103 學年度全國大專電腦軟體設計競賽
台大校內初賽

National Taiwan University

2014/10/04

Table 1: Problem Set Information

	Problem Title	**Runtime Limits**
Problem A	AAAAAAAAAAAAAA...	2 second(s)
Problem B	Fence	8 second(s)
Problem C	ATP and ADP	2 second(s)
Problem D	Warming	2 second(s)
Problem E	Ambiguous Numbers	1 second(s)
Problem F	Idol Master	1 second(s)
Problem G	Mysterious Diamonds	10 second(s)
Problem H	Rumor of Tomato Kingdom	3 second(s)
Problem I	Counting Substrings	3 second(s)
Problem J	Networking Deployment	7 second(s)

Table 2: Compilation Information

Language	Parameters	Version
C	`gcc -lm -std=c11 -O2 -o filename.exe filename.c`	4.9.1
C++	`g++ -lm -std=c++11 -O2 -o filename.exe filename.cpp`	4.9.1

This page is intentionally left blank.

Problem A
AAAAAAAAAAAAAA...

Time Limit: 2 seconds

The magic bacteria **A** grows very fast. If there are x bacteria **A** now, there will be x^x after one day! For example, if there are 3 bacteria **A** in day 0, there will be 27 bacteria **A** in day 1, and 443426488243037769948249630619149892803 bacteria **A** in day 2. As a talent scientist, you must know that how many bacteria **A** will be there in day d, if there are n bacteria **A** in day 0, right? The answer may be very very large, so please output it modulo m.

Input Format

The first line of input contains a integer T, indicating the number of test cases. Each test case contains one line with three integers n, d, m.

- $1 \leq T \leq 1500$
- $1 \leq n, d, m \leq 10000$
- There are at most 30 test cases with $\max(n, d, m) > 1000$.

Output Format

For each test case, output the number of bacteria **A** modulo m in one line.

Sample Input

```
3
1 10000 10000
3 1 1000
3 2 1000
```

Sample Output

```
1
27
803
```

This page is intentionally left blank.

Problem B
Fence

Time Limit: 8 seconds

You are going to buy some land in Meow Kingdom, which is a N Meow meters by M Meow meters grid from $(0, 0)$ to (N, M) on 2-dimension Cartesian coordinate system. The land that you buy should have positive area. You want to build a fence to surround it along the border of the land.

The fence is composed with small pieces. According to the law, each piece of the fence should has exactly one Meow meter length and both the two end points should be on integer lattice points. That is, the end points of a legal piece of the fence must be $(x_1, y_1) - (x_1, y_1 + 1)$ or $(x_1, y_1) - (x_1 + 1, y_1)$, which x_1, y_1 is integer and $0 \leq x_1 < N, 0 \leq y_1 < M$.

The cost to build a piece of the fence is variant on different locations. Find out the minimum cost to build the fence that surround your land. It means that it is impossible to leave your land without crossing the fence.

Input Format

The first line of input contains a integer T ($T \leq 50$), indicating the number of test cases.

Each test case starts with N, M, ($1 \leq N, M \leq 100$), indicating the size of the kingdom. Then followed by $2N + 1$ lines. The $(2i + 1)$-th line contains M integer $A_{i,j}$ ($0 \leq i \leq N, 0 \leq j < M$) and each integer $A_{i,j}$ is the cost to build a fence between the point (i, j) and $(i, j + 1)$. The $(2i + 2)$-th line contains $M + 1$ integer $B_{i,j}$ ($0 \leq i < N, 0 \leq j \leq M$) and each integer $B_{i,j}$ is the cost to build a fence between the point (i, j) and $(i + 1, j)$. All the costs are positive and no more than 100,000.

Output Format

For each test case, only output a line contains the minimum cost to build the fence. Notice that even if your land is neighbor to the border of kingdom, you still have to build the fence on the border.

Sample Input

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

Sample Output

```
9
```

Problem C
ATP and ADP

Time Limit: 2 seconds

Did you heard about ATP? ... and his friend ADP?

Adenosine triphosphate (ATP) and Adenosine diphosphate (ADP) are both organic compounds in metabolism and are essential to the flow of energy.

You know ATP is so cute. You want to observe and take photo of ATP-ADP cycle which is a process by which energy is stored and used in the bodies of many animals.

