Problem 1: Another Day in Rainbowland

Our journey into Rainbowland starts with a serene day at Rainbowland Zoo. Ronny, a tourist from Earth, was first in line to see the rainbrelephants and the rainguins, but when the keepers were getting ready for work, they realized that they were overstaffed! One keeper is going to get the day off, while the rest of them have to feed the rainboraffes and the raingers. The only appropriate way to settle this, of course, is a Rock, Paper, Scissors tournament. Your job is to help Ronny (who is a good adventurer but a terrible programmer) write a program that judges the tournament and determines who wins. They get the day off to tell the citizens of neighboring Thunderlandia how great Rainbowland is while the rest take Ronny to see the raindeer.

Rules of Rock, Paper, Scissors: Played by two players at a time, each player will make a series of “throws”, either Rock, Paper, or Scissors. These throws will be judged head-to-head: Rock beats Scissors, Scissors beats Paper, and Paper beats Rock. The winner of each throw gets a point, and after all the throws are judged, whoever has the most points is the winner of the match. If both players throw the same object, neither receive a point.

Rules of the tournament: The tournament will be a single elimination tournament with a standard bracket structure. This means that each player starts with a label: Player 1, Player 2, Player 3, etc, until Player N. The tournament starts with Player 1 vs. Player 2, Player 3 vs. Player 4, etc, where the winners advance to round 2 and the losers are eliminated. Then, the winner of Player 1 vs Player 2 goes against the winner of Player 3 vs Player 4 and the winner of that match advances to the 3rd round, etc, until there is only one player remaining. That remaining player is the champion! (and gets to relax for the day)

- If there are an odd number of players in a round, the last player advances unopposed. (e.g. if there are 3 players in a round, Player 1 goes against Player 2 and Player 3 automatically advances to the next round)
- If there is a tie in any match, the first player in the list advances. (seniority!!!)

Input Format

The file will contain multiple test cases. The first line of the input file will have a number **C**, representing the number of cases to consider.

Following that will be **C** cases, each starting with a number **N** on its own line representing the number of players in the tournament. Following that will be **N** lines, each containing a string made of the characters “R”, “S”, and “P” (quotes for clarity). The first line represents the throws made by Player 1, the second line is the throws made by Player 2, etc. Each line will have the same number of characters. Note that players will bring their series of throws with them to each new round: If the 2nd round has Player 1 against Player 3, use the same throws you used for Player 1 and Player 3 in the 1st round for their 2nd round match.

C will be at least 1 and no more than 100. **N** will be at least 2 and no more than 64, and each set of throws will be no more than 100 characters.

Output Format

For each test case, print a single line of the format “Player X gets the day off!” (quotes, again, for clarity) where X is replaced by the number of the Player who won the tournament.

Example Input

```
2
2
PPP
SSS
3
R
P
S
```

Example Output

```
Player 2 gets the day off!
Player 3 gets the day off!
```

Problem 2: Racing on the Rainbow Road

One of the most popular pastimes in Rainbowland is yoshi racing. Rainbowland is full of rolling green hills, and all of the Rainbowlanders love to saddle their yoshis and race along the scenic Rainbow Road, an enormous grid of roads that seems to be made out of nothing but rainbows. They nearly always beat Thunderlandia in the yearly race along the border (a dramatic place where the rainbows-fields end at a steep cliff) due to their diligent practice. Ronny, still in Rainbowland, wants to participate. These races tend to be very competitive and the Rainbow Road can be very confusing at times, so he needs your help to make sure his paths are as efficient as possible! Ronny will tell you his paths through the Rainbow Road, and he wants to know if he will ever reach the same intersection twice in a single path. If he is, he needs to change his path, so get going!

Input Format

The file will contain multiple test cases. The first line of the input file will have a number **C**, representing the number of cases to consider.

Following that will be **C** cases, each containing a single string, on its own line, made of only the characters “U”, “D”, “L”, and “R”. (These and all other quotes in the problem set are just for clarity. I’m going to stop saying this now, so I hope you are paying attention!) U for Up, D for Down, L for Left, and R for Right, meaning Ronny moves one rainbow of distance in the given direction.

C will be at least 1 and no more than 100. Each string will contain no more than 1000 characters.

Output Format

For each test case, print a single line with either “I’m the best!” if Ronny does not reach the same intersection twice, or “Waaaaaaaah” (that’s 8 a’s) if Ronny does reach an intersection more than once.

