# C++ Fundamentals – Exam (17 November 2019)

Write C++ code for solving the tasks on the following pages.

Code should compile under the C++11 standard.

Submit your solutions here:  
<https://judge.softuni.bg/Contests/1751/CPlusPlus-Fundamentals-Exam-17-November-2019>

Only source code will be accepted as solution for each task.

# Task 2 – Square Manipulator

Your task is to write a program, which for a given rectangle of numbers performs calculations for smaller squares(with given size) within that rectangle.

For each small square, in which the sum of numbers constructing it **exceeds some given value** -   
output the **average number** for that square.

The **average number** for a square is calculated by the sum of all the numbers constructing it  
divided by the count of the numbers.

NOTE: the result should be **rounded down** to the **nearest integer**.

Since they may be more than one small squares that fulfill the requirements you should  
**output** all of their **averages** in **sorted ascending order** divided by a **whitespace**.

Example input:  
> 3 2 (rectangle height and rectangle width)  
 0 2  
 6 4  
 5 -1  
> Square side: 2  
> Target sum: 7

Example output:

**3 3**

Explanation:

Both squares **0 2** and **6 4** fulfill the requirements(their sum is bigger than target sum 7) and should be processed.  
 **6 4** **5 -1**  
**0 + 2 + 6 + 4** = 12 / 4 = **3**  
**6 + 4 + 5 – 1** = 14 / 4 = **3**  
They are sorted in ascending order and the output is “**3 3**”

### Input

First read two integers (Rows) and (Cols) indicating the size of the input rectangle.

Next read (Rows) **lines** of **whitespace** separated integers. Each row should contain exactly (Cols) integers.

On the last line read two integers (squareSize and targetSum).

### Restrictions

Time limit: 250ms (0.25s)  
Memory limit: 16 MB

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3 2  0 2  6 4  5 -1  2 7 | 3 3 |
| 3 3  1 2 3  4 5 6  7 8 9  1 6 | 6 7 8 9 |
| 3 4  9 2 4 6  9 5 3 1  9 5 7 9  3 41 | 4 5 |