`, `T`, `C`, and `G`. Thus, each strand of the DNA can be simplified as a sequence containing characters `ATCG`. The beginning and the end of any DNA strand are called the $5'$-end and the $3'$-end, respectively.

The nucleotides on the two strands are bound to each other, which is why the two strands can coil together. According to base pairing rules, `A` is always paired with `T` while `C` is always paired with `G`. The direction of the nucleotides in one strand is opposite to the direction in the other strand; that is, one strand's $5'$-end matches the other strand's $3'$-end. Hence, to find the sequence for the other strand, one just needs to reverse the sequence and do the mapping (`A` $\to$ `T`, `T` $\to$ `A`, `C` $\to$ `G`, `G` $\to$ `C`). See the figure below for a better understanding.

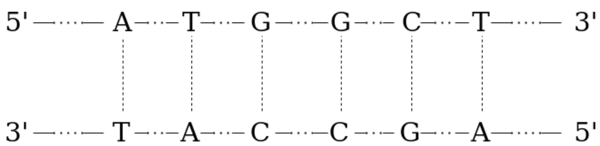

Figure 1: The DNA structure.

Alan is a biologist conducting experiments with DNA. He is playing with a special kind of DNA, just finding the sequence for one strand from the $5'$-end to the $3'$-end is s. Can you help him find the sequence for the other strand from the $5'$-end to the $3'$-end?

Input Format

The first line contains the number of test cases T.

Each test case contains two lines. The first line contains an integer n denoting the length of s. The second line contains the string s.

Output Format

For each testcase, print the sequence for the other strand from the $5'$-end to the $3'$-end in one line.

Technical Specification

- $1 \leq T \leq 2000$
- $1 \leq n \leq 100$
- $n = |s|$
- Each character in s is one of `ATCG`.

Sample Input 1

```
3
4
ATCG
4
ACGT
6
ATGGCT
```

Sample Output 1

```
CGAT
ACGT
AGCCAT
```

Problem B
Buying Mascots

Time limit: 8 seconds

Memory limit: 2048 megabytes

Problem Description

Brian is a fan of the double-sided octopus mascots. He visits a fair with n stands providing numerous octopus double-sided mascots to collect as many mascots as possible.

The stands in the fair are numbered from 1 to n, and Brian will also visit the stands in this order. When visiting the i-th stand, he can do exactly one of the following:

1. Pay a_i dollars to obtain a_i tokens. Brian can hold at most m tokens at a time. If Brian earns more than m at any time, the rest of the tokens must be returned to the stand.

2. Pay b_i tokens to obtain b_i octopus double-sided mascots.

At this moment, Brian has no tokens. What is the maximum number of octopus double-sided mascots he can receive after visiting the n stands? You may assume that Brian always has enough money to pay for the tokens.

Input Format

The first line of the input contains two integers n, m. The second line of the input contains n integers $a_1, a_2, \ldots, a_n$. The third line of the input contains n integers $b_1, b_2, \ldots, b_n$.

Output Format

Print the the maximum number of double-sided octopus mascots Brian can receive.

Technical Specification

- $1 \leq n \leq 10^5$
- $1 \leq m \leq 100$
- $0 \leq a_i, b_i \leq m$ for $i = 1, 2, \ldots, n$

Sample Input 1

```
5 10
5 6 0 10 2
2 3 10 0 3
```

Sample Output 1

```
13
```

Sample Input 2

```
7 100
21 15 0 32 21 23 14
8 23 20 67 31 72 15
```

Sample Output 2

```
87
```

Problem C
Colorful Pictures

Time limit: 3 seconds

Memory limit: 2048 megabytes

Problem Description

Carol is a prestigious painter in the PCCA Kingdom. Unfortunately, she has to spend time in quarantine during the pandemic, so she comes up with the game described below.

At the beginning of the game, she paints a tree with n nodes numbered $1, 2, \ldots, n$. There are $n-1$ edges, i-th of which is (u_i, v_i). She also draws a picture with n squares numbered $1, 2, \ldots, n$. The i-th square is colored with color $a_i = i$.

Then she can perform the following operation any number of times (possibly zero): Select an existing edge (u_i, v_i) from the tree, swap the colors in the cell u_i and v_i (that is, swap a_{u_i} and a_{v_i}) and remove the edge (u_i, v_i) from the tree.

