

Problemset

DIU Take Off Programming

Contest, Fall 2017

A. Take off right now!!!

Time Limit: 1 second

Memory Limit: 64 MB

Competitive programming is a mind sport usually held over the internet or a local network, involving participants trying to program according to provided specification. Solving a problem is very much fun. Daffodil International University has a Competitive Programming and Problem Solving Community. The community organizes classes to train students those who are interested in problem-solving. As a first lesson, the community gives a task to students to write a program which will print "Hello World!". But this time we want you to keep something in mind before you start your programming career. Assume that you are starting programming from now and you are given a task to print "Every accomplishment starts with a decision to try." without quotes. For now, you just have to write the following code exactly and just submit.

```
#include <stdio.h>

int main()
{
    printf("Every accomplishment starts with a decision to try.\n");
    return 0;
}
```

Input

There is no input in this problem. Only write the code exactly and submit.

Output

Every accomplishment starts with a decision to try.

Sample Input

/* There is no input for this problem */

Sample Output

Every accomplishment starts with a decision to try.

Note: Be careful of "new line" at the end of output and spaces between words!

Problem Setter: Unknown

Original Idea: Unknown

B. I have a long way to go

Time Limit: 1 second

Memory Limit: 32 MB

Iqbal has bought a new bicycle to go to his University. He rides his bicycle v meters per second. Every day he wakes up in the morning and rushes to his University having T seconds in hand.

Now, he's thinking of making an android app that tells how much distance he can cover having T seconds in hand with v velocity.

As you know the formula:- Distance (S) = Velocity (v) * Time(T), can you write the code to help him with that?

Input

There will be only two integers in the input v and T .

Output

Print a single integer S that is the distance he can cover with v velocity and T time. You can safely assume that the input and output will fit in an integer data type.

Sample Input	Sample Output
5 3	15
2 18	36

Problem Setter: Md. Ferdouse Ahmed Foysal

Special Thanks: Mahmud Sajjad Abeer

C. Hobe ki?

Time Limit: 1 second

Memory Limit: 64 MB

Jim Moriarty is a student of CSE. His respected teacher Habibi assigned him a task. He has to print some important documents containing N pages for him. Jim decided not to waste money and print the papers from the department office.

So, he went to the office and asked Kawsar vaiya, how many papers do you have? Print hobe ki? He said he has K papers.

Can you tell Jim if it's possible to print the N pages with K papers?

Input

There will be a single line containing two integers N and K . The input will be such that N and K will be within 1 to 100.

Output

If Raju printed everything perfectly and didn't print on any faulty papers your output will be "Yes" (without the quotations) otherwise your output will be "No" (without the quotations).

Sample Input

10 10

50 49

Sample Output

Yes

No

Problem Setter: Aquibuzzaman Sayem

Special Thanks: Mahmud Sajjad Abeer

D. Contest Reminder

Time Limit: 1 second

Memory Limit: 64 MB

The setter of this problem updates the CodeForces Round (Div 2) contest reminder on Facebook. There he uses 24 hours time format to share the starting time of the contest. He doesn't like 12 hours time format as hearing the same Time twice a day seems weird to him as nobody says AM or PM when they speak.

But you are his complete opposite, you don't like watching 24 hours time format. So, you are going to program your phone. Whenever the setter is going to share the CodeForces contest reminder on *DIU ACM Solver Forum's* fb group, in 24 hours format, your phone will convert and show you the time in 12 hours format only. Now, you just have to write a program that converts 24hours time format to 12 hours time format. Just to let you know that, 24 hours time starts at 00:00 and ends at 23:59 whereas, 12 hours time starts at 12:00 AM and ends at 11:59 PM.

Input

The first line will contain an integer T ($1 \leq T \leq 1440$). Then there will be T lines, each containing a Time in the format "H:M". Here, H ($0 \leq H \leq 23$), an integer, denoting the hour in 24 hours format, then a colon(:) with no consecutive spaces, then M ($0 \leq M \leq 59$), another integer, denoting the minutes.

Output

You have to output T lines each containing the format- "Case X : H:M FF", where, X is the case number starting from 1, H meaning the hour in 12 hours format, then a single colon with no consecutive spaces, then M meaning the minutes, then a space and FF should be replaced with AM(Ante Meridiem) or PM(Post Meridiem) on capital letter. Check out the samples to understand the output format properly.

Sample Input

2
20:35
10:55

Sample Output

Case 1: 8:35 PM
Case 2: 10:55 AM

Note: Be careful about the `input and output formatting`.

