

# DIU Take-Off Programming Contest Fall 2022 Slot C

Daffodil International University

<https://toph.co/c/diu-take-off-fall-2022-slot-c>



## Schedule

The contest will run for **3h15m0s**.

## Authors

The authors of this contest are anisur\_rahman, farhin\_khaled, Fellow\_junior, mh\_naim, Nabid1221, nadimnesar, RAIKO\_, and wrench786.

## Rules

This contest is formatted as per the official rules of ICPC Regional Programming Contests.

You can use C++11 GCC 5.3, C++11 GCC 7.4, C++14 GCC 5.3, C++14 GCC 8.3, C++17 GCC 9.2, C++20 GCC 12.1, C11 GCC 12.1, and C11 GCC 9.2 in this contest.

Be fair, be honest. Plagiarism will result in disqualification. Judges' decisions will be final.

## Notes

There are 8 challenges in this contest.

Please make sure this booklet contains all of the pages.

If you find any discrepancies between the printed copy and the problem statements in Toph Arena, please rely on the later.

# A. Inception

Nahil was recently admitted to Daffodil International University. One day, his university alumni named Kian came to his university to join a seminar to educate students about programming. He gave students a lot of important information about programming. To define competitive programming, he told:

*"Programming is fun. Programming is a mental sport and when this sport is held on the internet involving a sports programmer as a contestant, then it is called Competitive Programming."*

He mentioned about Daffodil International University's "Problem Solving Community" and its activities like Training classes, organizing the weekly contest and a lot more.

To train programmers, DIU ACM has a "Blue Division". Blue Division programmers have to solve selected problems from the "Blue Sheet". DIU ACM provides many special programming-related facilities for them. They can attend advanced training classes and have the chance to get prizes in several categories based on their performance. Also, they have the opportunity to attend "Programmers Meetup" where they can meet with their inspiring alumni. To get into the blue division you have to solve selected problems from "Beecrowd Online Judge".

Nahil felt inspired a lot. So, he decided to start his programming journey by printing, "I believe, I can be a good programmer."

So, your task is to write a program to print this line: **"I believe, I can be a good programmer."** without quotes.

Let me help you with the C program to print "Hello World".

```
#include <stdio.h>
int main() {
    printf("Hello World\n");
    return 0;
}
```

## Input

There is no input in this problem. Only print the mentioned sentence and submit.

## Output

Print "I believe, I can be a good programmer." without quotes.

## Samples

<u>Input</u>	<u>Output</u>
	I believe, I can be a good programmer.

## B. Equability

Rehnuma and Syma are two friends. Syma loves to solve problems. One day Rehnuma thought of challenging Syma with a problem. So, Rehnuma gave Syma three numbers  $A$ ,  $B$  and  $K$ . Rehnuma asks to compare the result of adding  $A$  and  $B$  with  $K$ .

There are three possibilities,

- $A + B$  equal to  $K$
- $A + B$  less than  $K$
- $A + B$  greater than  $K$

Rehnuma has to tell whether the result of adding  $A$  and  $B$  is equal or greater or less than  $K$ .

### Input

Each test case contains three integers  $A$ ,  $B$  and  $K$ .

$(1 \leq A, B, K \leq 1000)$

### Output

Print "Equal" if the result of adding  $A$  and  $B$  is equal to  $K$ , otherwise, print "Less" if the result is less than  $K$ , else print "Greater" if the result is greater than  $K$  without quotes.

### Samples

<u>Input</u>	<u>Output</u>
1 2 3	Equal

<u>Input</u>	<u>Output</u>
7 3 15	Less

<u>Input</u>	<u>Output</u>
4 6 8	Greater

## C. A Cricket Match

Prof. Bari and his friends went to watch a one-day cricket match. They were enjoying the match as their supported team was winning. Suddenly a technical problem arose and the audience couldn't see the scoreboard. Bari wrote down the score and the number of balls his favorite team had played but didn't calculate the remaining overs. So, the other friends do not have any idea about how many overs are left to bat and Bari is busy writing the score and counting the ball numbers. So as a friend of Bari and as you are a programmer, Bari asked you to make a program so that the rest of the friends could know the remaining overs and watch the game without any disturbance.