How many different pictures can be painted after performing any number of operations? Two pictures are considered different if a square i is painted in different colors in the two pictures.

Input Format

The first line of the input contains an integer n. The i-th of the following $n-1$ lines contain two integers u_i, v_i.

Output Format

Print the number of different rectangles can be painted after performing any number of the operations. Output the result modulo 998244353.

Technical Specification

- $1 \le n \le 1.5 \times 10^5$
- $1 \le u_i, v_i \le n$
- It is guaranteed that the $n-1$ edges form a tree.

Sample Input 1

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

Sample Output 1

```
16
```

Sample Input 2

```
2
1 2
```

Sample Output 2

```
2
```

Sample Input 3

```
23
3 13
9 14
8 20
11 4
8 22
6 10
6 2
18 8
19 8
8 9
4 8
4 5
3 8
23 8
15 12
1 14
16 8
2 12
7 6
8 21
8 6
17 8
```

Sample Output 3

```
616298673
```

Almost blank page

Problem D
Duo of Magicians

Time limit: 1 second

Memory limit: 2048 megabytes

Problem Description

Neko-chan and Laffey are a famous duo of magicians in the PCCA kingdom. Recently they came up with the following new trick for their next show.

In the beginning, Neko-chan will invite an audience to write down a random permutation $a = [a_1, a_2, \ldots, a_n]$ of $1, 2, \ldots, n$ without showing it to Laffey. After seeing the permutation, Neko-chan will hand out a deck of cards (at most n cards, otherwise, it would be too apparent that they are cheating) to another audience. Neko-chan will ask the audience to shuffle the cards in any order and hand it to Laffey. Here comes the miracle: Laffey then can read out a sequence of operations to swap elements in a, and the array a would be sorted in increasing order!

If you wonder how this trick works, here is the secret. After the array a is written, Neko-chan secretly writes down a pair of integers (x_i, y_i) on each card. The pair (x_i, y_i) denotes that Laffey have to swap a_{x_i} and a_{y_i}. When Laffey receives the deck of cards, she only has to sort the pairs on each card lexicographically and read out the sorted pairs. In other words, Neko-chan gives Laffey an array of integer pairs *op* (which is also a sequence of operations); all Laffey has to do is sort *op*.

To make the trick work, Neko-chan requires an efficient algorithm to help him produce the array of pairs *op*. This task is too complex for Neko-chan. Can you help him?

Input Format

The first line of the input contains an integer T, denoting the number of test cases.

Each testcase consists of two lines. The first line contains an integer n. The second line contains n integers $a_1, a_2, \ldots, a_n$.

Output Format

For each testcase, print an integer k denoting the number of cards given to Laffey (which is also the length of *op*).

k lines follow. In the i-th line, print two integers x_i, y_i denoting the i-th pair given to Laffey.

Your solution will be considered correct if it satisfies all the following conditions:

- $0 \le k \le n$
- $1 \le x_i < y_i \le n$ for $i = 1, 2, \ldots, k$
- When sorted in lexicographical order, the sequence of operations will sort a in increasing order.

If there are multiple possible solutions, print any.

Technical Specification

- $1 \le T \le 200$
- $2 \le n \le 10^5$
- $\sum n \le 10^6$
- $1 \le a_i \le n$ for $i = 1, 2, \ldots, n$
- $a_i \neq a_j$ for $i \neq j$

Sample Input 1

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

Sample Output 1

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

Note

To sort the pairs lexicographically means to sort the pairs in the increasing order of the first element and the increasing order of the second element in case of ties.

Almost blank page

Problem E
Exterior

Time limit: 3 seconds

Memory limit: 2048 megabytes

Problem Description

Shokuhou (食蜂) is a girl who lives in Academy City. The Academy City comprises n districts numbered $1, 2, \ldots, n$. There are m bidirectional roads between the districts. The i-th road connects city u_i and v_i, and traversing it takes exactly c_i minutes. In addition, each district has a portal; the portal in the i-th city has power a_i. With these portals, one can travel from the i-th city to the j-th city in $a_i + a_j$ minutes for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, n$.

Shokuhou is currently at district 1, and her home is at district n. Since she learned she had just had a break-in at her house, she would like to return home immediately. She can go back home using any number of roads and portals. What is the minimum number of minutes she must spend to return home?

