# A. Dexter n Dee Dee

Time Limit – 1 seconds

Dexter is Dee Dee's little brother. She loves "playing with" him but she destroys things in his lab and drives him crazy. It may seem that Dee Dee is nothing but a pure idiot who just annoys Dexter all the time, but truthfully deep down of her candy canes and lollipop glitter heart, she loves Dexter and thinks he's awesome. In fact, to her, Dexter is her favorite thing in the whole entire world.

Today, Dexter is experimenting on a tree. It is a rooted tree consisting of n nodes numbered from 1 to n. The root of the tree is one. Each node has a color, white or black. Now he is trying to find how many nodes are there who has at least one black node as sibling. Two nodes, a and b, are called siblings, if they share the same parent. The color of the root is always white. As it is bit hard for him to count that thing, he asked Dee Dee to help, but she could not help him in this task either. So, today Dexter needs your help.

**Input :**

Input starts with an integer T (≤ 100), denoting the number of test cases. Each case starts with an integer n (2 <= n <= 100), the number of nodes in the tree. Each of the next n - 1 lines will contain three integers $U_i$, $P_i$ and $C_i$, where $P_i$ denotes the parent of $U_i$ and $C_i$ denotes the color of $U_i$. If $C_i$ is 0, the node is white. If $C_i$ is 1, the node is black.

**Output:**

For each case, print the case number and the expected answer of the problem described. See the output format below.

| Sample Input | Sample Output |
|---|---|
| 2<br>3<br>2 1 1<br>3 1 0<br>3<br>2 1 1<br>3 1 1 | Case 1: 1<br>Case 2: 2 |

# B. The Saint Martin Tragedy

Time Limit – 0.5 seconds

Recently a sad incident occurred in AUST. On the 14th of April, 2014 some students of AUST CSE 25th batch went to enjoy their completion of graduation. They went to the Saint – Martin Island. Who would have thought that some of them are never going to back! The low tide taken away all the pride and hopes of six families. 6 brilliant minds were lost in that dreadful incident. We hope such incident would never to any more. It was an outsize tragedy for the families who will have no solace for the sudden loss of their near ones who were about to become graduate engineers. This problem is given as a tribute to them.

In this problem you will be given only six names who are not with us anymore. You have to print the below lines when these names appeared as input –

**"bappi" - a great athlete**

**"sabbir" - a great photographer**

**"noman" - the style icon**

**"uday" - an extra ordinary person**

**"evan" - a kind and helpful person**

**"ankur" - the best player**

## Input / Output:

Input starts with an integer **T (≤ 6)**, denoting the number of test cases.

Each case begins with a non-empty string as described above. Output the lines as descried above in the problem statement. Print a line **"we will never forget you"** at the end of the test case output without quotes. See the sample I/O format.

| Sample Input | Sample Output |
|---|---|
| 3<br>bappi<br>sabbir<br>uday | a great athlete<br>a great photographer<br>an extra ordinary person<br>we will never forget you |

# C. Difficult Job

Time Limit – 1 seconds

Kodu has given up on the difficult coach job and switched to food tasting instead. Having skipped breakfast like a professional connoisseur, he is visiting a cured meat festival. The most renowned cook at the festival, Monu Mia, has prepared N equal hamburgers which need to be distributed to M tasters such that each taster gets a precisely equal amount. He will use his trusted knife to cut them into pieces.

In order to elegantly divide the hamburgers, the number of cuts splitting individual hamburgers must be as small as possible. For instance, if there are two hamburgers and six tasters (the first test case below), it is sufficient to split each sausage into three equal parts, making a total of four cuts. On the other hand, if there are three hamburgers and four tasters (the second test case below), one possibility is cutting off three quarters of each sausage. Those larger parts will each go to one of the tasters, while the fourth taster will get the three smaller pieces (quarters) left over.

Kodu wants to try the famous hamburgers, so he volunteered to help Monu Mia. Help them calculate the minimum total number of cuts needed to carry out the desired division.

**Input :**

First line of input contains an integer **T (1 ≤ T ≤ 100)**, number of test cases to follow.

The first and only line of each test case contain two positive integers, **N** and **M (1 ≤ N, M ≤ 1000)**, the number of sausages and tasters, respectively.

