Codeforces Round #635 (Div. 1)

A. Linova and Kingdom

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Writing light novels is the most important thing in Linova's life. Last night, Linova dreamed about a fantastic kingdom. She began to write a light novel for the kingdom as soon as she woke up, and of course, she is the queen of it.

There are n cities and $n-1$ two-way roads connecting pairs of cities in the kingdom. From any city, you can reach any other city by walking through some roads. The cities are numbered from 1 to n, and the city 1 is the capital of the kingdom. So, the kingdom has a tree structure.

As the queen, Linova plans to choose **exactly** k cities developing industry, while the other cities will develop tourism. The capital also can be either industrial or tourism city.

A meeting is held in the capital once a year. To attend the meeting, each **industry city** sends an envoy. All envoys will follow the shortest path from the departure city to the capital (which is unique).

Traveling in tourism cities is pleasant. For each envoy, his *happiness* is equal to the number of **tourism cities** on his path.

In order to be a queen loved by people, Linova wants to choose k cities which can maximize the sum of *happinesses* of all envoys. Can you calculate the maximum sum for her?

Input

The first line contains two integers n and k ($2 \le n \le 2 \cdot 10^5$, $1 \le k < n$) — the number of cities and industry cities respectively.

Each of the next $n-1$ lines contains two integers u and v ($1 \le u, v \le n$), denoting there is a road connecting city u and city v.

It is guaranteed that from any city, you can reach any other city by the roads.

Output

Print the only line containing a single integer — the maximum possible sum of *happinesses* of all envoys.

Examples

input

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

output

```
7
```

input

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

output

```
2
```

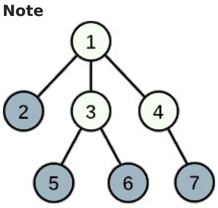
input

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

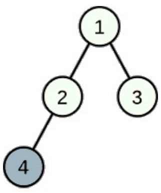
output

```
9
```

Note



In the first example, Linova can choose cities 2, 5, 6, 7 to develop industry, then the *happiness* of the envoy from city 2 is 1, the *happiness* of envoys from cities 5, 6, 7 is 2. The sum of *happinesses* is 7, and it can be proved to be the maximum one.



In the second example, choosing cities 3, 4 developing industry can reach a sum of 3, but remember that Linova plans to choose **exactly** k cities developing industry, then the maximum sum is 2.

B. Xenia and Colorful Gems

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Xenia is a girl being born a noble. Due to the inflexibility and harshness of her family, Xenia has to find some ways to amuse herself.

Recently Xenia has bought n_r red gems, n_g green gems and n_b blue gems. Each of the gems has a weight.

Now, she is going to pick three gems.

Xenia loves colorful things, so she will pick exactly one gem of each color.

Xenia loves balance, so she will try to pick gems with little difference in weight.

Specifically, supposing the weights of the picked gems are x, y and z, Xenia wants to find the minimum value of $(x-y)^2+(y-z)^2+(z-x)^2$. As her dear friend, can you help her?

Input

The first line contains a single integer t $(1 \le t \le 100)$ — the number of test cases. Then t test cases follow.

The first line of each test case contains three integers n_r, n_g, n_b $(1 \le n_r, n_g, n_b \le 10^5)$ — the number of red gems, green gems and blue gems respectively.

The second line of each test case contains n_r integers $r_1, r_2, \ldots, r_{n_r}$ $(1 \le r_i \le 10^9)$ — r_i is the weight of the i-th red gem.

The third line of each test case contains n_g integers $g_1, g_2, \ldots, g_{n_g}$ $(1 \le g_i \le 10^9)$ — g_i is the weight of the i-th green gem.

The fourth line of each test case contains n_b integers $b_1, b_2, \ldots, b_{n_b}$ $(1 \le b_i \le 10^9)$ — b_i is the weight of the i-th blue gem.

It is guaranteed that $\sum n_r \le 10^5$, $\sum n_g \le 10^5$, $\sum n_b \le 10^5$ (the sum for all test cases).

Output

For each test case, print a line contains one integer — the minimum value which Xenia wants to find.

