A. Hard Code

Description

Some strange code is sent to Da Shan High School. It's said to be the prophet's note. The note is extremely hard to understand. However, Professor Meng is so smart that he successfully found the pattern of the code. That is, the length of the code is the product of two prime numbers. He tries to reallocate the code into a grid of size N*M, where M is the bigger prime. In specific, he writes down the letters of the code to the cells one by one, from left to right, and from top to button. In this way, he found the code eventually readable.

Professor Meng wants to know all the secrets of the note right now. But he doesn't take his computer with him. Can you help him?

Input

The first line of the input is L ($L \le 1000$), which means the number of test cases.

For each test case, the first line contains two prime numbers, which indicates N, M ($0 < N * M \le 1000$) as describe above. The second line contains a string, i.e., the code, containing only lowercase letters. It's guaranteed the length of the string equals to N * M.

Output

For each test case, output N lines, each line with M letters representing the readable code after the reallocation.

Sample Input

```
1

2 5

klmbbileay
```

Sample Output

```
klmbb

ileay
```

B. Golden Radio Base

Background

Golden ratio base (GRB) is a non-integer positional numeral system that uses the golden ratio (the irrational number $(1+\sqrt{5})/2 \approx 1.61803399$ symbolized by the Greek letter ϕ) as its base. It is sometimes referred to as base-ϕ, golden mean base, phi-base, or, phi-nary.

Any non-negative real number can be represented as a base-ϕ numeral using only the digits 0 and 1, and avoiding the digit sequence "11" – this is called a *standard form*. A base-ϕ numeral that includes the digit sequence "11" can always be rewritten in standard form, using the algebraic properties of the base ϕ — most notably that $\phi + 1 = \phi^2$. For instance, $11(\phi) = 100(\phi)$. Despite using an irrational number base, when using standard form, all on-negative integers have a unique representation as a terminating (finite) base-ϕ expansion. The set of numbers which possess a finite base-ϕ representation is the ring $\mathbf{Z}[1 + \sqrt{5}/2]$; it plays the same role in this numeral systems as dyadic rationals play in binary numbers, providing a possibility to multiply.

Other numbers have standard representations in base-ϕ, with rational numbers having recurring representations. These representations are unique, except that numbers (mentioned above) with a terminating expansion also have a non-terminating expansion, as they do in base-10; for example, 1=0.99999….

Description

Coach MMM, an Computer Science Professor who is also addicted to Mathematics, is extremely interested in GRB and now ask you for help to write a converter which, given an integer N in base-10, outputs its corresponding form in base-ϕ.

Input

There are multiple test cases. Each line of the input consists of one positive integer which is not larger than 10^9. The number of test cases is less than 10000. Input is terminated by end-of-file.

Output

For each test case, output the required answer in a single line. Note that trailing 0s after the decimal point should be wiped. Please see the samples for more details.

Sample Input

```
1
2
3
6
10
```

Sample Output

```
1
10.01
100.01
1010.0001
10100.0101
```

Hint

Besides $\phi + 1 = \phi^2$, we have another useful property of GRB, i.e., $2 * \phi^2 = \phi^3 + 1$.

C. Little Tiger vs. Deep Monkey

Description

A crowd of little animals is visiting a mysterious laboratory – The Deep Lab of SYSU.

"Are you surprised by the STS (speech to speech) technology of Microsoft Research and the cat face recognition project of Google and academia? Are you curious about what technology is behind those fantastic demos?" asks the director of the Deep Lab. "Deep learning, deep learning!" Little Tiger raises his hand briskly. "Yes, clever boy, that's deep learning (深度学习/深度神经网络)", says the director. "However, they are only 'a piece of cake'. I won't tell you a top secret that our lab has invented a Deep Monkey (深猴) with more advanced technology. And that guy is as smart as human!"

"Nani ?!" Little Tiger doubts about that as he is the smartest kid in his kindergarten; even so, he is not as smart as human, "how can a monkey be smarter than me? I will challenge him."

To verify their research achievement, the researchers of the Deep Lab are going to host an intelligence test for Little Tiger and Deep Monkey.

The test is composed of N binary choice questions. And different questions may have different scores according to their difficulties. One can get the corresponding score for a question if he chooses the correct answer; otherwise, he gets nothing. The overall score is counted as the sum of scores one gets from each question. The one with a larger overall score wins; tie happens when they get the same score.