There are N molecules of ATP and ADP. To simplify this problem, you can consider that all molecule stay in a one-dimensional line. The i-th molecule locates at position x_i and is either a ATP or a ADP. No two molecule occupy the same position and there is at least one ATP.

You wants to take a photo of a contiguous interval of molecule. However, to make it balanced and beautiful, you want to ensure there are equal numbers of ATP and ADP. You want to determine the maximum size of such a beautiful photo, where the size of a photo is the difference between the maximum and minimum positions of the molecule in the photo.

To give you an even better chance of taking a larger photo. You can pick some of ATP as you want and then transform them to ADP by hydrolysis.

Please determine the largest size of a beautiful photo you can take.

Input Format

The first line of input contains a integer T, indicating the number of test cases.

Each test case starts with a single integer N, indicating the number of molecules.

Each of the following N lines contains x_i and c_i, indicating the position and the kind of molecules. c_i is either '`T`' (for ATP) or '`D`' (for ADP).

- $T = 10$
- $2 \leq N \leq 100,000$
- $0 \leq x_i \leq 1,000,000,000$ and for all $i \neq j$, $x_i \neq x_j$
- $c_i =$ '`T`' or '`D`' and there is at least one ATP.

Output Format

For each test case, only output a line contains the largest size of a beautiful photo you can take, after possibly pick some of ATP and transform them.

Sample Input

```
1
5
8 T
11 D
3 T
10 T
5 D
```

Sample Output

```
7
```

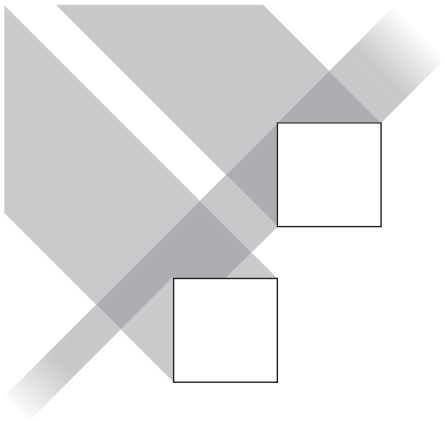
Problem D
Warming

Time Limit: 2 seconds

Global warming becomes an important issue on Meow Planet. As a great scientist, you are trying to predict the temperature in the future. Thus you need to figure out the effect of heating from sunlight.

Since the sun of Meow Planet is far enough, you can assume that the sunlight is parallel and uniformly spread in the space. Meow Planet is a convex polygon. It's obvious that those surface which facing the sun is heating by the sunlight. An 1 Meow meter surface gets 1 Meow joule if the surface is perpendicular to the sunlight. But if the surface is not perpendicular to the sunlight, the energy absorbing may reduce. For example, a surface with 30° angle to sunlight gets only 0.5 Meow joule, since the equivalent length which facing to sun remains half. The equivalent length is defined as $-L\cos\theta$, where L is the length of the surface and θ is the angle between the normal vector of the surface and the direction vector of sunlight.

Besides, Meow Planet also has a Meow Moon, which partially reflect sunlight and is a convex polygon too. The Meow Moon reflects sunlight with a specific ratio *alpha*, which is a real number between 0 to 1. These reflected light should also be considered, but the energy is multipled by *alpha*. Notice that if an area on Meow Planet is simultaneously lighted by sun and moon, the energy from both should be summed together.



Input Format

The first line of input contains a integer T ($T \leq 20$), indicating the number of test cases.

Each test case starts with N, M, α, V_x, V_y. N, M are the number of the points of Meow Planet and Meow Moon and α is the reflection ratio of Meow Moon. V_x, V_y is the X and Y component of the direction vector of sunlight. N, M are positive integer and $3 \leq N, M \leq 50,000$. α, V_x, V_y are real number, which $0 \leq alpha \leq 1$ and $-10 \leq V_x, V_y \leq 10$.

Then followed by $N + M$ lines. Each of the first N lines is a point X_i, Y_i of the polygon of Meow Planet and Each of the rest M lines is a point X_i, Y_i of the polygon of Meow Moon. Both polygon is convex and given in counter-clockwise order. Each point X_i, Y_i of both polygon is a real number that $-100,000 \leq X_i, Y_i \leq 100,000$.

Output Format

For each test case, only output a line contains the total energy absorbed by the Meow Planet. Round off the output to 6th decimal place.