**Careful,** As it is a one-day cricket match, A team can play a maximum of 50 overs. An over consists of 6 balls.

### Input

A single integer  $N$  indicates the number of Balls played.

$$1 \leq N \leq 300$$

### Output

You have to output one line in the format " $X$  Over and  $Y$  Ball remaining" (without quotes) where  $X$  is the number of remaining overs and  $Y$  is the number of remaining balls. Check out the samples for clarification.

### Samples

<u>Input</u>	<u>Output</u>
295	0 Over and 5 Ball remaining

<u>Input</u>	<u>Output</u>
294	1 Over and 0 Ball remaining

## D. Line Segment

A random kid named 'Mofiz' loves line segments in the **number line**.

One day he and his friend 'Sofik' playing a game about line segments. Mofiz gives two random line segments  $(x_1, y_1)$ ,  $(x_2, y_2)$  and asked to Sofik either  $(x_2, y_2)$  is inside or equal to  $(x_1, y_1)$ .

if the  $(x_2, y_2)$  segment is equal to the  $(x_1, y_1)$  segment then output will be "Equal". Else if the  $(x_2, y_2)$  segment is inside of  $(x_1, y_1)$  segment then output will be "Inside" otherwise output will be "Nothing".

**More Formally,**

**Equal:** If  $x_2$  is equal to  $x_1$  and  $y_2$  is equal to  $y_1$ .

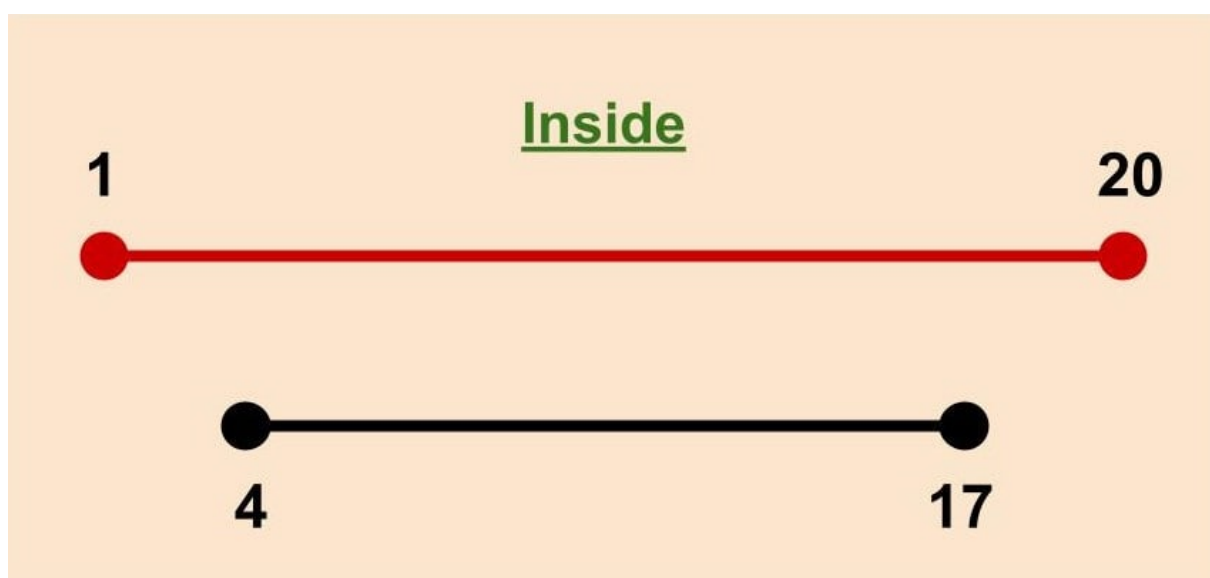
**Inside:** If its not equal then,

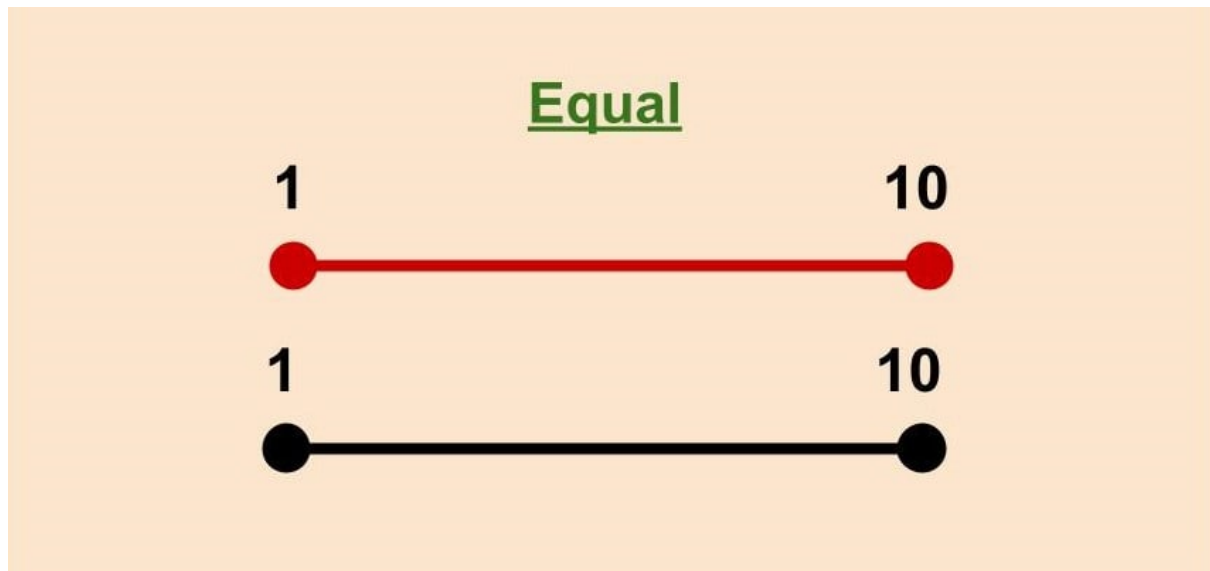
- If  $x_2$  is greater than or equal to  $x_1$
- and  $y_2$  is less than or equal to  $y_1$ .

**Nothing:** If above both conditions not hold its nothing.

Sofik is so busy that he can't play this easy game. So he asked for your help to write a program to give the right answer either "Inside", "Equal", "Nothing".