Little Tiger assumes that Deep Monkey will choose the answer randomly as he doesn't believe the monkey is smart. Now, Little Tiger is wondering "what score should I get at least so that I will not lose in the contest with probability of at least P? ". As little tiger is a really smart guy, he can evaluate the answer quickly.

You, Deep Monkey, can you work it out? Show your power!

Input

The first line of input contains a single integer T (1 ≤ T ≤ 10) indicating the number of test cases. Then T test cases follow.

Each test case is composed of two lines. The first line has two numbers N and P separated by a blank. N is an integer, satisfying 1 ≤ N ≤ 40. P is a floating number with at most 3 digits after the decimal point, and is in the range of [0, 1]. The second line has N numbers separated by blanks, which are the scores of each question. The score of each questions is an integer and in the range of [1, 1000]

Output

For each test case, output only a single line with the answer.

Sample Input

```
1

3 0.5

1 2 3
```

Sample Output

```
3
```

D. Bathysphere

Description

The Bathysphere is a spherical deep-sea submersible which was unpowered and lowered into the ocean on a cable, and was used to conduct a series of dives under the sea. The Bathysphere was designed for studying undersea wildlife.

The Bathysphere was conducted from the deck of a ship. After counted, the ship should not move, so choosing the position where the Bathysphere was conducted is important.

A group of scientists want to study the secrets of undersea world along the equator, and they would like to use the Bathysphere. They want to choose the position where the Bathysphere can dive as deep as possible. Before conducting the Bathysphere, they have a map of the seabed, which tell them the shape of the seabed. They draw a line on the equator of the map to mark where they will release the Bathysphere, as a number axis. Suppose the axis is draw from 0 to L. But when they release the Bathysphere, they can't know where they are accurately, i.e., if they choose position x to release the Bathysphere, the real position will distribute between x-d and x+d with an equal probability, where d is given. The objective of the scientists is very simple, i.e., to maximize the expected depth.

For the ease of presentation, the shape of the seabed is described as a poly line. Given N points (X_i, Y_i) as the vertices, where X_i and Y_i indicate the position and the depth of the i-th vertex, respectively, the ploy line is composed of the line segments that connect consecutive vertices.

Input

The first line contains an integer T (1 ≤ T ≤ 25), the number of test cases.

Then T test cases follow. In each test case, the first line contains two integers N (2 ≤ N ≤ 2*10^5) and L (2 ≤ L ≤ 10^9), as described above. Then N lines follow, each line contains two integer X_i and Y_i (1≤i≤N, 0≤ Y_i ≤10^9), where point (X_i, Y_i) is a vertex of the ploy line. It is assumed that X_1 == 0 and X_n == L and $X_i < X_{i+1}$ for 1 ≤ i < N. Then the following line contains one integer d (0 ≤ d ≤ L/2), as described above.

Output

For each test case, choose a position between d and L-d, both inclusive, to conducted the Bathysphere, and calculate the expected depth. Output the expected depth in a line, rounded to 3 digits after the decimal point.

Sample Input

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

Sample Output

```
5.900
9.192
```

E. Min-max-multiply

Description

MMM is solving a problem on an online judge, but her solution got TLE (Time limit exceeded). Here is her submitted Java solution:

```
public class PleaseUseProperClassNameAccordingToYourContestEnvironment {
    private static Scanner cin;
    private static int n;
    private static char[] command;
    private static long[] arr;
    private static long minValue, maxValue;

    public static void read() {
        n = cin.nextInt();
        minValue = cin.nextLong();
        maxValue = cin.nextLong();
        command = new char[n];
        arr = new long[n];
        cin.nextLine();

        String[] tokens = cin.nextLine().split(" ");
        for (int i = 0; i < n; i++) {
            command[i] = tokens[i].charAt(0);
            arr[i] = Long.valueOf(tokens[i].substring(1));
        }
    }

    public static void go() {
```

```
    int numQuery = cin.nextInt();
    long ans = 0;
    long y, y0;
    for (int i = 0; i < numQuery; i++) {
        y0 = cin.nextLong();
        y = y0;
        for (int j = 0; j < n; ++j) {
            switch (command[j]) {
                case '+':
                    y += arr[j];
                    break;
                case '-':
                    y -= arr[j];
                    break;
                case '*':
                    y *= arr[j];
                    break;
                case '@':
                    y += y0 * arr[j];
            }

            y = Math.min(maxValue, y);
            y = Math.max(minValue, y);
        }
        ans += y;
    }
    System.out.println(ans);
}
public static void main(String argv[]) throws IOException {
```

```
        cin = new Scanner(System.in);


        int numTest = cin.nextInt();
        while (numTest-- > 0) {
            read();
            go();
        }
    }
}
```