Input Format

The first line of the input contains two integers n, m. The second line contains n integers $a_1, a_2, \ldots, a_n$.

m lines follow. The i-th line contains three integers u_i, v_i, c_i denoting the i-th road.

Output Format

Print the minimum number of minutes she must spend to return home.

Technical Specification

- $2 \le n \le 10^5$
- $0 \le m \le \min(10^5, \frac{n(n-1)}{2})$
- $1 \le a_i \le 10^5$ for $i = 1, 2, \ldots, n$
- $1 \le u_i < v_i \le n$ for $i = 1, 2, \ldots, m$
- Each pair of (u_i, v_i) appears in the input at most once.
- $1 \le c_i \le 10^5$ for $i = 1, 2, \ldots, m$

Sample Input 1

```
4 4
6 2 1 5
1 2 1
2 3 6
1 3 8
3 4 2
```

Sample Output 1

```
6
```

Sample Input 2

```
5 4
12 23 34 45 56
1 2 1
2 3 1
3 4 1
4 5 1
```

Sample Output 2

```
4
```

Sample Input 3

```
6 7
8 5 1 7 8 6
3 2 1
2 5 1
3 5 4
2 6 3
4 5 17
3 4 10
5 6 2
```

Sample Output 3

```
13
```

Almost blank page

Problem F
Formidable Team

Time limit: 3 seconds

Memory limit: 2048 megabytes

Problem Description

As a PCCA winter camp staff member, you have been tasked with creating teams of m members from a list of n participants. Each participant has m skills; the i-th participant has skill level $a_{i,j}$ for the j-th skill for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, m$.

To facilitate team selection, you define the metrics for the skill levels. For a team b consisting of m members $b_1, b_2, \ldots, b_m$, define the **strength** of S as $\max\limits_{p \in P_m} \sum\limits_{i=1}^{m} a_{b_i,p_i}$, where P_m is the set of all possible permutations of $1, 2, \ldots, m$. That is, the strength of a team is the maximum possible sum of skill levels, considering all possible combinations of the skills from the m team members.

Your goal is to find the best team by maximizing the team's strength.

Input Format

The first line of the input contains two integers n, m. m lines follow, i-th of which contains m integers $a_{i,1}, a_{i,2}, \ldots, a_{i,m}$.

Output Format

Print the maximum possible strength k in the first line. In the i-th line of the following m lines, print two integers s_i, t_i denoting that the team contains s_i-th member with t_i-th skill.

Your solution will be considered correct if it satisfies all the following conditions:

- All t_i are distinct.
- $\sum\limits_{i=1}^{m} a_{s_i,t_i} = k$

If there are multiple possible solutions, print any.

Technical Specification

- $1 \leq n \leq 150000$
- $1 \leq m \leq 60$

- $m \leq n$
- $n \times m \leq 2 \times 10^6$
- $1 \leq a_{i,j} \leq 10^9$

Sample Input 1

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

Sample Output 1

```
19
1 5
2 3
3 4
4 1
5 2
```

Problem G
Genetic Sequence Searching

Time limit: 5 seconds

Memory limit: 2048 megabytes

Problem Description

Recently scientists have been looking into a new kind of pathogenic virus. To learn more about this virus, they did DNA sequencing on the samples and identified a typical pattern t in all the virus samples.

The scientists then want to identify the pattern in other DNA sequences. They just collected a DNA sequence s from cells of other creatures and were eager to discover all the occurrences of t in s. However, since DNA sequencing can go wrong, they want to find out the occurrences that differ in at most 1 character.

More formally, Let $s = s_1 s_2 \ldots s_{|s|}$ and $t = t_1 t_2 \ldots t_{|t|}$ are two DNA sequences. The scientists want to know all integer i between 1 and $|t| - |s| + 1$, such that substrings $s_i s_{i+1} \ldots s_{i+|t|-1}$ and $t_1 t_2 \ldots t_{|t|}$ have at most 1 different character.

Input Format

The first line of the input contains the string s. The second line of the input contains the string t.

Output Format

Print the number of occurrences in the first line. If there is at least one occurrence in t, then in the second line, print all i such that substrings $s_i s_{i+1} \ldots s_{i+|t|-1}$ and $t_1 t_2 \ldots t_{|t|}$ have at most 1 different character. Print the answers in increasing order.