Problem Setter: Mahmud Sajjad Abeer

E. Spider-Man DhakaComing

Time Limit: 1 second

Memory Limit: 512 MB

Marvel is here at Dhaka to shoot their upcoming movie of Spider-Man series "Spider-Man DhakaComing!" It's almost time to start the shoot but where is Peter Parker aka Spiderman??! He is missing!! But I'll tell you the truth; he is right now at Star Cineplex, watching DC's Justice League secretly. Honestly, who wants to miss a movie in which *Batman* is present?

Now that the movie has finished, watching at his MI band 2, Spiderman just had a little heart attack; there are only a few minutes left for his shooting to start. So, he rushed to the bus stand as fast as he could. There were many buses in front of him. He has to choose the bus which will *leave* the bus stand *first*.

There are N buses with capacity c_i where i is the bus number. There is a queue, and Peter Parker is standing in M th position in that queue! After every passenger standing in front of Peter gets on any bus or leaves the queue, each bus has p_i passengers on it. Whenever a Bus is full, it will leave the bus stand for its destination and lesser the remaining capacity a bus has, better the chances of leaving it has. If two buses have the same remaining capacity, Peter will choose the bus which comes first on the bus line!

As Peter is not much good with these complex math, he asked for your help. Now, given the aforementioned information, you are here to help Peter to find out the best ride.

Input

The first line contains one integer T ($1 \leq T \leq 100$), represents number of test cases. Each case will start with two numbers N ($1 \leq N \leq 100$) and M ($1 \leq M \leq 1000$) — the number of Buses and Peter's position in the queue. Then N lines follow, each line i containing a pair of numbers C_i ($1 \leq C_i \leq 100$) and P_i ($1 \leq P_i \leq C_i$) — the total capacity of i th bus and the current number of passengers of i th bus.

Output

Print as the following — "Case X: Bus number Y", where X is test case number and Y is the serial no of Bus on which Peter should get into. Or else if he is still in the queue waiting for a new bus to come, print — "Case X: He is in the queue"(without the quotes). Check out the sample for clarification.

Sample Input

```
3
5 4
10 8
12 5
9 7
15 11
```

Sample Output

```
Case 1: Bus number 5
Case 2: Bus number 1
Case 3: He is in the queue
```

12 11 4 9 9 7 10 7 15 13 12 10 2 3 3 3 5 5	
Note: Be careful about the output format.	
Problem Setter: Azharul Islam Tazib DataSet & Alternate Writer: Mahmud Sajjad Abeer Special Thanks: Muhaiminul Islam Jim	

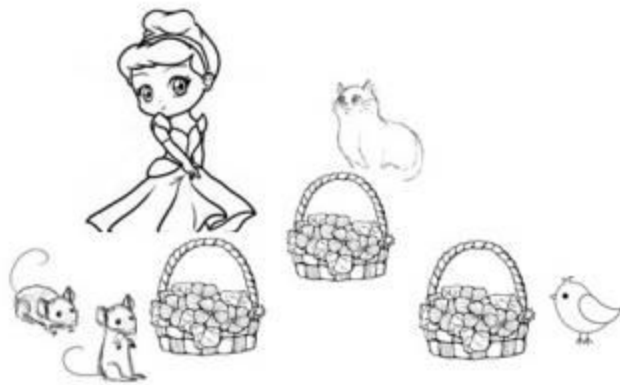
F. Intelligent Cinderella

Time Limit: 2 seconds

Memory Limit: 1024MB

Little Cinderella! Her stepmother gives her lots of work for a day. She has to work from morning till night. After finishing all her works, her happiness limits no bound because a beautiful fairy comes to her every night & teaches her different things.

Today, Cinderella has learned about Prime Numbers, Even Numbers & Odd Numbers. When beautiful fairy leaves Cinderella, Cinderella's friend's cat, mouse & bird come to visit her with some flowers. So, Cinderella decides that she'll make some garlands(माला) based on Prime numbers, Odd numbers & Even numbers.



First of all, she numbered all her flowers from 1 to N uniquely. Then she takes 3 baskets. In the first basket, she puts all the Prime numbered flowers. After that's done she puts all the Odd numbered flowers in the second basket & in the last basket she puts all the even numbered flowers. Then she gives the first basket to the cat, second to mouse & third to bird. After that, she told them to make garlands from their basket with the flowers on them. But she has set up a rule, each garland should have exactly K flowers and each of them should be made from their own basket.