Sample Input

```
1
4 4 0.5 1 -1
0 0
1 0
1 1
0 1
1 1.5
2 1.5
2 2.5
1 2.5
```

Sample Output

```
1.590990
```

Problem E
Ambiguous Numbers

Time Limit: 1 second

As an intelligent hacker, Cebrusfs often play with hex numbers like `0xBADCA7`. But many programs such as `objdump` simply omit the `0x` part, so `0xBADCA7` will be displayed as `BADCA7`. This should be fine in most cases. But if the hex number consists of only `0123456789`, it's hard to distinguish it from a decimal number. We say these numbers are ambiguous. For example, 514 is ambiguous, since `hex(514) = 0x202`; but 50216 is not ambiguous, since `hex(50216) = 0xc428`. Can you tell me how many positive intergers less than or equal to n are ambiguous?

Input Format

The first line of input contains a integer T, indicating the number of test cases. Each test case contains one line with one integer n.

- $1 \leq T \leq 10000$
- $1 \leq n \leq 10^9$

Output Format

For each test case, output the number of ambiguous numbers in one line.

Sample Input

```
2
5
14
```

Sample Output

```
5
9
```

This page is intentionally left blank.

Problem F
Idol Master

Time Limit: 1 second

Idols have become a main trend of artists these years. The number of fans of each idol group increases tremendously. Not only the girl idols are able to lead a huge economical effects, but also the boy idols can make the modern life different. Moreover, the '2D idols' are also available for the comic-anime lovers.

However, it is not easy to make a new idol project popular. Since there are so many customers that may like different type of idols with specific characteristics, a successful idol project usually needs to have the diversity on its members.

Now, you are one of the producers of a rising idol group. There are N girls in your office, each of them have their own specialty and defects on the stage. Nevertheless, since you are a smart producer, at first you want to form a sub-group among them that the members can 'cover' the other members in the sub-group. That is, you want to make a sub-group such that each member can help one of the other members in the sub-group.

In order to do this, your assistants have already evaluated the '*instability*' of each pair of girls in your office. The higher instability a pair has, the higher possibility on making mistakes when the first girl in the pair covers the second girl in the pair. Based on the technical reports, we know that for a particular type of sub-group, which is called 'CircleTP', is a sub-group that "every member in the sub-group is covered by **exactly one** member in the sub-group, and every member in the sub-group also covers **exactly one** member in the sub-group", the overall instability of each member is considered as the 'average' of instabilities of the sub-group members.

Given all the information mentioned above, as a professional producer, you want to find the minimum overall instability if you form a 'CircleTP' sub-group.

Input Format

The first line of input contains a integer T ($T \leq 30$), indicating the number of test cases.

For every test case, the first line has one integers N ($2 \leq N \leq 200$) which indicates the total number of girls in your office. An $N \times N$ matrix M is followed, where the entry a_{ij} equal to the instability when girl i covers girl j ($0 \leq a_{ij} \leq 200000$). If $a_{ij} = -1$, then it means that girl i is not able to cover girl j.

Notice that we let $a_{ii} = -1$ for all i.

Output Format

For each test case, please print a single line with a fraction in form `p/q` denoting the minimum overall instability among all possibly sub-groups.

If there has no valid sub-group, then print "`Impossible`" (without quotes).

The greatest common divisor of `p` and `q` needs to be 1.

Sample Input

```
3
5
-1 7 7 7 7
7 -1 7 7 7
7 7 -1 7 7
7 7 7 -1 7
7 7 7 7 -1
3
-1 10 15
-1 -1 20
-1 -1 -1
6
-1 12 -1 8 -1 -1
-1 -1 9 -1 -1 -1
7 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 10
-1 -1 -1 10 -1 -1
-1 -1 -1 -1 10 -1
```

Sample Output

```
7/1
Impossible
28/3
```

Problem G
Mysterious Diamonds

Time Limit: 10 seconds

Yarg is an famous adventurer who always goes to ancient ruins to find hidden treasures. Now, he is finding treasures in a cave of somewhere in deep Forbidden Mountains. After several hours since he entered the cave, he finally finds out the hidden treasure!