Technical Specification

- $1 \leq |t| \leq |s| \leq 10^6$
- Each character in s and t has an ASCII code in the rage $[33, 126]$. In other words, each character is one of `` !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[ ]^_`abcdefghijklmnopqrstuvwxyz{|}~ ``.

Sample Input 1

```
PCCA_Winter_Camp_2023
AC
```

Sample Output 1

```
4
1 2 4 12
```

Sample Input 2

```
meowmeow
owo
```

Sample Output 2

```
1
3
```

Problem H
Heritage in the PCCA Kingdom

Time limit: 20 seconds
Memory limit: 2048 megabytes

Problem Description

The PCCA kingdom has a world heritage site that everybody knows: The Triangular Cave. Legend says there was an ancient civilization of triangles in the cave, and they left a powerful relic somewhere nobody knows. One day, you find a mysterious triangular stone plate, and it turns out to be the legendary relic of the Holy Triangle!

The relic is divided into n layers of small triangles, and the imprints enclose the small triangles. You notice that some imprints have been charged with ancient power while others have not. To fully empower the relic, you must restore it by making all the imprints charged.

However, you cannot charge the imprints directly. Instead, you can replace the small triangle with the energy runestones you prepared beforehand. An energy runestone is also in the shape of a triangle and can charge the imprints on the three sides of it once it is put onto the relic.

The figures below show examples of a relic before and after restoration:

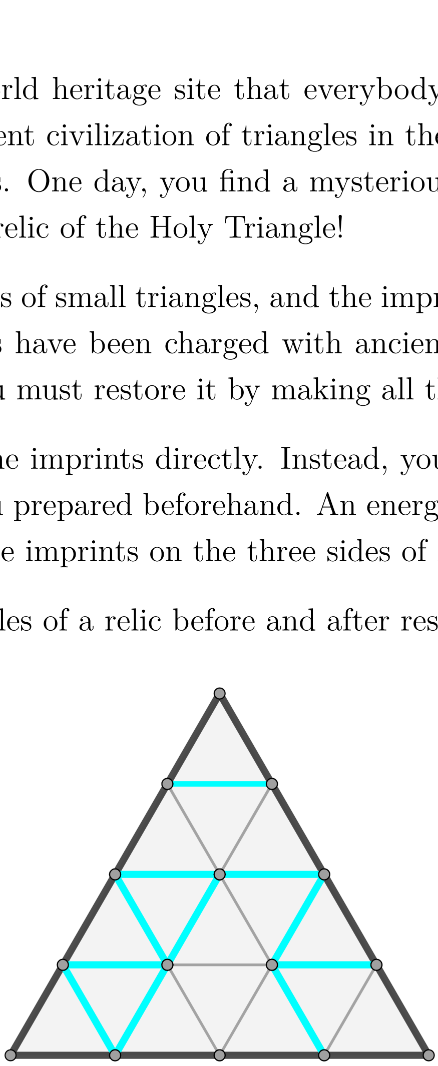

Figure 2: A partially charged relic before restoration. The blue edges are charged imprints and the grey edges are imprints not charged yet.

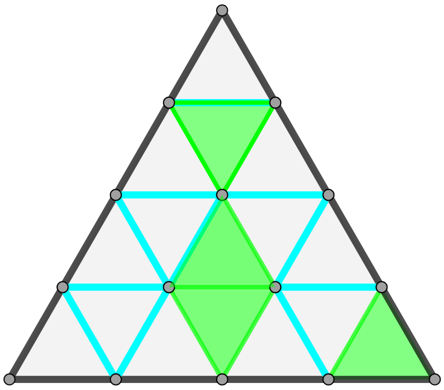

Figure 3: A fully charged relic after restoration. After putting 4 runestones onto the relic, all of the imprints are charged.

Here comes the problem: What is the minimum number of energy runestones required to empower the relic fully?

Input Format

The first line contains an integer n.