She thought that maybe this is due to the slowness of Java compared to C++. So she changed her program into C++, however, she got TLE again:

```
const int kMaxN = 1000000;
char command[kMaxN];
long long arr[kMaxN];
int main() {
    int num_case = 0;
    scanf("%d", &num_case);
    assert(1 <= num_case && num_case <= 10);
    for (int icase = 0; icase < num_case; ++icase) {
        int n;
        long long min_value, max_value;
        scanf("%d%lld%lld", &n, &min_value, &max_value);
        assert(1 <= n && n <= 1000000);
        assert(1 <= min_value && min_value <= max_value);
        assert(max_value <= 10000000000LL);    // 10^10
        for (int i = 0; i < n; ++i) {
            scanf(" %c%lld", command + i, arr + i);
```

```
    assert(1 <= arr[i] && arr[i] <= 10000000000LL);    // 10^10
    if (command[i] == '*' || command[i] == '@') {
        assert(arr[i] <= 100000);    // 10^5
    }
}

int num_query;
scanf("%d", &num_query);
assert(num_query <= 1000000);
long long ans = 0;
for (int iquery = 0; iquery < num_query; ++iquery) {
    long long start_value;
    scanf("%lld", &start_value);
    assert(1 <= start_value && start_value <= 10000000000LL);    // 10^10
    long long sum = start_value;
    for (int i = 0; i < n; ++i) {
        switch(command[i]) {
            case '+':
                sum += arr[i];
                break;
            case '-':
                sum -= arr[i];
                break;
            case '@':
                sum += start_value * arr[i];
                break;
            default:
                assert(command[i] == '*');
                sum *= arr[i];
```

```
            }

            sum = min(sum, max_value);
            sum = max(sum, min_value);
        }
        ans += sum;
    }
    printf("%lld\n", ans);
  }
  return 0;
}
```

MMM was so desperate that she asked the judge for help. The judge found out that both programs produce the correct output; however, they cannot finish execution within the time limit. Could you, our talented contestant, help her optimize the algorithm and got AC?

Input

The first line of input is an integer T (0 < T ≤ 100) indicating the number of test cases.

For each case, there are 3 integers n, min_value, max_value in the first line, which denote the number of operations, the minimum and maximum value after each operation, respectively (1 ≤ n ≤ 10^6, 1 ≤ min_value ≤ max_value ≤ 10^10).

Then there are n operations in the next line. Each operation is an operator, '+', '-', '@' or '*', followed by a positive integer, without any spaces between them.

For operations of format +x and -x, you can assume 1 ≤ x ≤ 10^10.

And for operations of format *x and @x, you can assume 1 ≤ x ≤ 10^5.

After the line of operations, there would be an integer num_query indicating the number of queries (1 ≤ num_query ≤ 10^6).

Then num_query integers x_i follows, each of them is of range 1 ≤ x_i ≤ 10^10

Output

For each case, output num_query numbers. Please refer to the problem statement for the meaning of the result.

Sample Input

```
3


9 1 10

-6 +1 +2 +3 -4 *3 -5 @1 -5

10

1

2

3

4

5

6

7

8

9

10


6 1 10

-6 +1 +2 +3 -4 *3

10

1

2

3

4

5
```

```
6

7

8

9

10


2 20 25

+20 -1

1

3
```

Sample Output

```
36

93

22
```

Hint

Don't submit your code until you believe that significant optimization over MMM's solution has been achieved. MMM's solution would run for an hour on our test data but we expect your solution to finish within seconds.

F. RP problem

Description

As an ACMer, you must have heard of the legend of Ren Pin (RP), which means a person's lucky degree. Neither can RP be created, nor eliminated without foundation. It can only be transferred from one person to another. Moreover, everyone has his social circle. It's guaranteed that each person has at least one friend and he cannot be the friend of himself. The relationship is directed. In RP system, one's RP is transferred from himself to all of his friends uniformly every day. For example, A has three friends, B, C and D and his current RP is 1. In tomorrow, 1/3 of his RP will be transferred to B, 1/3 to C and 1/3 to D and RP from those treat him as friends will be transferred to him as well.