**Example,**





## Input

The first line will contain four integers  $x_1, y_1, x_2, y_2$  separated by spaces.

$1 \leq (x_1 < y_1), (x_2 < y_2) \leq 100000$

## Output

You have to output either "Inside", "Equal", "Nothing" (without quotes).

## Samples

<u>Input</u>	<u>Output</u>
1 7 1 7	Equal
<u>Input</u>	<u>Output</u>
1 3 2 9	Nothing
<u>Input</u>	<u>Output</u>
1 20 1 15	Inside
<u>Input</u>	<u>Output</u>
1 100 5 100	Inside



## E. Lucky Numbers V1

Nekobasu is a random college student who thinks he is an expert in number theory. One day Nekobasu found a theory that a number has lots of divisors.

**Divisor of a Number:** In division, we divide a number by any other number and get a remainder. If we divide  $N$  with  $X$  and found  $remainder = 0$ , then  $X$  is divisor of  $N$ .

**For Example,**

The diagram shows a division operation:  $4 \overline{)20} 5$ . Arrows point from the parts of the operation to labels: an arrow from the '4' points to 'Divisor', an arrow from the '20' points to 'Dividend', an arrow from the '5' points to 'Quotient', and an arrow from the '0' in the result points to 'Remainder'.

So, we can say that 4 is a divisor of number 20.

Nekobasu loves certain kinds of numbers which have divisor 3 or 5. He calls them lucky numbers.

Today he is walking to college which is in a mountain. He has to walk  $M$  stairs to reach college. The stairs are numbered from 1 to  $M$ . In one step he can go to the next stairs like, if he is in  $i$ 'th stair, in one step he can go  $i + 1$  stair. Currently he is in step  $N$ 'th stair.