The following $2n$ lines indicate the status of the imprints. For each $i = 1, 2, \cdots, n$, two lines follow, indicating the status of the imprints of the i-th layer. The format for the i-th layer is the following:

- Both of the two lines begin with $n - i$ spaces.
- Exactly $2i$ characters follow the spaces in the first line. Each character represents an imprint. `\` or `/` denotes a charged imprint, while `.` denotes an uncharged imprint.
- Exactly $2i$ characters follow the spaces in the second line. A consecutive pair of characters represents an imprint. `..` denotes an uncharged imprint, while `--` denotes a charged imprint.

Output Format

Print the minimum number of runestones required.

Technical Specification

- $1 \leq n \leq 500$

Sample Input 1

```
4
   /\
   --
  /..\
  ----
 /\/./\
 --..--
/\/..\.\
--------
```

Sample Output 1

```
4
```

Sample Input 2

```
4
   /\
   ..
  /..\
  ....
 /....\
 ......
/......\
--------
```

Sample Output 2

```
6
```

Sample Input 3

```
6
     /\
     ..
    /./\
    ..--
   /....\
   --....
  /./.../\
  ----..--
 /\/\..../\
 --....--..
/..././\../\
------------
```

Sample Output 3

```
14
```

Note

The first sample corresponds to the figure in the statement.

Problem I
Interval Cover

Time limit: 1 second

Memory limit: 2048 megabytes

Problem Description

Ian has a multiset S of intervals on the number axis, i-th of which is $[l_i, r_i]$. Surprisingly, he found that every interval is within the $[0, l]$ range, where l is Ian's favorite positive integer.

As Ian thinks everything should be perfectly balanced, he wants to add the minimum number of intervals to S, so that the same number of intervals covers every non-integer coordinate between 0 and l. The newly added intervals should also be within the $[0, l]$ range. Let $f(S)$ be the minimum number of intervals to be added. Note that S is not changed after calculating $f(S)$.

For example, consider the case that S contains intervals $[0, 3], [2, 8], [7, 10]$ and $l = 10$. Then Ian can add 3 intervals $[0, 2], [3, 7], [8, 10]$ to S, and then every non-integer coordinate between 0 and 10 are covered by 2 intervals. Therefore $f(S) = 3$ in this case.

Due to the instability of S, Ian observes that S is prone to change. He wonders if the value of $f(S)$ can be changed. More formally, there are q queries of three types:

1. 1 ql_i qr_i —Adding an interval $[ql_i, qr_i]$ to S.
2. 2 ql_i qr_i —Removing an interval $[ql_i, qr_i]$ to S. It is guaranteed that $[ql_i, qr_i]$ appears in S.
3. 3 —Ian wants to know the value of $f(S)$.

Can you help Ian answer all the queries?

Input Format

The first line of the input contains two integers n, l. The i-th of the following n lines contains two integers l_i, r_i.

The following line contains an integer q. The i-th of the following q lines contains the i-th query in the format as in the problem description.

Output Format

For each query of type 3, print the value of $f(S)$ in one line.

Technical Specification

- $1 \leq n, l \leq 2 \times 10^5$
- $0 \leq l_i < r_i \leq l$ for $i = 1, 2, \ldots, n$
- $1 \leq q \leq 2 \times 10^5$
- $0 \leq ql_i < qr_i \leq l$ for $i = 1, 2, \ldots, q$ of query type $1, 2$
- It is guaranteed that there is at least one query of type 3 in the input.

Sample Input 1

```
5 10
0 3
3 4
4 10
0 7
7 10
7
3
1 1 6
3
1 0 1
3
2 4 10
3
```

Sample Output 1

```
0
2
1
2
```

Problem J
Jewelry Box

Time limit: 2 seconds

Memory limit: 2048 megabytes

Problem Description

Jen has a jewelry box with n rows and m columns. The rows are numbered from 1 to n from top to bottom, and the columns are numbered from 1 to m from left to right. Each cell is identified by a pair (x, y), which means it is located in the x-th row and the y-th column.

There are some jewels located in some cells of the box. Each cell contains, at most, one jewel. Surprisingly, no two jewels are located in adjacent cells. In this problem, two cells are considered adjacent if they share an edge.

For some reason, Jen would like to form some pairs of jewels. Each pair of jewels must be connected by a strand of rope, which may occupy some other cells of the jewelry box. More formally, suppose a strand of rope connects jewels located at (a_1, b_1) and (a_k, b_k), and then the rope can be seen as a sequence of k cells $(a_1, b_1), (a_2, b_2), \ldots, (a_k, b_k)$. Each pair of consecutive cells must be adjacent, and all the k cells should be distinct. Due to aesthetics, no jewels should be attached to more than one rope, and no two ropes should occupy the same cell.