On the other hand, the god in charge of the RP system is tired of the transferring RP from one person to another every day. Making the assumption that the total amount of RP is 1, he wants to know whether there is a stable distribution in RP system, so that by reallocating the RP of each person, he can lighten his workload. Stable distribution means if the structure of social network keeps unchanged, no matter how many days pass by, everyone's RP keeps unchanged. You will see an example as follows. For the ease of presentation, let "X->Y" denotes X takes Y as his friend. There are only three persons A, B and C in the world with A->B, B->C, C->A and C->B. If the current RPs of A, B, C are 0.2, 0.4, 0.4 respectively, then the RPs will still be 0.2, 0.4, 0.4 in tomorrow. It's obviously that the distribution keeps unchanged no matter how many days pass by, so (0.2, 0.4, 0.4) is a stable distribution. However, if the current RPs of A, B, C are 0.3, 0.3, 0.4, respectively, then the RPs will be 0.2, 0.5, 0.3 in tomorrow, so (0.3, 0.3, 0.4) is not a stable distribution. The god wonders, for a given a social network, how many stable distributions exist?

Furthermore, if there is one and only one stable distribution, your friend A, who is lack of luck, wants to know whether he can increase his RP by making one more new friend. More specifically, letting RP(A, G) be the RP of A in the stable distribution for a network G, the RP of A after adding A->X to G is RP(A, G ⋁ {A->X}). Your friend A wants to know if there exists a person X, such that RP(A, G ⋁ {A->X}) is strictly larger than RP(A, G). For example, there are four persons A, B, C, D in the world with A->B, B->C, C->A, D->A. The only one stable distribution is (1/3, 1/3, 1/3, 0). If A->D is added to the network, the stable distribution will be (2/5, 1/5, 1/5, 1/5) and 2/5 > 1/3, so A can increase his RP by making friend with D. If there are multiple qualifying persons, your friend A wants to know which one can increase his RP to the maximum extent.

Input

The first line of input contains an integer T (T ≤ 50) indicating the number of datasets.

Each dataset starts with two integers n and m (n ≤ 100, n ≤ m ≤ n*(n-1)), where n and m indicate the number of persons and the number of relationships, respectively. Next m lines describe the relationships in the social network. Each of these lines will contain two integers u and v (0 ≤ u, v < n), indicating u->v.

Output

For each test case, output a single line. If the number of stable distributions is infinite, print "INF", otherwise, print the number of stable distributions. If there is only one stable distribution, print the person whom your friend A can increase the most RP by making friend with. If there exists multiple such persons, print the one with the smallest ID. However, if no person satisfies the requirement, print "-1".

Your friend A's ID is always (n-1).

Sample Input

```
3
4 4
0 1
1 0
2 3
3 2
3 3
0 1
1 2
2 0
3 3
0 1
1 0
2 1
```

Sample Output

```
INF
1 1
1 -1
```

G. Mosaic

Description

The God of sheep decides to pixelate some pictures (i.e., change them into pictures with mosaic). Here's how he is gonna make it: for each picture, he divides the picture into n x n cells, where each cell is assigned a color value. Then he chooses a cell, and checks the color values in the L x L region whose center is at this specific cell. Assuming the maximum and minimum color values in the region is A and B respectively, he will replace the color value in the chosen cell with floor((A + B) / 2).

Can you help the God of sheep?

Input

The first line contains an integer T (T ≤ 5) indicating the number of test cases. Then T test cases follow.

Each test case begins with an integer n (5 < n < 800). Then the following n rows describe the picture to pixelate, where each row has n integers representing the original color values. The j-th integer in the i-th row is the color value of cell (i, j) of the picture. Color values are nonnegative integers and will not exceed 1,000,000,000 (10^9).

After the description of the picture, there is an integer Q (Q ≤ 100000 (10^5)), indicating the number of mosaics.

Then Q actions follow: the i-th row gives the i-th replacement made by the God of sheep: xi, yi, Li (1 ≤ xi, yi ≤ n, 1 ≤ Li < 10000, Li is odd). This means the God of sheep will change the color value in (xi, yi) (located at row xi and column yi) according to the Li x Li region as described above. For example, an query (2, 3, 3) means changing the color value of the cell at the second row and the third column according to region (1, 2) (1, 3), (1, 4), (2, 2), (2, 3), (2, 4), (3, 2), (3, 3), (3, 4). Notice that if the region is not entirely inside the picture, only cells that are both in the region and the picture are considered.

Note that the God of sheep will do the replacement one by one in the order given in the input.

Output

For each test case, print a line "Case #t:"(without quotes, t means the index of the test case) at the beginning.

For each action, print the new color value of the updated cell.