Example Input

3
URDL
UUDDLLRR
UUURR

Example Output

Waaaaaaaah
Waaaaaaaah
I’m the best!

Problem 3: Killing Time

The people of Rainbowland like to stay efficient, and a very important factor in being efficient is everyone having the right time. One less obvious factor in being efficient is quickly resetting your clock after a power outage so you have the right time! Ronny is taking a tour of Rainbowland's clock-resetting control center, and is curious how the rainbow-resetters find the quickest way to set Rainbowland's clocks from the wrong time to the right time. The rainbow-resetters tell him that they find the minimum number of button presses required to switch from one time to another. Ronnie needs your help checking his work!

This clock has 5 buttons, and we are measuring speed here by the number of button presses required to get from one time to the other. The five buttons do the following:

- Add one minute
- Subtract one minute
- Add one hour
- Subtract one hour
- Switch from A.M. to P.M., or from P.M. to A.M.

It's important to notice that if the time is 8:59 A.M. and you add one minute, you do **not** go to 8:00 A.M., but instead to 9:00 A.M. -- the minute and the hour are not separated. Additionally, the day starts with 12:00 A.M., goes to 11:59 A.M., then come noon, or 12:00 P.M., then goes to 11:59 P.M. You are not responsible for the date, only the time.

Input Format

The file will contain multiple test cases. The first line of the input file will have a number **C**, representing the number of cases to consider.

Following that will be **C** cases, each containing a single string, on its own line, containing nothing but two times: the current time on Curious George's clock, and the true time, which he wishes to set his clock to. Each time will be of the format "**XX:XX Y.M.**" where the **X**s can be any digit (they are not all the same!) as long as the two hour digits make a number between 1 and 12, and the two minute digits make a number between 0 and 59. If the number is a single digit, there will be a leading zero. **Y** will be either "A" or "P".

C will be at least 1 and no more than 100.

Output Format

For each test case, print a single line consisting of just a single number: the smallest number of button presses required to turn the first time into the second time.

Example Input

2
01:14 A.M. 11:04 A.M.
09:37 A.M. 02:43 P.M.

Example Output

13
11

Problem 4: KILLING TIME!

Rainbowland had been a peaceful land for some time, but Thunderlandia has finally gotten sick of the Rainbow King's condescending attitude toward Thunderlandia, and suddenly there was an invasion! Thunderbolts and Thunderbirds flooded into Rainbowland, led by Thunderlandia's greatest general, Dr. Darkcloud. Ronny has been given the gravely important task of escorting Rainba, the Rainbow Princess, to safety! Dr. Darkcloud knows that the morale of the Rainbowland armies would be completely broken if she was hurt, so he has sent an army to capture her! Ronny must get Rainba to the safehouse without being caught. Given the position on Dr. Darkcloud's army, you need to tell Ronny and Rainba if they have a safe path from their current location to their safehouse.

The Thunderlandian army will capture Ronny or Rainba if they are ever within one rainbow (remember, this is the national unit of distance!) of a Thunderlandian soldier.

Input Format

The file will contain multiple test cases. The first line of the input file will have a number **C**, representing the number of cases to consider.

Following that will be **C** cases, each starting with the line "**X Y**", representing the number of rows and the number of columns on the grid which represents the current area Chelsea and Ronny are in.

Next will be **Y** rows, each containing a string of **X** characters, each of which will be one of "#", "R", "A", "_", "T", or "S".

- "#" means this location is unaccessible.
- "_" means this location is available (and safe)
- "R" is Ronny's starting location.
- "A" is Rainba's starting location.
    - Ronny and Rainba can move up, down, left, and right, one rainbow at a time, but not on to any location marked with a "#". They may not move on to any square located outside the grid.
- "T" is a member of the Thunderlandian Army.
    - A Thunderlandian Army member will capture Ronny or Rainba if either of them are in a square above, below, or to either side of the Army member. If they start next to Ronny and/or Rainba, they immediately capture whoever they are next to.
- "S" is the location of the safehouse.
    - If the safehouse is one square away from an Army member, Ronny and

Rainba can still safely make it inside.

C will be at least 1 and no more than 100.
X and **Y** will be at least 3 and no more than 25.
There will be exactly one “R”, “C”, and “S” in every case.