**Output:**

For each test case, print the test case number and the required minimum number of cuts.

| Sample Input | Sample Output |
|---|---|
| 3<br><br>2 6<br><br>3 4<br><br>6 2 | Case 1: 4<br>Case 2: 3<br>Case 3: 0 |

# D. Thinking Rational

Time Limit – 1.5 seconds

In mathematics, a rational number is any number that can be expressed as the quotient or fraction **P/Q** of two integers, with the denominator Q not equal to zero. Since Q may be equal to 1, every integer is a rational number. A rational number can have many representatives.

In this problem, you are given a rational number in the form of **P/Q** and two other numbers, **A** and **B**. You have to find how many representatives of **P/Q** are there where for some

$$X/Y = P/Q$$

$$\text{where, } -A < X < +A \text{ and } -B < Y < +B$$

**Input :**

Input starts with an integer T (≤ 1000), denoting the number of test cases.

Each case contains a line of four integers P, Q, A, B.

Both P and Q are integer numbers ( $-10^8 <= P, Q <= 10^8$ and $P != 0$ and $Q != 0$ ) and both A and B are positive integers ( $1 <= A, B <= 10^8$ )

**Output:**

For each case, print the case number and the expected answer. See the sample I/O format.

| Sample Input | Sample Output |
|---|---|
| 4<br><br>1 2 3 4<br><br>4 3 2 1<br><br>4 4 4 4<br><br>6 3 6 3 | Case 1: 2<br>Case 2: 0<br>Case 3: 6<br>Case 4: 4 |

# E. Devil's Triangle !

Time Limit – 2 seconds

The **Bermuda Triangle**, also known as the **Devil's Triangle**. It is a region of the northwestern Atlantic Ocean in which a number of aircraft and surface vessels have disappeared. Some people have claimed that these disappearances fall beyond the boundaries of human error or acts of nature. Some of these disappearances have been attributed to the paranormal, a suspension of the laws of physics, or activity by extraterrestrial beings by popular culture. Though a substantial documentation exists showing numerous incidents to have been inaccurately reported or embellished by later authors, and numerous official agencies have gone on record as stating the number and nature of disappearances to be similar to any other area of ocean, many have remained unexplained despite considerable investigation.

We are interested in drawing some devil triangle. You will be given two numbers **M** and **N**. **N** denotes the number of levels and **M** denotes the size. That means you have to draw a devil's triangle which should have **N** levels, the first level will contain one triangular room.

Now, for triangle a devil's room of size **M**, first you have to put

**/\ (forward slash and a backward slash)**

***N.B:** ASCII value of forward slash and a backward slash are 47 and 92 respectively.

Then for each **i-th** row (**i = 2 to M-1**), you have to put a **'/'** (forward slash) and **2*(i-1)** spaces and after that a **'\'** (backward slash). The **M-th** row will contain a **'/'**, **2*(M-1) '_'**(underscore) and a **'\'**. And add spaces to the previous parts so that the overall shape looks fully triangular. Suppose, **M = 4**. Then a room will be like:

```
...\
../..\
./....\
/______\
```

(Here spaces are shown as dots)

Now, for devils building all the **N** levels, for each **i-th** level (**i = 2 to N**), you have to build **i** devil's triangular rooms, each will start from the base vertices of the previous triangular rooms. (Note that the first level will contain one room only). If **N = 1** and **M = 2**. Then the museum will be like:

```
./\
/__\
```

Now given M and N. You have to draw the Devil's Triangle.

**Input :**

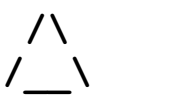
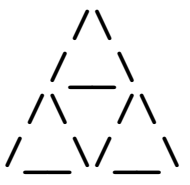
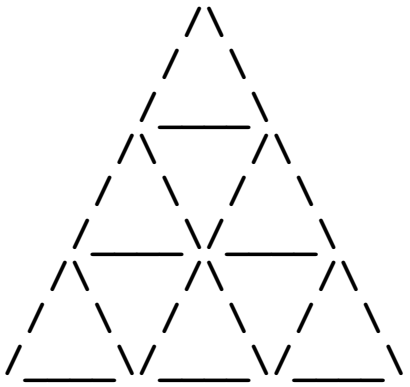
Input starts with an integer **T (≤ 100)**, denoting the number of test cases.