Example

input

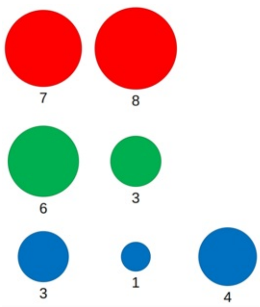
```
5
2 2 3
7 8
6 3
3 1 4
1 1 1
1
1
1000000000
2 2 2
1 2
5 4
6 7
2 2 2
1 2
3 4
6 7
3 4 1
3 2 1
7 3 3 4
6
```

output

```
14
1999999996000000002
24
24
14
```

Note

In the first test case, Xenia has the following gems:



If she picks the red gem with weight 7, the green gem with weight 6, and the blue gem with weight 4, she will achieve the most balanced selection with $(x-y)^2+(y-z)^2+(z-x)^2=(7-6)^2+(6-4)^2+(4-7)^2=14$.

C. Kaavi and Magic Spell

time limit per test: 2 seconds

memory limit per test: 512 megabytes

input: standard input

output: standard output

Kaavi, the mysterious fortune teller, deeply believes that one's fate is inevitable and unavoidable. Of course, she makes her living by predicting others' future. While doing divination, Kaavi believes that magic spells can provide great power for her to see the future.

Kaavi has a string T of length m and all the strings with the prefix T are magic spells. Kaavi also has a string S of length n and an empty string A.

During the divination, Kaavi needs to perform a sequence of operations. There are two different operations:

- Delete the first character of S and add it at the **front** of A.
- Delete the first character of S and add it at the **back** of A.

Kaavi can perform **no more than** n operations. To finish the divination, she wants to know the number of different operation sequences to make A a magic spell (i.e. with the prefix T). As her assistant, can you help her? The answer might be huge, so Kaavi only needs to know the answer modulo $998\,244\,353$.

Two operation sequences are considered different if they are different in length or there exists an i that their i-th operation is different.

A substring is a contiguous sequence of characters within a string. A prefix of a string S is a substring of S that occurs at the beginning of S.

Input

The first line contains a string S of length n $(1 \le n \le 3000)$.

The second line contains a string T of length m ($1 \le m \le n$).

Both strings contain only lowercase Latin letters.

Output

The output contains only one integer — the answer modulo $998\,244\,353$.

Examples

input
```
abab
ba
```
output
```
12
```

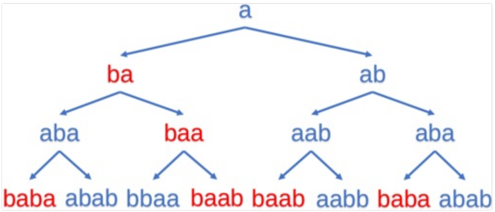
input
```
defineintlonglong
signedmain
```
output
```
0
```

input
```
rotator
rotator
```
output
```
4
```

input
```
cacdcdbbbb
bdcaccdbbb
```
output
```
24
```

Note

The first test:



The red ones are the magic spells. In the first operation, Kaavi can either add the first character "a" at the front or the back of A, although the results are the same, they are considered as different operations. So the answer is $6 \times 2 = 12$.

D. Yui and Mahjong Set

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

This is an interactive problem.

Yui is a girl who enjoys playing Mahjong.

She has a mysterious set which consists of tiles (this set can be empty). Each tile has an integer value between 1 and n, and **at most n tiles** in the set have the same value. So the set can contain at most n^2 tiles.

You want to figure out which values are on the tiles. But Yui is shy, she prefers to play a guessing game with you.

Let's call a set consisting of **three** tiles *triplet* if their values are the same. For example, $\{2,\ 2,\ 2\}$ is a triplet, but $\{2,\ 3,\ 3\}$ is not.

Let's call a set consisting of **three** tiles *straight* if their values are consecutive integers. For example, $\{2,\ 3,\ 4\}$ is a straight, but $\{1,\ 3,\ 5\}$ is not.

At first, Yui gives you the number of triplet subsets and straight subsets of the initial set respectively. After that, you can insert a tile with an integer value between 1 and n into the set **at most n times**. Every time you insert a tile, you will get the number of triplet subsets and straight subsets of the current set as well.

Note that two tiles with the same value are treated different. In other words, in the set $\{1,\ 1,\ 2,\ 2,\ 3\}$ you can find 4 subsets $\{1,\ 2,\ 3\}$.

