Lancaster University

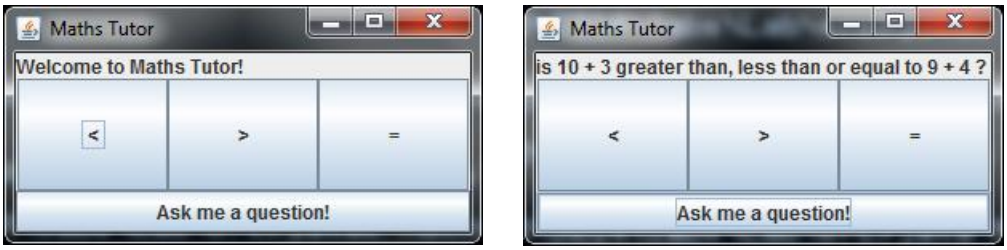
Problem Set 7: Building your own GUIs in Swing

Aims

This week you'll be working with the Java Swing classes to build your own simple application with a graphical user interface (GUI). After completing this lab, you should have developed a deeper grasp of integrating a range of existing classes to build a more complex application, and have gained your first experience of using asynchronous programming techniques.

The Task

Simple computer programs often make good tools to support education. Your task this week is to write a simple Maths Tutor program to help primary school children learn about inequalities (less than and greater than).



You'll probably need to refer to the Week16 and Week17 lecture notes and examples to complete this task, and don't forget you can also refer to the Java API documentation to find out what constructors and methods the Swing classes have: http://docs.oracle.com/javase/8/docs/api/

Task Specification

Your program should:

- use a class to represent your application. ***Do not put all your code in your main method***
- use Swing to build the user interface. ***Do not use System.out.println(), except for debugging***
- use a read-only piece of text and four buttons, as shown in the example above. Try to make the layout of your GUI simple, but easy to use.
- display a welcome message when the program starts.
- display a random question of the form "**is a + b greater than, equal to or less than c + d**" when the 'ask me a question' button is pressed. a, b, c and d should be randomly generated integers in the range 1 ... 10.
- display "correct" or "incorrect" appropriately when the user presses the appropriate less than, greater than or equal to button.

HINT: The **Math.random()** method returns a floating point number in the range 0.0 ... 1.0
HINT: You can round a floating point number into an integer using a *cast*, just as you do in C. e.g.

```
int i = (int) 3.14159265;      // Would assign i with the value 3.
```

Lancaster University

Additional Tasks

Still looking for more? Why not try one of these additional tasks…

Add different mathematical operations. Simply asking all questions of the form "is a + b greater than, equal to or less than c + d" can be rather limiting… extend your program so that it can handle different forms of question. Perhaps involving subtraction, multiplication, and combinations of all the above! You could add selectable difficulty levels too…

Add a score. Time how long it takes a user to answer the questions, and record how many correct/incorrect answers given by the user. Display how well the user has performed through your GUI. Perhaps even consider a high score table, based on the time taken to correctly answer ten questions…

Build a GUI interface to your games from last week. I'm sure many of the games you wrote over the last few weeks might benefit from a start screen, settings window, high score table etc… Try writing a simple GUI to interface with your own GameArena applications!