For each test case a line contains a pair of integers *M* and *N*

Constraints: (2 ≤ **M** ≤ 25) and (1 ≤ N ≤ 25).

**Output:**

For each case, print the case number in a single line. Then draw The Devil's Triangle according to the description. Finally print a blank line. See the output format.

| Sample Input | Sample Output |
|---|---|
| 4<br>2 1<br>2 2<br>3 3<br>3 1 | Case 1:<br>  <br>Case 2:<br>  <br>Case 3:<br>  <br>Case 4:<br>  |

# F. Distinct Sequence

Time Limit – 2 seconds

Mr. Shakil Ahmed got a set of intervals for his birthday. There are many games he can play with them. In one of them , Shakil must find the longest sequence of distinct intervals such that each interval in the sequence is in the set and that each interval contains the one that follows in the sequence.

Write a program which finds one such longest sequence.

**Input :**

The first line of input contains a integer T ( <= 15) which denotes the number of test case and next T lines the integer N ($1 \leq N \leq 100\,000$) , the number of intervals in the set. Each of the following N lines contains two integers A and B describing one interval.

Constraints: $1 \leq A < B \leq 1\,000\,000$

**Output:**

For each case, print the case number followed by the length K (the longest sequence) on the first line. Each of the following K lines should contain one element of the sequence, an interval in the same format it was given in the input.

| Sample Input | Sample Output |
|---|---|
| 3<br>3<br>3 4<br>2 5<br>1 6<br>5<br>10 30<br>20 40<br>30 50<br>10 60<br>30 40<br>6<br>1 4<br>1 5<br>1 6<br>1 7<br>2 5<br>3 5 | Case 1: 3<br>1 6<br>2 5<br>3 4<br>Case 2: 3<br>10 60<br>30 50<br>30 40<br>Case 3: 5<br>1 7<br>1 6<br>1 5<br>2 5<br>3 5 |

# G. Evil Emperor

Time Limit – 1 seconds

The evil emperor Cactus has in his possession the Magic Keg and has flooded the Enchanted Forest! The Painter and the three little hedgehogs now have to return to the Beaver's den where they will be safe from the water as quickly as possible!

The map of the Enchanted Forest consists of **R** rows and **C** columns. Empty fields are represented by **'.'** characters, flooded fields by **'*'** and rocks by **'X'**. Additionally, the Beaver's den is represented by '**D'** and the Painter and the three little hedgehogs are shown as **'S'**.

Every minute the Painter and the three little hedgehogs can move to **4** neighboring fields (up, down, left or right). Every minute the flood expands as well so that all empty fields that have at least one common side with a flooded field become flooded as well. Neither water nor the Painter and the three little hedgehogs can pass through rocks. Naturally, the Painter and the three little hedgehogs cannot pass through flooded fields, and water cannot flood the Beaver's den.

Write a program that will, given a map of the Enchanted Forest, output the shortest time needed for the Painter and the three little hedgehogs to safely reach the Beaver's den.

Note: The Painter and the three little hedgehogs cannot move into a field that is about to be flooded (in the same minute).

**Input :**

First line of input contains an integer **T (1<=T<=15)**, the number of test cases to follow.

The first line of each test case will contain two integers, **R** and **C**, **(1<=R, C<=50)**. The following **R** lines will each contain **C** characters ('.', '*', 'X', 'D' or 'S'). The map will contain exactly one 'D' character and exactly one 'S' character.

**Output:**

For each test case, print the test case number and the shortest possible time needed for the Painter and the three little hedgehogs to safely reach the Beaver's den. If this is impossible output the word "**Go to Brazil**".

| Sample Input | Sample Output |
|---|---|
| 3<br>3 3<br>D.*<br>...<br>.S.<br>3 3<br>D.*<br>...<br>..S<br>3 6<br>D...*.<br>.X.X..<br>....S. | Case 1: 3<br>Case 2: Go to Brazil<br>Case 3: 6 |

Explanation of the second sample test case: The best they can do is to go along the lower border and

Then the left border, and get flooded one minute before reaching the den.