For example, if anyone uses Prime numbered flower to make one garland he or she can't use Odd numbered flower to make that garland & same for even numbered flower & also odd numbered flower & she also tells them how many flowers will be used to make one garland. Now, you have to output how many garlands Cinderella's friends will make?

N.B: A Prime number is a number that has exactly two divisors, 1 and the number itself > 1 .

Input

The first line of the input is an integer T ($1 \leq T \leq 2000$) that is the number of the test cases. Then there will be T lines containing two integers N ($1 \leq N \leq 1000$) and K ($1 \leq K \leq 1000$). Here, N is the total number of flowers and K is the number of flowers needed to make each garland.

Output

There will be T lines of output, each containing 3 integers a , b , c separated by spaces (no leading or trailing spaces). Where a is the number of garlands the cat made, b is the number of garlands the mouse made and c is the number of garlands the bird made.

Sample Input

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

Sample Output

```
0 1 0
1 1 0
1 0 0
```

Note: Be careful about the output format.

Problem Setter: Tanzina Afroz Rimi

Special Thanks: Mahmud Sajjad Abeer

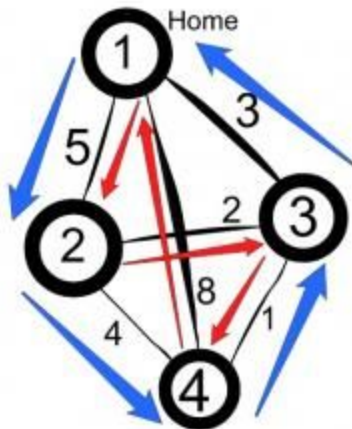
G. Bachao!!!

Time Limit: 1 second

Memory Limit: 64 MB

Your Best Friend Afrin Akhter has joined a new software farm recently. Their farm is working on a new mobile app “Bachao!!!”. This is a google map based application.

The app is trying to solve a specific type of problem:



Suppose, you have some specific tasks in 4 different places numbered 1, 2, 3, 4 in Dhaka where your home is in 1 and you must do all of them on that day following no specific order by visiting each place exactly once and your home twice (Getting out from home and getting back to home).

Here comes the app, it'll tell you the minimum time to go out from home, visit all those places and return back to home.

But as Afrin Akhter has so many boyfriends, (s)he is going to stay busy this week. As this problem is not so hard, she has asked you to solve this problem for her. She will provide a 4x4 time-cost table to you. You have to write a program that outputs the minimum time the user needs to do all of the tasks.

Input

The first line will contain an integer T ($1 \leq T \leq 100$) denoting the number of test cases. Then on each test case you will be given a single 4x4 time-cost table (explained below). Where each of the number x_{ij} will be within 0 to 100 where i is the row number and j is the column number. The test cases will be separated by a single blank line.

Output

For each test case, you will have to output a single line with a single integer N , the minimum time to visit all 4 places starting from home and ending at home visiting the other 3 places exactly once.

See the explanation for better understanding.

Cost Table

Let,

0 5 2

5 0 1

2 1 0

Be a 3x3 time-cost table. The places are numbered 1, 2, and 3. Where the row number denotes the source and column number denotes the destination.

The first line means it will cost 0 second to go from 1 to 1, 5 seconds from 1 to 2 and 2 seconds from 1 to 3.

The second line means it will cost 5 seconds to go from 2 to 1, 0 second from 2 to 2 and 1 second from 2 to 3.

The third line means it will cost 2 seconds to go from 3 to 1, 1 second from 3 to 2 and 0 second from 3 to 3.

See the picture for clarification.

Let, **a** be the source and **b** be the destination. Here, path **a->b** and **b->a** will cost exactly same for every source and destination.

Sample Input

2

0 5 3 8
5 0 2 4
3 2 0 1
8 4 1 0

0 9 5 2
9 0 7 4
5 7 0 1
2 4 1 0

Sample Output

13
18

Explanation of sample test case: The optimal path is(1 -> 2 -> 4 -> 3 -> 1) [Marked BLUE] and output is 5 + 4 + 1 + 3 = 13. (1 -> 2 -> 3 -> 4 -> 1) [Marked RED] is not optimal as it'll cost 16.

Problem Setter: Mahmud Sajjad Abeer

Alternate Writer: Muhaiminul Islam Jim

Original Idea: Mohammad Mahmudur Rahman

H. Startup

Time Limit: 1 second
Memory Limit: 1024 MB

Remember the Hoto Doridro Vampire we've mentioned last semester?