Suddenly, he got an idea that he can count how many stairs there are, which are lucky from steps  $N$  to  $M$ . So that after reaching college he can tell his friends about how many lucky numbers he had encountered.

**More formally,** he will check the numbers  $n, n + 1, \dots, m - 1, m$  ( $N < M$ ) and will count how many of them are lucky and print that.

**Example:**

$$N = 8, M = 17$$

Number	Divisors	Verdict
8	1,2,4,8	Not Lucky
9	1,3,9	Lucky
10	1,2,5,10	Lucky
11	1,11	Not Lucky
12	1,2,3,4,6,12	Lucky
13	1,13	Not Lucky
14	1,2,7,14	Not Lucky
15	1,3,5,15	Lucky
16	1,2,4,8,16	Not Lucky
17	1,17	Not Lucky

So, here you can print 4.

## Input

The first line will contain two integers  $N, M$  separated by spaces.

$$1 \leq N < M \leq 100000$$

## Output

You have to output a single integer  $X$ , where  $X$  is the result of the test case. Check out the samples for clarification.

## Samples

<u>Input</u>	<u>Output</u>
2 10	5

<u>Input</u>	<u>Output</u>
8 17	4

## F. Palindrome Revelation

Once a brave man went out in search of treasure in a cave called "Matai Hakor". Then while searching for the treasure, he finds a big box. He jumps for joy. But opened the box and saw only one paper with one word written on it. After that moment he was upset but reading the paper carefully he realized that the word written on it with random English alphabets was not an ordinary word. He discovered that reversing the word written on this paper would not change the word meaning. Then he named this type of word "Palindrome".

**For example,** consider this word "AibohphoBia". After reversing this word it becomes "aiBohphobiA". We can see meaning of the both word is same. So it is considered as a palindrome. (uppercase or lowercase letter doesn't effect in meaning of a word)

Now let's come up with the problem, you are given an integer  $N$  size of the string  $S$  and string  $S$  itself with lowercase or uppercase English alphabets. You have to check whether string is palindrome or not.

### Input

The first line will be an integer  $N$  size of string  $S$  and the next line will be string  $S$  itself.

$(1 \leq N \leq 1000)$

### Output

You have to output "Yes" if the given string is palindrome, otherwise print "No" without quotes.

### Samples

<u>Input</u>	<u>Output</u>
10 palindRome	No

<u>Input</u>	<u>Output</u>
15 adeiLmnonmliEda	Yes

## G. Milky Way Galaxy

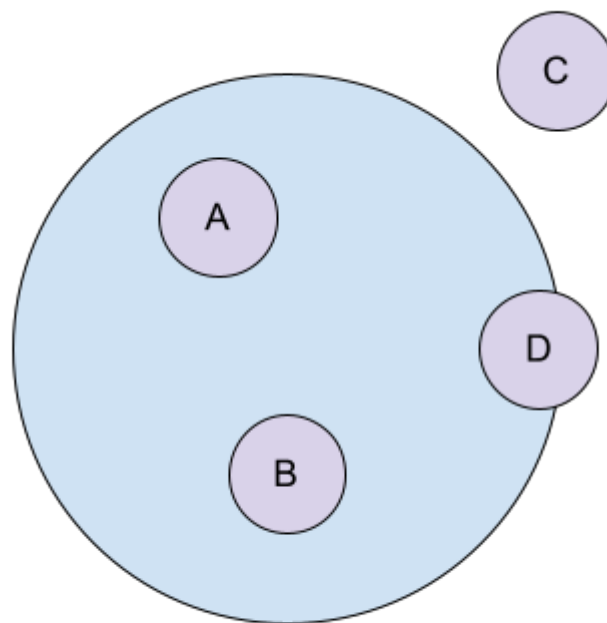
There are so many planets in our Milky way galaxy. It's really amazing that all planets are actually circular. Let's imagine our milky way galaxy is also a circle and you know the radius and center point of our milky way galaxy.