# H. Omega

Time Limit – 2 seconds

The function Omega (n) counts the number of distinct prime factors of n where n is a positive integer. So

Omega (10) = 2

Omega (20) = 2

Omega (1) = 0

In this problem, you will be given three integers a, b and k. You will have to find out how many x are there where a <= x <= b and Omega (x) = k

**Input :**

Input starts with an integer T (≤ 100000), denoting the number of test cases. Each case contains three integers a, b and k. Here 1 <= a <= b <= 10000 and 0 <= k <= 10000

**Output:**

For each case, print the case number and the expected answer of the problem described. See the output format below.

| Sample Input | Sample Output |
|---|---|
| 2<br>1 1 0<br>20 20 2 | Case 1: 1<br>Case 2: 1 |

# I. Super Hashing

Time Limit – 2 seconds

Hash functions are primarily used to generate fixed-length output data that acts as a shortened reference to the original data. This is useful when the original data is too cumbersome to use in its entirety. One practical use is a data structure called a hash table where the data is stored associatively. Searching linearly for a person's name in a list becomes cumbersome as the length of the list increases, but the hashed value can be used to store a reference to the original data and retrieve constant time (barring collisions).

It involves mapping an element into a numerical value using some mathematical function. In this problem we will consider a simple hash system. It involves assigning numerical value to the alphabets and summing these values of the characters.

For example, the string **"acm"** is mapped to 1 + 3 + 13 = 17. Unfortunately, this method does not give one-to-one mapping. The string "adl" also maps to 17 (1 + 4 + 12). This is called collision.

In this problem you will have to find the number of strings of length **L,** which maps to an integer **S,** using the above hash function. You have to consider strings that have only lowercase letters and uppercase letters in strictly ascending order.

***NB:** all lowercase letters are smaller than all uppercase letters: Example: 'z' < 'A'

Suppose **L** = 3 and **S** = 10, there are 4 such strings: **( abg , acf , ade , bce )**

**agb** also produces 10 but the letters are not strictly in ascending order.

**bh** also produces 10 but it has 2 letters.

**Input :**

The first line of input contains a integer T ( <= 100) which denotes the number of test case. Each case consists of 2 integers **L and S**. **(0 < L , S < 100 000 )**.

**Output:**

For each case, print the case number followed by the expected answer.

| Sample Input | Sample Output |
|---|---|
| 3<br>2 10<br>27 100<br>5 19 | Case 1: 4<br>Case 2: 0<br>Case 3: 5 |

# J. Tom & Jerry

Time Limit – 2 seconds

You all are very familiar with Tom & Jerry. Tom & Jerry always fight with each other. Today, Tom wants to date a beautiful Kitty. So, Tom begs Jerry not to disturb him. But, Jerry wasn't going to listen his request. Tom told Jerry that he would do anything for that. Then, Jerry proposed a hard task for Tom.

Jerry loves sums of the interval. He gives Tom a pair of integers $l$ and $r$ ( $l \leq r$ ). Tom has to find the number of such integers $x$ ( $l \leq x \leq r$ ) , that the first digit of integer $x$ equals the last one (in decimal number system). For example, such numbers as 22,101, 4774 or 9 will be included in the answer and 41, 365 or 1120 will not.

As Tom it not very good at mathematics, he is seeking your help. He knows you are one of the best programmers of AUST. Help him and count described numbers $x$ for a given pair $l$ and $r$.

**Input :**

Input starts with an integer **T (≤ 2000)**, denoting the number of test cases.

For each test case a line contains a pair of integers $l$ and $r$ ($1 \leq l \leq r \leq 10^{18}$)

**Output:**

For each case, print the case number and the expected answer of the problem described. See the output format.

| Sample Input | Sample Output |
|---|---|
| 5<br>10 99<br>10 20<br>100 200<br>6 1000<br>50 5050050 | Case 1: 9<br>Case 2: 1<br>Case 3: 10<br>Case 4: 103<br>Case 5: 505000 |

Explanation : In the 2nd test case : between 10 to 20 there are ten numbers : 10 , 11 , 12,13,14,15,16,17,18,19,20 . But only 11 has the first and last digit same. So the answer is 1.