Sample Input

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

Sample Output

```
Case #1:
5
6
3
4
6
```

H. Tower

Description

The land price on the MMM island is extremely high. So people on the island all live in a very tall tower with one million floors!! There are N electric service companies (labeled 1 to N) and M power stations on the island. One power station belongs to exactly one company.

At the beginning, the company of each power station is given, and a company can choose one floor to build a power center. Assuming that there are k power stations (on floor a1,a2,...,ak) that belong to a company, and the company chooses floor x to build the power center, the daily cost of this company would be |a1-x|+|a2-x|+...+|ak-x|. MMM, the master of the island, has made a special rule:

x1 <=> x2 <=> x3 <=> ... <=> xn

The operator <=> can be either <= or >=. It means that the power centers chosen by the adjacent companies should follow the specified order. For example, if the rule is x1 <= x2 >= x3, it means that the power center of company 1 should not be higher than that of company 2, and the power center of company 2 should not be lower than that of company 3. Though there is no constraint between company 1 and company 3. If company 1 chooses floor 2 for the power center and company 2 chooses floor 3, the station of company 3 can only be in floor 1, 2 or 3.

Your task is to help the companies calculate their minimum total daily cost.

Input

There are multiple test cases. The first line has an integer T (T ≤ 10), which indicates the number of test cases.

Each test case begins with two integers N and M (1 ≤ N ≤ M ≤ 5*10^5).

There are N - 1 operators on the second line, either '<=' or '>=', separated by spaces, indicating the rules between adjacent companies.

The third line has M pairs of integers. Each pair x_i, y_i describes the i-th power station, where x_i indicates the label of the floor and y_i indicates the label of the company, (1 ≤ x_i ≤ 10^6, 1 ≤ y_i ≤ N). It is guaranteed that each company has at least one power station.

Output

For each test case, output the minimum total daily cost of the island.

Sample Input

```
3
3 5
<= <=
2 1 3 1 4 2 4 2 5 3
3 8
<= <=
7 3 9 3 3 2 4 2 5 2 6 1 7 1 8 1
4 4
<= >= <=
3 1 1 2 4 3 2 4
```

Sample Output

```
1
11
4
```

I. String

Description

Given a string S and two integers L and M, we consider a substring of S as "recoverable" if and only if

(i) It is of length M*L;

(ii) It can be constructed by concatenating M "diversified" substrings of S, where each of these substrings has length L; two strings are considered as "diversified" if they don't have the same character for every position.

Two substrings of S are considered as "different" if they are cut from different part of S. For example, string "aa" has 3 different substrings "aa", "a" and "a".

Your task is to calculate the number of different "recoverable" substrings of S.

Input

The input contains multiple test cases, proceeding to the End of File.

The first line of each test case has two space-separated integers M and L.

The second ine of each test case has a string S, which consists of only lowercase letters.

The length of S is not larger than 10^5, and 1 ≤ M * L ≤ the length of S.

Output

For each test case, output the answer in a single line.

Sample Input

```
3 3
abcabcbcaabc
```

Sample Output

```
2
```

J. Tri-war

Description

Three countries, Red, Yellow, and Blue are in war. The map of battlefield is a tree, which means that there are N nodes and (N – 1) edges that connect all the nodes. Each country has a base station located in one node. All three countries will not place their station in the same node. And each country will start from its base station to occupy other nodes. For each node, country A will occupy it iff other two country's base stations have larger distances to that node compared to country A. Note that each edge is of the same length.

Given three country's base station, you task is to calculate the number of nodes each country occupies (the base station is counted).

Input

The input starts with a single integer T (1 ≤ T ≤ 10), the number of test cases.

Each test cases starts with a single integer N (3 ≤ N ≤ 10 ^ 5), which means there are N nodes in the tree.

Then N - 1 lines follow, each containing two integers u and v (1 ≤ u, v ≤ N, u ≠ v), which means that there is an edge between node u and node v.

Then a single integer M (1 ≤ M ≤ 10 ^ 5) follows, indicating the number of queries.

Each the next M lines contains a query of three integers a, b, c (1 ≤ a, b, c ≤ N, a, b, c are distinct), which indicates the base stations of the three countries respectively.

Output

For each query, you should output three integers in a single line, separated by white spaces, indicating the number of nodes that each country occupies. Note that the order is the same as the country's base station input.

Sample Input

1

9

1 2

1 3

1 4

2 5

2 6

2 7

6 8

6 9

2

1 2 8

2 1 4

Sample Output

3 3 1

6 2 1