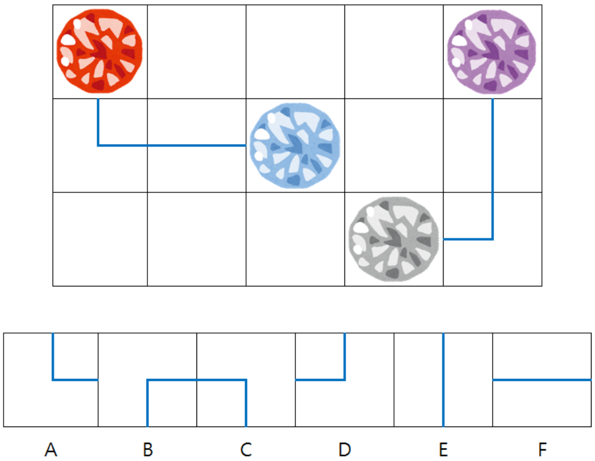

Figure 4: A valid configuration of jewels and ropes. There are 2 pairs of jewels in the figure.

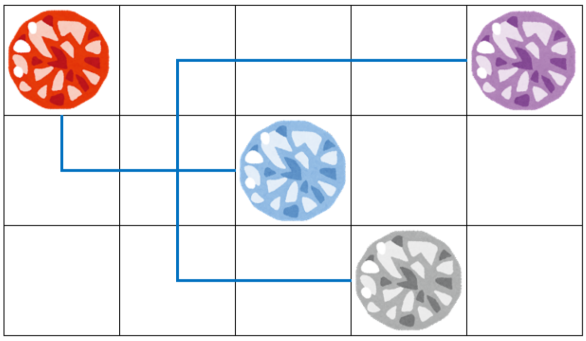

Figure 5: An invalid configuration of jewels and ropes because the two ropes share a common cell.

Jen hopes to form as many pairs of jewels as possible. She also wonders about a way to place these strands of ropes. Can you help her out?

Input Format

The first line of the input contains an integer T, denoting the number of test cases.

The first line of each test case consists of two integers n, m. Each of the following n lines contains m characters. For each $i = 1, 2, \ldots, n$, the j-th character denotes whether there is a jewel in cell (i, j) for $j = 1, 2, \ldots, m$. # denotes a cell containing a jewel, while . denotes a cell not containing a jewel.

Output Format

For each test case, print n lines containing m characters denoting the configuration of the jewels and the ropes. The jewel pairs should be as many as possible.

Each character should be one of the following:

- `#` denotes a cell with jewels. The places of jewels should be the same as the input.
- One of `ABCDEF` denotes a cell with ropes. Refer to the figure in the problem description to know the meaning for each character. In formal:
  - `A` denotes a cell connecting the cell to the top and to the right.
  - `B` denotes a cell connecting the cell to the bottom and to the right.
  - `C` denotes a cell connecting the cell to the bottom and to the left.
  - `D` denotes a cell connecting the cell to the top and to the left.
  - `E` denotes a cell connecting the cell to the top and to the bottom.
  - `F` denotes a cell connecting the cell to the left and to the right.
- `.` denotes a cell with nothing.

Technical Specification

- $1 \leq T \leq 10$
- $1 \leq n, m \leq 100$

Sample Input 1

```
2
2 2
#.
.#
3 4
#..#
..#.
#..#
```

Sample Output 1

```
#C
.#
#FF#
BF#.
#..#
```

Problem K
k-restricted Induced Subgraphs

Time limit: 2 seconds

Memory limit: 2048 megabytes

Problem Description

You are given an undirected graph G of n vertices and m edges. The vertices in G are numbered from 1 to n. For each vertex i in G, a weight a_i is associated with it.

Given an integer k, define a k-restricted induced subgraph of G are induced subgraphs satisfying all the following conditions:

- The induced subgraph is connected.
- For every pair of vertices u, v in the induced subgraph, $|a_u - a_v| \leq k$ should hold. Note that an edge does not necessarily connect u and v.

