**CO520 Assignment 3 GUI Application 50 marks**

**Due Date: 18th March 2019, 23h59, Week 22**

**Jacqui Chetty**

***Submission:*** *Moodle, submit your whole BlueJ project as a single .jar or .zip file on the Moodle webpage. Make sure that the source files are included.*

*Background description:*

You have been tasked to create a game called chasing-bombs-login (login being your login). The aim of the game is for a player to click on a square and hopefully avoid a bomb that is hidden under the square. Each time a player clicks on a square and the bomb is not on that square, the player gains a point. The player continues to receive points until the square with the bomb is clicked and then a player loses the game. A player can choose difficulty levels, where the number of clicks on a square is reduced (to make it easy for a player wants to win the game).

*Container:*

Make use of a JFrame in which all other containers and components should be populated. [2 marks]

*Components:*

The frame is to be divided into 3 panels (see picture below),   
namely: [5 marks]

* 5 X 2 grid that should contain the rows and columns;
* The middle panel should contain 2 buttons, namely “Play the game” and “exit”;
* The last panel will should contain 3 buttons, namely “Easy”, “Intermediate” and “Difficult”, to represent the difficulty level.

Let’s refer to each panel as Panel A, Panel B and Panel C.

*Layout:*

You may decide on the colours the game should consist of.   
Hint: You may need to consider your layout managers carefully to ensure that your 3 panels are properly situated and that they are uniform. Also consider that each panel may require a different layout manager depending on the layout of the components within that panel. [5 marks]

*Functionality:*

Possible Listeners *–* MouseListener / ActionListener [2 marks]

Panel A –

* As seen above Panel A consists of 10 smaller panels which we will call squares (set in a grid). Hint: Consider the Checkerboard example that we covered in a lecture. [5 marks]
* The aim of the game is for a player to click on each square and gain points. The player must score a point every time they click on a   
  square. [5 marks]
* If they click on a square and the bomb is not there that square must change colour (any colour) to indicate that they have clicked on that   
  square. [6 marks]
* Your program must include the functionality to ensure that the “bomb is hidden” inside one of these squares within Panel A. By the bomb being hidden, all this means is that when the player clicks on the square with the “bomb”, a message is displayed that says “You lose! You got 5 points” (the points being a variable number depending on the player’s outcome). This message should appear in Panel B. [5 marks]

Panel B -

* This panel must ensure that a player can begin the game as well as exit the game when they choose to. [3 marks]
* This panel must ensure that a player can start a new game once a game is completed (Hint: If the previous game has changed the colours of the squares these need to revert back to their original colour. [5 marks]
* This panel is also responsible for outputting the message “You lose! You got 5 points” when the player loses. [2 marks]

Panel C -

* This panel is responsible for ensuring that the player can play the game at a difficulty level that they feel comfortable.
* If the player chooses “easy” then the player will win after 5 safe panels (tries). If “intermediate” is chosen 7 tries and there are no safe panels for a player playing at “difficult” level. [5 marks]

***Further Advice:***

***Comments:*** *Comments are fundamental to understanding programs, whether the programs are written by other people (especially in a development team) or we ourselves have forgotten details after some time. Every class should have a meaningful class comment. If you modify a given class, you should add yourself as an author and update the version. The given projects use dates for versions.*

*Also, every method should have some basic comments so that anyone using a class can understand what a method does without having to read its implementation.*

***Plagiarism:*** *The work you submit must be your own. We will run checks on all submitted work in an effort to identify possible plagiarism, and take disciplinary action against anyone found to have committed plagiarism.*

*Some guidelines on avoiding plagiarism*

*One of the most common reasons for programming plagiarism is leaving work until the last minute. Avoid this by making sure that you know what you have to do (that is not necessarily the same as how to do it) as soon as an assessment is set. Then decide what you will need to do in order to complete the assignment. This will typically involve doing some background reading and programming practice. If in doubt about what is required, ask a member of the course team.*

*Another common reason is working too closely with one or more other students on the course. Do not program together with someone else, by which I mean do not work together at a single PC, or side by side, typing in more or less the same code. By all means discuss parts of an assignment, but do not thereby end up submitting the same code.*

*It is not acceptable to submit code that differs only in the comments and variable names, for instance. It is very easy for us to detect when this has been done and we will check for it.*

*Never let someone else have a copy of your code, no matter how desperate they are. Always advise someone in this position to seek help from their class supervisor or lecturer. Otherwise they will never properly learn for themselves.*

*It is not acceptable to post assignments on sites such as RentACoder and we treat such actions as evidence of attempted plagiarism, regardless of whether or not work is payed for.*

*Further advice on plagiarism and collaboration is also available*

*You are reminded of the rules about plagiarism that can be found in the Stage I Handbook. These rules apply to programming assignments. We reserve the right to apply checks to pro- grams submitted for assignment in order to guard against plagiarism and to use programs submitted to test and refine our plagiarism detection methods both during the course and in the future.*