There are N diamonds line on the table in front of Yarg. All of them shine brightly and make unbelievable attractions toward Yarg. When Yarg is going to take them off the table, a sound raises in his mind. "If you see the cursed diamonds in a line, DO NOT TAKE THEM." a wise old man said to Yarg before he climbed up the mountains, "You will die if you didn't solve the puzzle before you take them. Remember this..." Yarg now reminds that he needs to solve the puzzle for dispelling the curse before he take the 'cursed diamonds'. He looks around, and he finds out a piece of paper, which is wrote "*Make the diamonds matched, by the power of fusing.*" After he read the words on paper, that paper burned, and then the second line of stones appeared on the table. The number of the new stones are slightly greater than the number of diamonds, and the sizes seems quite different.

Since Yarg is smart, he remember that this kind of puzzle can be solved by making the sizes of stones identical. Just like the hint on that mysterious paper, two neighborhood stones can be merged into one with summed size. Once the size of each stone is identical to the size of diamond in some corresponding position of the sub-sequence of diamonds, the puzzle is solved. Specifically, suppose the diamonds' size sequence $D =< d_1, d_2, \ldots, d_M >$, and the stones' size sequence $S =< s_1, s_2, \ldots, s_N >$, then Yarg can solve this puzzle at stone-position k iff there exists a sequence $A =< a_0, a_1, \ldots, a_M >$, where $k = a_0$, and $a_{i-1} < a_i$ for $1 \leq i \leq M$, and $\sum_{j=a_{i-1}}^{a_i-1} s_j = d_i$ for $1 \leq i \leq M$.

Now, Yarg wants to know that, **how many different ways** he can solve the puzzle?

Input Format

The first line of input contains a integer T ($T \leq 30$), indicating the number of test cases.

For each test case, first line contains an integer M which denotes the number of diamonds. The second line contains M positive integers $d_1, d_2, \ldots, d_M$ which represent the size of diamonds. The third line contains an integer N which denotes the number of stones. The fourth line contains N positive integers $s_1, s_2, \ldots, s_N$ which represent the size of stones. ($1 \leq M \leq 10^5$, $1 \leq N \leq 10^6$)

Notice that the size sum of all diamonds and the size sum of all stones will not exceed 2×10^6 in a test case.

Output Format

For each test case, please output the number of ways that Yarg can solve the puzzle.

Sample Input

```
2
3
1 1 2
13
1 1 1 1 1 514 1 1 1 1 1 1 1
3
1 1 1
13
1 1 1 1 1 83 1 1 1 1 1 1 1
```

Sample Output

```
6
8
```

Problem H
Rumor of Tomato Kingdom

Time Limit: 3 seconds

You are a fat fat fish living in the Cebrusfs Kingdom. As a fan of MythBusters, you like to challenge anything that seems strange to you.

One day, you heard a rumor about the Tomato Kingdom. It says that **every** citizen in the Tomato Kingdom is kind to foreigners. If a foreigner tries to exchange some coins with them, the total value of coins they give will never less than the value of the coins they get.

In your opinion, it's impossible for all the citizens to be such good guys. There must be some avaricious guys in the Tomato Kingdom. However, all your friends believe the rumor and try to convince you.

To prove them wrong, you went to the Tomato Kingdom. There are three types of coins in the Tomato Kingdom. You prepare many coins of each type and exchange some with N guys.

Now, assume the value of each type of coin is positive. You wonder whether those exchanges cause a contradiction.

Input Format

There is an integer T in the first line, indicating the number of test cases.

Each test case starts with a single integer N, which is the number of exchanges. Each of the following N lines describes an exchange. Each line contains three integers, a_i, b_i, c_i, which means the difference of number of each types of coins after that exchange.

- $1 \leq T \leq 20$
- $1 \leq N \leq 100000$
- $1 \leq |a_i|, |b_i|, |c_i| \leq 100$

Output Format

For each test case, output `"Possible"` if it won't cause a contradiction; otherwise, output `"Rumor"`.

Sample Input

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

Sample Output

```
Rumor
Possible
Possible
Rumor
```

Problem I
Counting Substrings

Time Limit: 3 seconds

Chih-chih and Ning-ning are friends love playing games together. They love strategic and puzzle games like Civilization V, SpaceChem, Portal 2 and Trine 2. However, as they are the smartest students in the National Tomato University, those games are usually too easy for them.