Try to guess the number of tiles in the initial set with value i for all integers i from 1 to n.

Input

The first line contains a single integer n ($4 \le n \le 100$).

The second line contains two integers which represent the number of triplet subsets and straight subsets of the initial set respectively.

Output

When you are ready to answer, print a single line of form "`!` a_1 a_2 `...` a_n" ($0 \le a_i \le n$), where a_i is equal to the number of tiles in the initial set with value i.

Interaction

To insert a tile, print a single line of form "`+` x" ($1 \le x \le n$), where x is the value of the tile you insert. Then you should read two integers which represent the number of triplet subsets and straight subsets of the current set respectively.

After printing a line, do not forget to flush the output. Otherwise, you will get `Idleness limit exceeded`. To do this, use:

- `fflush(stdout)` or `cout.flush()` in C++;
- `System.out.flush()` in Java;
- `flush(output)` in Pascal;
- `stdout.flush()` in Python;
- see documentation for other languages.

You will get `Wrong answer` if you insert more than n tiles.

Hacks

To make a hack you should provide a test in such format:

The first line contains a single integer n ($4 \le n \le 100$).

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le n$) — a_i is equal to the number of tiles with value i in the set.

Example

input

```
5
1 6
2 9
5 12
5 24
6 24
```

output

```
+ 1
+ 1
+ 2
+ 5
! 2 1 3 0 2
```

Note

In the first test, the initial set of tiles is $\{1, 1, 2, 3, 3, 3, 5, 5\}$. It has only one triplet subset $\{3, 3, 3\}$ and six straight subsets, all equal to $\{1, 2, 3\}$. After inserting a tile with value 1 the set of tiles will be $\{1, 1, 1, 2, 3, 3, 3, 5, 5\}$ and will have two triplet subsets $\{1, 1, 1\}$, $\{3, 3, 3\}$ and nine straight subsets, all equal to $\{1, 2, 3\}$.

E1. Chiori and Doll Picking (easy version)

time limit per test: 3 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

This is the easy version of the problem. The only difference between easy and hard versions is the constraint of m. You can make hacks only if both versions are solved.

Chiori loves dolls and now she is going to decorate her bedroom!

As a doll collector, Chiori has got n dolls. The i-th doll has a non-negative integer value a_i ($a_i < 2^m$, m is given). Chiori wants to pick some (maybe zero) dolls for the decoration, so there are 2^n different picking ways.

Let x be the bitwise-xor-sum of values of dolls Chiori picks (in case Chiori picks no dolls $x = 0$). The value of this picking way is equal to the number of 1-bits in the binary representation of x. More formally, it is also equal to the number of indices $0 \le i < m$, such that $\left\lfloor \frac{x}{2^i} \right\rfloor$ is odd.

Tell her the number of picking ways with value i for each integer i from 0 to m. Due to the answers can be very huge, print them by modulo $998\,244\,353$.

Input

The first line contains two integers n and m ($1 \le n \le 2 \cdot 10^5$, $0 \le m \le 35$) — the number of dolls and the maximum value of the picking way.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i < 2^m$) — the values of dolls.

Output

Print $m + 1$ integers $p_0, p_1, \ldots, p_m$ — p_i is equal to the number of picking ways with value i by modulo $998\,244\,353$.

Examples

input

```
4 4
3 5 8 14
```

output

```
2 2 6 6 0
```

input

```
6 7
11 45 14 9 19 81
```

output

```
1 2 11 20 15 10 5 0
```

E2. Chiori and Doll Picking (hard version)

time limit per test: 3 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

This is the hard version of the problem. The only difference between easy and hard versions is the constraint of m. You can make hacks only if both versions are solved.

Chiori loves dolls and now she is going to decorate her bedroom!

As a doll collector, Chiori has got n dolls. The i-th doll has a non-negative integer value a_i ($a_i < 2^m$, m is given). Chiori wants to pick some (maybe zero) dolls for the decoration, so there are 2^n different picking ways.

Let x be the bitwise-xor-sum of values of dolls Chiori picks (in case Chiori picks no dolls $x = 0$). The value of this picking way is equal to the number of 1-bits in the binary representation of x. More formally, it is also equal to the number of indices $0 \le i < m$, such that $\left\lfloor \frac{x}{2^i} \right\rfloor$ is odd.

