# Exam "Fundamentals of programming with C#" – 21 January 2018

## Problem 4. Make it Rain

## History

You had an amazing success with the Liquid Sorting Algorithm and you decide to create a new Crypto Currency called – “Roin” (short from “Rain-coin”). Your goal is to make as much money as you can from this currency, so you want it to be perfect. **Your job is to implement an innovative algorithm for the new “Roin”.**

## Description

On the first line you will receive an integer – **N**. On the next **N\*3** lines you are going to receive triplets (pack of three) of integers – **A**, **B** and **C, each on a new line**.

The algorithm you came up is the following:

First check **IF YOU CAN** divide **A** by **B** and get a result equal to **C.** The result of the check should be **“True”** or **“False”.** You get the first letter of the result (either **‘T’** or **‘F’**) and you must increment a **counter for each occurrence** of the two letters **but in a special way**:

1. Each time you get a ‘**T**’ you add to its count the **ASCII code** of the symbol ‘**T**’ and modify the count of ‘**F**’ by the following \***special formula** - **fCount = fCount / (ASCII code of ‘T’** **/ 10).**
2. Each time you get a ‘**F**’ you add to its count the **ASCII code** of the symbol ‘**F**’ and modify the count of ‘**T**’ by the following \***special formula** - **tCount = tCount / (ASCII code of ‘F’** **/ 10**).

In the end you need to understand if you have mined a “Roin”. The way you do that is by **dividing** ‘**T**’ count by ‘**F**’ count. **IF YOU CAN** do that and the **result** **is** an **even** **number** you have a “Roin”. The result of the check must also be “**True”** or **“False”**.

You must print **your output** in a special **format** which is described in the **Output section**.

### Input

1. **N** – integer [1 … 100]
2. On the next **N\*3** lines you will receive triplets of integers: **A**, **B** and **C**. **Each integer will be on a new line!**

### Output

You must print the following three strings, **each on a new line**:

“**T: {0}**” ({0} is the calculated count of the ‘T’ symbols)

“**F: {0}**” ({0} is the calculated count of the ‘F’ symbols)

“**Got a Roin coin: {0}**” ({0} is either “True” or “False”)

Examples:

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Description** |
| 3  12  1  8  6  3  2  24  12  2 | T: 168  F: 1  Got a Roin coin: True | 12 / 1 is not equal to 8 so you get a ‘F’.  You add 70 to the ‘F’ count and calculate ‘T’ count by the **\*special formula**. T: 0 and F: 70  6 / 3 is equal to 2 so you get a ‘T’.  You add 84 to the ‘T’ count and calculate ‘F’ count by the **\*special formula**. T: 84 and F: 8  24 / 12 is equal to 2 so you get a ‘T’.  You add 84 to the ‘T’ count and calculate ‘F’ count by the **\*special formula**. T: 168 and F: 1  168 / 1 = 168 which is an even number so the result is “True” |
| **Input** | **Output** | **Description** |
| 2  89  45  2  52  4  7 | T: 0  F: 140  Got a Roin coin: True | 89 / 45 is not equal to 2 so you get an ‘F’.  You add 70 to the ‘F’ count and calculate ‘T’ count by the **\*special formula**. T: 0 and F: 70  52 / 4 is not equal to 2 so you get an ‘F’.  You add 70 to the ‘F’ count and calculate ‘T’ count by the **\*special formula**. T: 0 and F: 140  0 / 140 = 0 which is an even number so the result is “True” |