

1183 – Computing Fast Average

Given an array of integers (**0** indexed), you have to perform two types of queries in the array.

1. **1 i j v** - change the value of the elements from i^{th} index to j^{th} index to **v**.
2. **2 i j** - find the average value of the integers from i^{th} index to j^{th} index.

You can assume that initially all the values in the array are **0**.

Input

Input starts with an integer **T** (≤ 5), denoting the number of test cases.

Each case contains two integers: **n** ($1 \leq n \leq 10^5$), **q** ($1 \leq q \leq 50000$), where **n** denotes the size of the array. Each of the next **q** lines will contain a query of the form:

1 i j v ($0 \leq i \leq j < n$, $0 \leq v \leq 10000$)

2 i j ($0 \leq i \leq j < n$)

Output

For each case, print the case number first. Then for each query of the form '**2 i j**' print the average value of the integers from **i** to **j**. If the result is an integer, print it. Otherwise print the result in '**x/y**' form, where **x** denotes the numerator and **y** denotes the denominator of the result and **x** and **y** are relatively prime.

Sample Input	Output for Sample Input
1 10 6 1 0 6 6 2 0 1 1 1 1 2 2 0 5 1 0 3 7 2 0 1	Case 1 : 6 16/3 7

Note

Dataset is huge. Use faster i/o methods.