Nevermind. Don't panic if you can't remember the story. He's just a Hoto Doridro Vampire who loves to eat **Pure Red Juicy Tomato**. However, his tomato storage has ended again. It's a serious issue as he can't live without tomato, thus he needs more money. It's not possible to starve like this all the year.

So, Hoto Doridro Vampire got an idea. He wants to startup a NokshiKatha business as the Winter is Coming. The idea is very simple. Initially, he has designed 2 patterns that can be sewed on the NokshiKatha. The customers can place an order of **NxM NokshiKatha** where **N** is the height and **M** is the width. So, there will be **NxM** cells on the NokshiKatha. The customer is going to set up the rule how those 2 patterns are going to be used on the NokshiKatha and **1** logo of the company on any of the cell. If everything's done perfectly he'll get 1000\$ per NokshiKatha.

But, before sewing a NokshiKatha first he needs to draw the Pattern on the Katha so that everything's done perfectly. Now, he needs a programmer who can help. Tough he's not gonna share his tomatoes, but he's offering 10% company share with the programmer. Can you help that Hoto Doridro Vampire so that he can get successful on his startup, get rich and have a lifelong tomato supply?

<pre> </pre>	<pre> </pre>	<pre> _> /_ ***** ***** ***** ***** NokshiKatha </pre>
Pattern 1	Pattern 2	Company Logo

All of those patterns are exactly **11x11**. But when you print you'll have to use separator using **'+'** as outer border and **1 space padding** from the outer square. So the output of a single pattern will take a **15x15** space. However, to understand the output format properly, please check the sample output.

Input

The first line will contain a single integer **T** (**1 <= T <= 100**). Then on each test case, the first line will contain two integers **N** and **M** (**1 <= N, M <= 11**). Then there will be **N** lines containing **M** integers **X_{ij}** which denotes the pattern number that needs to be printed(here, **i** is the

row and j is the column number). Then on the last line of each test case, there will be the position P ($1 \leq P \leq N$) and Q ($1 \leq Q \leq M$) where P denotes the row and Q denotes the column where the logo of the company will be printed.

Output

For each test case you'll have to print "Case X:" on the first line where X is the number of test case starting from 1. And then there will be exactly $N14+1$ lines with $M14+1$ characters on each of them containing the required design. Check out the sample and picture for clarification.

Sample Input

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

Sample Output

```
Case 1:
+++++
+++++
+      +      +      +
+ |||| |||| + || |||| || + |||| |||| +
+ ||| | ||| + || | ||| + |||| |||| +
+ ||| ||| + || | ||| + ||_|>|/_||| +
+ || |||| || + | || ||| + |*****| +
+ | |||| || + || |||| || + |*****| +
+ ||| |||| + || | ||| + |*****| +
+ | |||| || + || |||| || + ||*****|| +
+ || |||| || + | || ||| + |||| |||| +
+ ||| ||| || + || | ||| + |||| |||| +
+ |||| | ||| + ||| ||| || + NokshiKatha +
+ |||| |||| + || |||| || + |||| |||| +
+      +      +      +
+++++
+++++
+      +      +      +
+ || |||| || + |||| |||| + || |||| || +
+ ||| ||| || + |||| | ||| + || | ||| +
+ ||| | ||| + || | ||| || + || | ||| +
+ | ||| ||| + || |||| || + | ||| ||| +
+ || |||| || + | |||| || + || |||| || +
+ ||| ||| || + |||| |||| + || ||| ||| +
+ || |||| || + | |||| || + || |||| || +
+ | ||| ||| + || |||| || + | ||| ||| +
+ ||| | ||| + ||| ||| || + || | ||| +
+ || |||| || + |||| |||| + || |||| || +
+      +      +      +
+++++
+++++
+      +      +      +
+ |||| |||| + || |||| || + |||| |||| +
+ |||| | ||| + || | ||| || + |||| | ||| +
+ ||| ||| || + || | ||| + || | ||| || +
+ || |||| || + | ||| ||| + || |||| || +
```

	<pre>+ + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + +++++++ +++++++ Case 2: +++++++ +++++++ + + + + + + + + + + + + + + + _> /_ + + + + ***** + + + + ***** + + + + ***** + + + + ***** + + + + + + + + + + + + NokshiKatha + + + + + + + + + +++++++ +++++++</pre>
--	---

Note: Please be careful about the output format.

Problem Setter: Mahmud Sajjad Abeer
Alternate Writer: Rajdip Saha