Output Format

For each test case, print, on its own line, one of the following strings:

- “Aw, here it goes!” if neither Ronny nor Rainba can make it to the safehouse without being captured.
- “Ronny, no!” if Ronny cannot make it to the safehouse without being captured, but Rainba can.
- “Rainba, no!” if Rainba cannot make it to the safehouse without being captured, but Ronny can.
- “Huzzah!” if both Rainba and Ronny can make it to the safehouse without being captured.

Example Input

```
1
4 4
RA__
____
TT__
TT_S
```

Example Output

```
Huzzah!
```

Problem 5: Rainbowland War Tactics

The Rainbowland General is leading her armies to battle, and she's looking for a spot to place the last of her very valuable, very powerful tanks. She's mapped the main battlefield as an 8x8 grid, but her aide has already moved 7 tanks onto the grid without her permission! Your job is to tell her that she can send the 8th tank out there if there's a good position for it, and otherwise to recall the tanks for repositioning.

Tanks can fire their rainbow rockets as far as they want horizontally, vertically, or diagonally at a 45 degree angle, but they don't have time to rotate in battle. None of the tanks should be able to accidentally shoot each other (they're very expensive!) if an enemy unit moves out of the way.

Your job is to tell the general if the 7 currently placed tanks are in good positions, and if there is a good position for the 8th tank.

Input Format

The file will contain multiple test cases. The first line of the input file will have a number **C**, representing the number of cases to consider.

Following that will be **C** cases, each containing 7 lines of the form (**X,Y**). These are the locations of the first 7 tanks on the battlefield.

C will be at least 1 and no more than 100. **X** and **Y** will be integers between 0 and 7, inclusive.

Output Format

For each test case, print a single line, containing either "Evacuate immediately!" if two of the 7 tanks could potentially shoot each other, or if there is no good position for the 8th tank, or "(**A,B**)", where the point (**A,B**) is a good position for the 8th tank and none of the first 7 tanks are able to shoot each other.

Example Input

2
(0,0)
(1,1)
(2,2)
(3,3)
(4,4)
(5,5)
(6,6)
(0,6)
(1,4)
(2,2)
(3,0)
(4,5)
(5,7)
(6,1)

Example Output

Evacuate immediately!
(7,3)

Problem 6: The Great Thunderwall

The Rainbowlandian forces are badly outnumbered, and their chances look grim. Thunderclouds are invading the skies, throwing a shadow on the Rainbow Road and gloom into the lofty halls of the Crystal Castle!

Ronny and Rainba (safely in hiding) have realized that Rainbowland's only hope is to disable Thunderlandia's digital command center, which has many different layers of security (thunderwalls) built in. (Neither of them is a very good hacker, but happily the Thunderlandians aren't much better!)

While Ronny codes, Rainba worries about the time he wastes writing attacks that will take too long to execute. She needs your help to determine how much data Ronny can process in time!

Ronny's attacks perform a number of operations determined by the formula $\mathbf{N}^{\mathbf{X}} + \log_2(\mathbf{N}^{\mathbf{Y}}) + \mathbf{N}^{(1/\mathbf{Z})}$, where **N** represents the amount of data Ronny has to process before he expects to have defeated this security feature, and Ronny's computer can perform **S** operations per second. Rainba doesn't want the attack to take more than **T** seconds. Your job is to determine the biggest value of **N** such that the attack will complete in **T** seconds or less.

P.S. Rainba doesn't like dealing with decimals, so you should round each term of the time formula down to the nearest integer.

Input Format

The file will contain multiple test cases. The first line of the input file will have a number **C**, representing the number of cases to consider.

Following that will be **C** cases, each on a single line, containing 5 integers in the form "**X Y Z S T**", as described above.

C will be at least 1 and no more than 100.

X, Y, Z will be between 0 and 1,000, inclusive, with the exception that **Z** will never be 0.

S will be between 1 and 1,000,000, inclusive.

T will be between 1 and 10^{10}, inclusive.

Output Format

For each test case, print a single line containing only the biggest possible value of **N** as defined above. If **N** would be above 10^{15}, instead print "Infinity!"