Tell her the number of picking ways with value i for each integer i from 0 to m. Due to the answers can be very huge, print them by modulo $998\,244\,353$.

Input

The first line contains two integers n and m ($1 \le n \le 2 \cdot 10^5$, $0 \le m \le 53$) — the number of dolls and the maximum value of the picking way.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i < 2^m$) — the values of dolls.

Output

Print $m + 1$ integers $p_0, p_1, \ldots, p_m$ — p_i is equal to the number of picking ways with value i by modulo $998\,244\,353$.

Examples

input

```
4 4
3 5 8 14
```

output

```
2 2 6 6 0
```

input

```
6 7
11 45 14 9 19 81
```

output

```
1 2 11 20 15 10 5 0
```

F. Journey

time limit per test: 4 seconds
memory limit per test: 1024 megabytes
input: standard input
output: standard output

In the wilds far beyond lies the Land of Sacredness, which can be viewed as a tree — connected undirected graph consisting of n nodes and $n-1$ edges. The nodes are numbered from 1 to n.

There are m travelers attracted by its prosperity and beauty. Thereupon, they set off their journey on this land. The i-th traveler will travel along the shortest path from s_i to t_i. In doing so, they will *go through* all edges in the shortest path from s_i to t_i, which is unique in the tree.

During their journey, the travelers will acquaint themselves with the others. Some may even become friends. To be specific, the i-th traveler and the j-th traveler will become friends if and only if there are **at least** k edges that both the i-th traveler and the j-th traveler will *go through*.

Your task is to find out the number of pairs of travelers (i, j) satisfying the following conditions:

- $1 \le i < j \le m$.
- the i-th traveler and the j-th traveler will become friends.

Input

The first line contains three integers n, m and k ($2 \le n, m \le 1.5 \cdot 10^5$, $1 \le k \le n$).

Each of the next $n-1$ lines contains two integers u and v ($1 \le u, v \le n$), denoting there is an edge between u and v.

The i-th line of the next m lines contains two integers s_i and t_i ($1 \le s_i, t_i \le n$, $s_i \neq t_i$), denoting the starting point and the destination of i-th traveler.

It is guaranteed that the given edges form a tree.

Output

The only line contains a single integer — the number of pairs of travelers satisfying the given conditions.

Examples

input

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

output

```
4
```

input

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

output

```
1
```

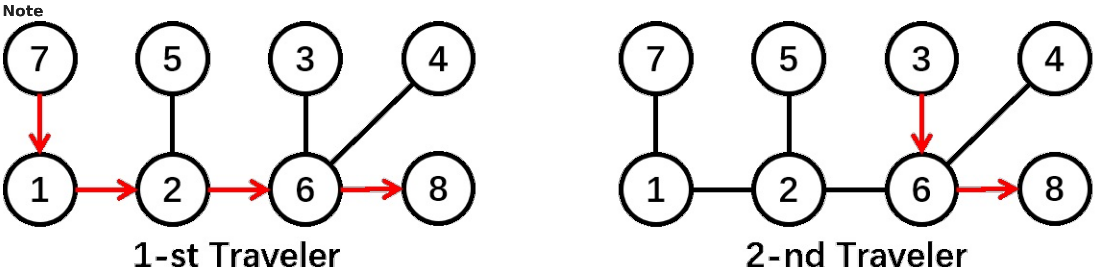
input

```
13 8 3
7 6
9 11
5 6
11 3
9 7
2 12
4 3
1 2
5 8
6 13
5 10
3 1
10 4
10 11
8 11
4 9
2 5
3 5
7 3
8 10
```

output

```
14
```

Note



In the first example there are 4 pairs satisfying the given requirements: $(1, 2)$, $(1, 3)$, $(1, 4)$, $(3, 4)$.

- The 1-st traveler and the 2-nd traveler both go through the edge $6-8$.
- The 1-st traveler and the 3-rd traveler both go through the edge $2-6$.
- The 1-st traveler and the 4-th traveler both go through the edge $1-2$ and $2-6$.
- The 3-rd traveler and the 4-th traveler both go through the edge $2-6$.