In order to make some fun, Chih-chih and Ning-ning develop their own game called **CSS** (Counting Substrings). At first, the two players negotiate a string S and both of them get a list of strings that contains only an empty string. The two players then take turns to select a number D and a character C. At the i-th turn, both of the players should take the D_i-th string in their own list and concatenate C_i at either the beginning or the end of the string and append the new string to their list (So it becomes the $(i+1)$-th string). After each turn, they will count the number of strings in their own list that is also a substring of S. The one that has more strings as S's substring will win that turn.

For example, the initial game state may looks like this:

S="abcab", list1=[""], list2=[""]

If player 1 select $(1, c)$, the state will looks like:

S="abcab", list1=["", "c"], list2=["", "c"]

If player 2 select $(2, a)$ and only player 1 decides to concatenate the character at the end of string, it becomes:

S="abcab", list1=["", "c", "ca"], list2=["", "c", "ac"]

The game is fun, however, it's too hard that nobody other than Chih-chih and Ning-ning wants to play it. So, Chih-chih and Ning-ning decide to simplify the game. In the new version, the first player always concatenate the character at the end of the string and the second player always concatenate the character at the beginning.

Soonly Chih-chih and Ning-ning find the game is too easy for them. Hence they develop their own AI to play the game. Unfortunately, the speed of counting substrings is too slow that their AIs take hours to play 201410 turns. As you know, Chih-chih and Ning-ning are not familiar with algorithm. The program still takes one minute to run even though they have try their best optimizing the code. As a result, they turn to you, a brilliant ACM-ICPC competitor, and hope to get some helps.

Input Format

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

Each test case starts with one line, containing the string S. The next line contains an integer R, indicating the number of turns going to play. Each of the next R lines contains an integer D_i and a character C_i, which are described above.

- $1 \leq T \leq 10$
- $1 \leq |S| \leq 100000$
- $1 \leq R \leq 201410$
- $1 \leq D_i \leq i$
- C_i is always an English lowercase letter.

Output Format

Output one line for each test case.

For each turn, output '`A`' if the first player win, '`B`' if the second player win and '`T`' if there's a tie.

Sample Input

```
1
abcab
4
1 c
2 a
2 b
4 a
```

Sample Output

```
TATB
```

Problem J
Networking Deployment

Time Limit: 7 seconds

Meow Meow city is full of cat. In the past, cats use computer and playing games. But in today, cats want to use Internet to play massive multiplayer online game (MMOG) such like *Large Polo III*.

By the way, this game is considered illegal by municipal government in Bark Bark city, so some dogs call this game as *Diablo III* on Internet to avoid arrest.

Back to the problem of Meow Meow city. You are a network deploy engineer and are appointed to make each cat house connect to each other either directly or indirectly with minimum cost.

There are N cat houses locate on a 2-dimension plane. You have 2 methods for deploying, as the following.

- Directly connect two cat houses with a wire. The cost will be the square of the Euclidean distance between them. That is, consider that there are 2 house locate at (x_1, y_1) and (x_2, y_2), the cost will be

$$(x_1 - x_2)^2 + (y_1 - y_2)^2$$

- Deploy a signal launcher on a cat house, it could connect to the network of local network provider by wireless. However, there are 2 local network providers and both have their own network. Two cat houses connect to each other only if they are in the same network.

  Besides, because of technical issues, the cost of deployment may be different for each cat house. The pricing of 2 network providers may be different, of course.

You could apply any method as you want to each cat house.

Please notice that you can deploy **both** network provider's launcher on the same cat house. Of course you should pay the cost separately.

Input Format

The first line of input contains a integer T, indicating the number of test cases.

Each test case starts a line with a single integer N, indicating the number of cat houses. The following N lines, describe the launchers' price and location of each cat house. Each line contains a_i, b_i, x_i, y_i, indicating the deploying cost of one of network operator's launcher is a_i, the cost of the other one is b_i and the location is (x_i, y_i).

- $T \leq 180$
- $N \leq 2000$
- $1 \leq a_i, b_i \leq 10^6$
- $|x_i|, |y_i| \leq 500$
- $(x_i, y_i) \neq (x_j, y_j)$ for all $i \neq j$
- At least 66% of test cases, $N \leq 500$.

Output Format

For each test case, only output a line contains minimum cost of optimal deployment that make all cat house could connect each other either directly or indirectly.

Sample Input

```
2
4
1 10000 0 0
10000 1 1 1
1 10000 -10 -10
10000 1 10 10
3
1 10000 10 10
10000 1 -10 -10
1 1 0 0
```

Sample Output

```
6
4
```