Example Input

```
2
1 1 1 1 1000
0 0 2 1 1000
```

Example Output

```
496
999999
```

Problem 7: The Other Great Thunderwall

Ronny (the tourist-turned-special agent) and Rainba (the adventurer-princess) are scaling the icy walls of the Thunder Keep, hoping to ambush the Thunder King and persuade him to withdraw from Rainbowland (or to rainbow-kill him). The wall, like everything else in Thunderland, is strange. It consists of a series of adjacent columns connected in certain places. Whenever one of our heroes reaches a connection, they will immediately cross it. They have noticed that no places on the wall have two connections at the same height (see Nonexample below). The heroes know which column they want to start at (namely the one without spikes on the bottom), but before they start climbing they need to know the column that they'll be at once they reach the top (hopefully one near a window).

```
Nonexample:
|   |   |
|===|===|
|   |   |

Example 1: N = 3, a 3 column wall
|   |   |
|===|   |
|   |   |
|   |===|
|   |===|
|===|   |
|   |   |

Example 2: N = 5, a 5 column wall
|   |   |   |   |
|===|   |===|   |
|   |===|   |   |
|===|   |   |===|
|   |   |===|   |
|   |===|   |   |
|   |   |   |===|
|===|   |===|   |
|   |   |===|   |
```

| | | | |

Input Format

The file will contain multiple test cases. The first line of the input file will have a number **C**, representing the number of cases to consider.

Following that will be **C** cases, each starting with the integer **N** on its own line.

Following that will be **N-1** lines, each of them form "**K** $\mathbf{H_1}$ $\mathbf{H_2}$ … $\mathbf{H_K}$", where **K** and all $\mathbf{H_x}$ are integers. On the **i**th of these lines are the connections between columns **i** and **i**+1. For example, the first line of the given form will be the heights of the connections between columns 1 and 2.

Following that will be a line containing a single integer **S**, the starting column (1-indexed, from left to right).

C will be at least 1 and no more than 100.

N will be between 2 and 100, inclusive.

K will be between 0 and 100, inclusive.

Each $\mathbf{H_x}$ will be between 1 and 1000, inclusive. Note: The $\mathbf{H_x}$s are not sorted.

S will be between 1 and **N**, inclusive.

There will be no more than 1 connection at a certain height for any particular column.

Output Format

For each test case, print a single integer on its own line, representing the column that Ronny and Rainba will be at when they reach the top.

Example Input

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

Example Output

```
1
```

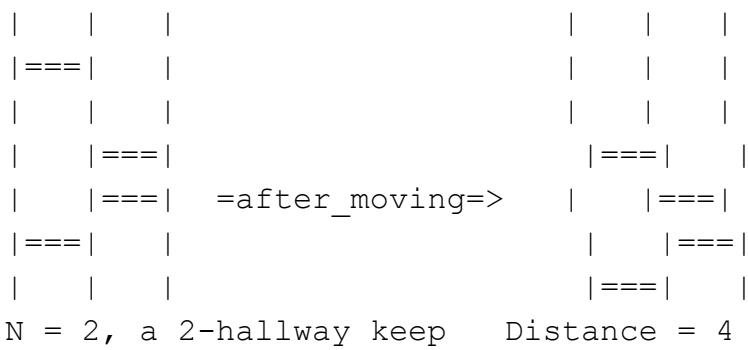
Problem 8: It’s a Trap!

Ronny and Rainba made it to the window, only to discover that it allowed access only to a chamber with a domed glass ceiling (affording a stunning view of the everlasting storm above!). N hallways started in the far wall. They had entered the Thunder Keep!

Upon closer inspection, Rainba announced that each of the hallways was lined with traps. Similar traps were used in the Crystal Castle in Rainbowland. Each one could be moved (from a secret control panel she found a few minutes later-- apparently the same architect had designed both the Crystal Castle and the Thunder Keep and didn’t like varying his trap designs), but only toward the room in which Ronny and Rainba were standing. Also, no trap could be moved past a trap closer to the starting room in an adjacent hallway without setting them all off. Ronny found two pairs of dusty super-jump boots in a corner.

Ronny and Rainba need to know if the super-jump boots can get them past all the traps if they are all moved as close to the starting room as possible, and how far they will have to travel with these terribly dusty boots on. Your job is to tell them the distance of the farthest trap down any of the hallways so that they know if they can jump over it.

Each trap is one lightning bolt wide. It can’t be moved past any traps in adjacent hallways. For example, if there was a trap between hallways 1 and 2 at a distance of 5lb, and a trap between hallways 2 and 3 at 3lb, then the first trap can be moved as close as 2lb, and the second to 1lb.



Input format

The file will contain multiple test cases. The first line of the input file will have a number **C**, representing the number of cases to consider.