Because you are always interested in the milky way galaxy, your science teacher gives you  $N$  planets radius and center point and asks you to calculate how many planets are inside of the milky way galaxy and also said to calculate distance between center of milky way galaxy and center of any planet he/she need Euclidean distance formula,

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

**Note:** Use "Double" in all calculation for precision errors safety.

**For Example,**



A and B inside, C and D outside.

### Input

The first line will contain radius and center point  $P$  and  $Q$  of the milky way galaxy. The next line will be  $N$  which is the number of planets we have to check. For each planet

there will be  $N$  lines containing radius and center point  $X$  and  $Y$  of the planet. Check out the samples for clarification.

$$1 \leq \text{Radius of Milky Way} \leq 100$$

$$0 \leq P, Q \leq 100$$

$$1 \leq N \leq 10$$

$$1 \leq \text{Radius of Planet} \leq 1000$$

$$0 \leq X, Y \leq 1000$$

## Output

You have to output a single integer which is the number of planets completely inside the milky way galaxy.

## Samples

<u>Input</u>	<u>Output</u>
7 2 3 2 3 1 3 8 2 3	1

<u>Input</u>	<u>Output</u>
100 0 0 5 53 0 1 12 50 70 20 90 90 88 5 10 1 70 70	4

# H. Lucky Pairs V2

— *This is the hard version of the previous lucky number V1 problem.*

After successfully solving the previous problem. Nekobasu trying to flex with other friends, how expert he is in Number Theory! At this moment a backbencher student Mikasa gives him a simple task if he can do it, then Mikasa will accept that Nekobasu is pure expertise in number theory. Mikasa just introduced to him two words “GCD” and “Lucky Pair” in this particular problem.

**GCD:** GCD means (Greatest Common Divisor). [For Divisors Concept see v1 Problem]

## Greatest Common Divisor

$$a = 45 = 3 \times 3 \times 5$$

$$b = 30 = 2 \times 3 \times 5$$

Common Factors

$$3 \times 5$$

$$\text{GCD of } 45 \text{ and } 30 = 15$$

**More formally,** If  $a$  and  $b$  are two numbers then the GCD = Greatest Common Divisor of both numbers is  $\gcd(a, b) = g$ . where  $g$  is the largest number which is divisors both  $a$  and  $b$ .

**Lucky Pair:** Pairs of Integers  $(a, b)$  which satisfied both of the below's conditions:

- GCD of pair  $(a, b)$  are greater than 1. more formally  $(\gcd(a, b) > 1)$
- $b$  is divided by  $a$ . more formally,  $(b \bmod a == 0)$

**Lucky Pairs Example:**

Pair ( a, b)	gcd( a, b)	b % a == 0	Verdict
1 , 5	1	Yes	Not - Lucky
2 , 4	2	Yes	Lucky
4 , 6	2	No	Not - Lucky
6 , 24	6	Yes	Lucky
23 , 16	1	No	Not - Lucky
7 , 7	7	Yes	Lucky
8 , 24	8	Yes	Lucky

Now consider yourself as Nekobasu. your friend Mikasa will give you an integer  $N$ . You have to tell how many distinct Lucky pairs  $(a, b)$  have where  $(a, b)$  range will be,  $(1 \leq a, b \leq N)$ . you need to count the number of pairs and answer the question to prove yourself an expert in Number theory.

**Note:** Two pairs of integers  $(a, b)$  and  $(c, d)$  are considered distinct if at least one element of  $(a, b)$  does not also belong to  $(c, d)$ .

## Input

- The first line of input data contains a single integer  $T$  ( $1 \leq T \leq 10^6$ ) — the number of test cases.
- each test case contains an integer  $N$  ( $1 \leq N \leq 10^7$ ).

## Output

- You have to output a single integer  $X$ , where  $X$  is the total number of Distinct Lucky Pairs.

## Samples

Input	Output
3	4
4	8
6	12
8	

<u>Input</u>	<u>Output</u>
For $N = 4$ , Total Lucky Pairs = 4 All possible distinct Lucky pairs are = $(2, 2), (2, 4), (3, 3), (4, 4)$	

**Warning:** Large I/O data, It is often recommended to use `scanf/printf` instead of `cin/cout` for fast input and output.