What is the maximum possible number of vertices in a k-restricted induced subgraph of G?

Input Format

The first line of the input contains three integers n, m, k. The i-th of the next m lines contains two integers u_i and v_i denoting an edge (u_i, v_i).

Output Format

Print the maximum number of vertices in a k-restricted induced subgraph of G.

Technical Specification

- $1 \leq n, m, k \leq 10^5$
- $1 \leq a_i \leq 10^5$ for $i = 1, 2, \ldots, n$
- $1 \leq u_i, v_i \leq n$ for $i = 1, 2, \ldots, n$

Sample Input 1

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

Sample Output 1

```
4
```

Sample Input 2

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

Sample Output 2

```
5
```

Sample Input 3

```
4 4 1
100 100 100 100
1 2
2 3
3 4
4 1
```

Sample Output 3

```
4
```

Note

An induced subgraph of G is formed by some non-empty subset of vertices S of G, and all edges in G such that both endpoints are in S. In this problem, we view G also as an induced subgraph of G itself.

Almost blank page

Problem L
Linear Classifers

Time limit: 2 seconds
Memory limit: 2048 megabytes

Problem Description

There are n points on the Cartesian plane. The ith point has coordinate (x_i, y_i). No two points concide and no three points are collinear.

Define a **linear classifier** as a straight line on the Cartesian plane with equation $\frac{q_a}{p_a}x + \frac{q_b}{p_b}y = \frac{q_c}{p_c}$, where $p_a, q_a, p_b, q_b, p_c, q_c$ are integers.

Let's say we have two linear classifiers. If the two classifiers have exactly one intersection, they divide the Cartesian plane into 4 regions. Can you find two linear classifiers so that each of the 4 regions contains an equal number of points? Of course, no points should lie on the linear classifiers, so each region should have exactly $\frac{n}{4}$ points.

Input Format

The first line contains an integer n. Each of the following n lines contain two integers x_i, y_i.

Output Format

Print two lines. On each line, print six integers $p_a, q_a, p_b, q_b, p_c, q_c$ denoting a linear classifier with equation $\frac{q_a}{p_a}x + \frac{q_b}{p_b}y = \frac{q_c}{p_c}$.

Your solution will be considered correct if it satisfies all the following conditions:

- $-10^{18} \le p_a, q_a, p_b, q_b, p_c, q_c \le 10^{18}$ for each linear classifier.
- $p_a, p_b, p_c \neq 0$ for each linear classifier.
- No points lie on the linear classifiers.
- The two linear classifiers have exactly one intersection.
- Each of the 4 regions contains exactly $\frac{n}{4}$ points.

If there are multiple possible solutions, print any. It can be proved that a solution always exists under these constraints.

Technical Specification

- $4 \le n \le 2024$
- $0 \le x_i, y_i \le 10^4$ for $i = 1, 2, \ldots, n$
- $4 \mid n$
- No two points concide and no three points are collinear.

Sample Input 1

```
8
0 0
7 2
4 0
5 7
3 9
8 10
1 6
7 10
```

Sample Output 1

```
5 52 -1 3 5 156
1 1 2 20 1 70
```

Sample Input 2

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

Sample Output 2

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

Note

Here is the figure for the first example. Note that point $D(5, 7)$ does not lie on the green line.

