DV1445 Coursework 1 : Triangles
Deadline: Fri 24th Feb

1   Description

This task involves mapping data from one representation to another. Your program will operate on 2-dimensional points (x, y). The values on the X axis will lie in the range $0 \leq x < 4$, and the values on the Y axis will lie in the range $0 \leq y < 4$. These points will not be integer so your program must be able to handle points such as $(2.31, 3.99)$. The 2-dimensional plane is divided into triangles as shown in Figure **??**. The core problem that you must solve is how to map the 2D-points into the number of the triangle that contains them. Once you have solved this problem you will reuse the solution in a program that performs the following actions:

1. Input an x and y value from the user. You can assume they are in the correct range (you do not need to check).

2. Convert the point to a triangle number.

3. Output the triangle number to the screen.

4. Do the following five times:

   Generate a random point in the correct range

   Convert it to a triangle number

   Output the x, y and triangle number to a line on the screen

NOTE: Your program must **not** contain any if-statements or loops. You are allowed to use input statements, output statements and assignments.
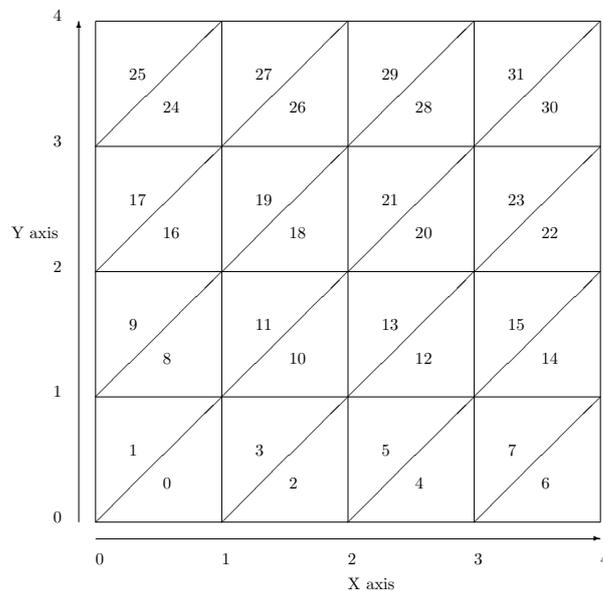

Figure 1: Triangle layout

2 Examples

First we give some examples of the mapping. For each case work out where the point lies on the diagram and then see which triangle number is produced:

Point	Contained in Triangle
$(2.31, 3.99)$	29
$(0, 0)$	0
$(1, 1)$	10
$(0.5, 0.5)$	0
$(3.6, 1.2)$	14
$(0.999, 3)$	24

Now we present an example execution of the program, in each execution the five random points should be different (i.e. you must seed the random number generator as shown in the lecture slides).

```
Please enter the first point:
2.67
3.01
Triangle number: 28
(0.124, 1.091) -> 8
(3.435, 0.941) -> 7
(1.770, 0.482) -> 2
(3.712, 1.811) -> 15
(2.309, 2.492) -> 21
```

3 Hints

It will be difficult to jump directly from the point coordinates to the triangle number. Think about the techniques that you have seen on the course for breaking a problem into simpler pieces. You will need an intermediate step to help you. Think about the numerical patterns that you have used so far, odd/even numbers, additive steps, division. You will need a conversion between types (casting) at some point — choose a point where it helps you the most to simplify your problem.

Think about how the geometry can help you, there are not just triangles in the diagram, they also form squares. Can these help you to split the problem into simpler steps? Something to consider is whether numbering the squares in some way would give you the intermediate step that makes the problem simpler.

Think about the original real number inputs, they may be more useful if you split them into the integer part and the fractional part, how would `floor` help you here? Each square in the diagram is cut into two pieces by a diagonal line, what pattern in the numbers separates the points below this line from those above it? Can you use this pattern directly or do you need to transform it?

Remember you have two weeks for this assignment — it is supposed to take 24 hours of your time (or three working days). We expect you to use that amount of time to experiment and discover how to solve this problem.

4 Submission

You must email your source code to `cw.bth.dv1445@gmail.com`. The subject of your message *must* be your 10-digit person number (two year digits, month, date and the final four digits). We will use this to automatically file your message, if you do not set this properly your submission will not count. You must attach your source code to the email message, use your person-number with `.cpp` on the end as the file name.

We will check the submissions and you will receive a reply on Monday 27th Feb, either telling you that your submission was valid (and that it compiles ok), or that you must resubmit to fix a problem.

5   Marking Scale

The first assignment is a U/G grade. In order to pass with a grade your program must fulfil ALL of the following criteria:

- It must compile without ANY modifications.

- It must input the first pair of coordinates.

- It must output the correct triangle number (we will make the test data it is marked with).

- It must generate five random points in the correct range.

- It must map each of the five points to the correct triangle number.

- It must output the information for each of the five points.

After we have checked that your program works you will need to sign up for a viva on your program. These vivas will be in the two weeks after the submission deadline, you must physically be here to attend the viva, in which you must:

- Attend the viva at the correct time

- Answer questions correctly about the design of your program

- Answer questions correctly about how your program works

If you are unable to demonstrate that you wrote the program yourself then you will fail the assignment.