Following that will be **C** cases, each starting with the integer **N** on its own line.

Following that will be **N** lines, each of them form "**K** $\mathbf{D_1}$ $\mathbf{D_2}$ … $\mathbf{D_K}$", where **K** and all $\mathbf{D_x}$ are integers. On the **i**th of these lines are the traps in hallway **i**. For example, the first line of the given form will be the traps in hallway 1.

C will be at least 1 and no more than 100.

N will be between 1 and 100, inclusive.

K will be between 0 and 100, inclusive.

Each $\mathbf{D_X}$ will be between 1 and 10000, inclusive. Note: The $\mathbf{D_X}$s are not sorted.

There will be no more than 1 lightning trap at a certain distance for any particular pair of adjacent hallways.

Output format:

For each test case, print a single integer, on its own line, representing the distance of the farthest trap after Ronny and Rainba have moved them as close as they can.

Example Input

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

Example Output

```
4
6
```

Problem 9: Persuasion Showdown

Ronny and Rainba have finally reached the throne room of the Thunder Keep; a long, high room containing lots of gloomy tapestries and an angry king. Rainba knows that the fate of her people rests on her colorfully-clad shoulders-- she has to convince the Thunder King, somehow, to leave Rainbowland alone. As heiress to the throne, she has had extensive diplomatic and rhetorical training. Unfortunately, though, she has too many ideas!

Given her extensive list of things to say, your job is to pick out what she and Ronny will end up saying. Rainba will end up saying whatever her longest idea is, of course, since that's most impressive, and she will tell Ronny to say whatever the second longest one is. In case there are two or more ideas that are tied for longest or second longest, Rainba and Ronny prefer ideas that came first in Rainba's list.

Input Format

The file will contain multiple test cases. The first line of the input file will have a number **C**, representing the number of cases to consider.

Following that will be **C** cases, each starting with the number **N** on a single line, representing the number of ideas Rainba has.

Following that will be **N** strings, each on their own line, each being a single idea of Rainba's.

C will be at least 1 and no more than 100.

N will be at least 2 and no more than 1,000.

Each idea will be no more than 100 characters.

Output Format

For each test case, print two lines. The first line will be of the form "Rainba: **X**" and the second line will be of the form "Ronny: **Y**" where **X** and **Y** are the two ideas chosen as described above.

Example Input

1
4
I am terrible
We are the greatest leaders alive!!!!
You should let us do whatever we want
I rather be watching TV than this!!!!

Example Output

Rainba: We are the greatest leaders alive!!!!
Ronny: You should let us do whatever we want

Problem 10: Party Time

The Rainbowlandians have won the war! To celebrate their victory, a huge party will be thrown. At this party, there will be thundersteed, thunderbison, and lightninglamb served to eat. Because of the war, resources are scarce, so there are no guarantees on how many servings of animal will be available for the party. Rainbow Roy, the master party planner of Rainbowland, is concerned that because of the food shortage not everyone will be able to get a meal of their preference. Help Roy figure out if he can accommodate each guest with one of their preferred meals.

You are given a list of the meal preferences of each party guest. Each guest can have a preference for any combination of the three animals, and they'll always have a preference for at least one. You are also given the number of servings of each type of animal. Determine if there is a way to distribute the servings of each animal such that each party attendee gets a meal of one of the animals they have a preference for.

Input Format

The input will begin with **T**, the number of test cases. Each test case will begin with a line with four integers on it: **A**, the number of servings of thundersteed, **B**, the number of servings of thunderbison, **C**, the number of servings of lightninglamb, and **P**, the number of party attendees. Following this will be **P** lines, each containing a set of characters denoting the animal preferences of a party attendee. The characters will either be "A" for thundersteed, "B" for thunderbison, or "C" for lightninglamb. Each guest's preference list will always be sorted in alphabetical order, and no preference will appear more than once.

T will be at least 1 and at most 100.

A, **B**, and **C** will be at least 0 and at most 1000.

P will be at least 1 and at most 1000.

Output Format

For each test case, output "Time to party!" if every party attendee will get a meal of an animal they prefer, or "Super bummer!" if not every attendee can get an animal they prefer.

Example Input

```
3
4 0 0 3
A
B
C
1 1 1 3
A
AB
C
4 0 1 5
ABC
AB
AB
A
A
```

Example Output

```
Super bummer!
Time to party!
Time to party!
```