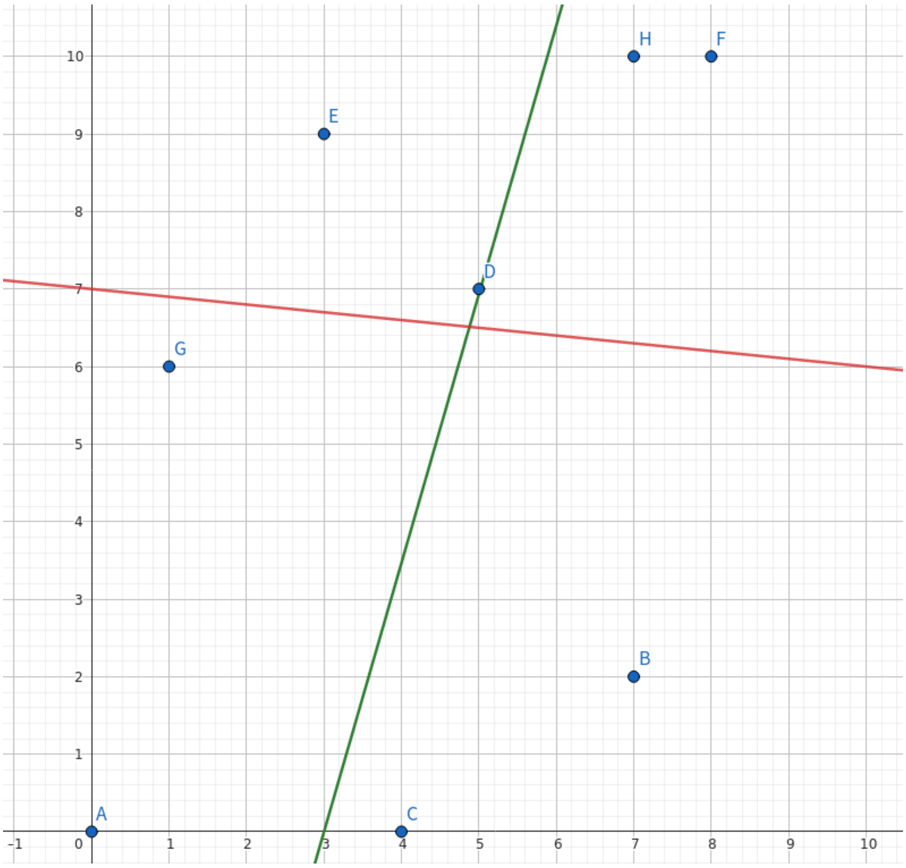

Figure 6: The figure for the first example. The two lines are the linear classifiers.

Almost blank page

Problem M
Mini Factorization Challenge

Time limit: 2 seconds

Memory limit: 2048 megabytes

Problem Description

Factorizing huge integers has caught more and more people's interest since it is unknown whether fast algorithms to factorize huge integers exist. Despite the difficulty, Mia challenges you to factorize a huge positive integer n. As a hint, she also gives you k, the number of positive factors of n.

Soon, you realize that Mia only learns primes less than 100, so all the prime factors in n are less than 100. You also noticed that Mia intentionally miswrote exactly one digit in both n and k. Now you are given n' and k' (the miswritten n and k), and you are to find n and k so that you can successfully factorize n.

Input Format

The first line contains the number of test cases T. Each test case contains two integers n' and k' in one line.

Output Format

Print each test case's a possible value of n and k in one line. n and k should have the same number of digits as n' and k', respectively. n and k should not contain leading zeros.

If there are multiple possible solutions, print the one with the minimum n. It is guaranteed that a solution always exists.

Technical Specification

- $1 \leq T \leq 100$
- $10^5 \leq n' \leq 10^{100} - 1$
- $10 \leq k' \leq 10^{18} - 1$
- n' and k' do not contain leading zeros.

Sample Input 1

```
1
10000 10
```

Sample Output 1

```
10004 12
```

Sample Input 2

```
2
12345 96
98789 10
```

Sample Output 2

```
12045 16
98189 12
```

Problem N
Nancy's Numbers

Time limit: 2 seconds

Memory limit: 2048 megabytes

Problem Description

Nancy has a huge collection of n positive integers $a_1, a_2, \cdots, a_n$. Unfortunately, she is not satisfied since there are duplicate integers in them.

To make Nancy happy, you can perform the following operation any number of times (possibly zero): Select an integer i from 1 to n (inclusive), and add 1 to a_i.

What is the minimum number of operations required so that all the integers in $a_1, a_2, \cdots, a_n$ are distinct (in other words, all integers are different)?

Input Format

The first line of the input contains an integer n. The second line of the input contains n integers $a_1, a_2, \cdots, a_n$.

Output Format

Print the minimum number of operations required to make all integers in $a_1, a_2, \cdots, a_n$ distinct.

Technical Specification

- $1 \le n \le 2 \times 10^5$
- $1 \le a_i \le 10^9$ for $i = 1, 2, \ldots, n$

Sample Input 1

```
7
3 1 4 1 5 9 2
```

Sample Output 1

```
5
```

Sample Input 2

```
1
77777
```

Sample Output 2

```
0
```

Sample Input 3

```
5
100 100 100 100 100
```

Sample Output 3

```
10
```