CSCI 1130A&B Introduction to Computing Using Java 2022-2023 First Term
Department of Computer Science and Engineering
The Chinese University of Hong Kong

Due date: 22 September 2022 (Thu) **Assignment 1** Full mark: 100
Expected normal time spent: 4 hours

Relax: this is a simple while very useful assignment to kick-off our course.

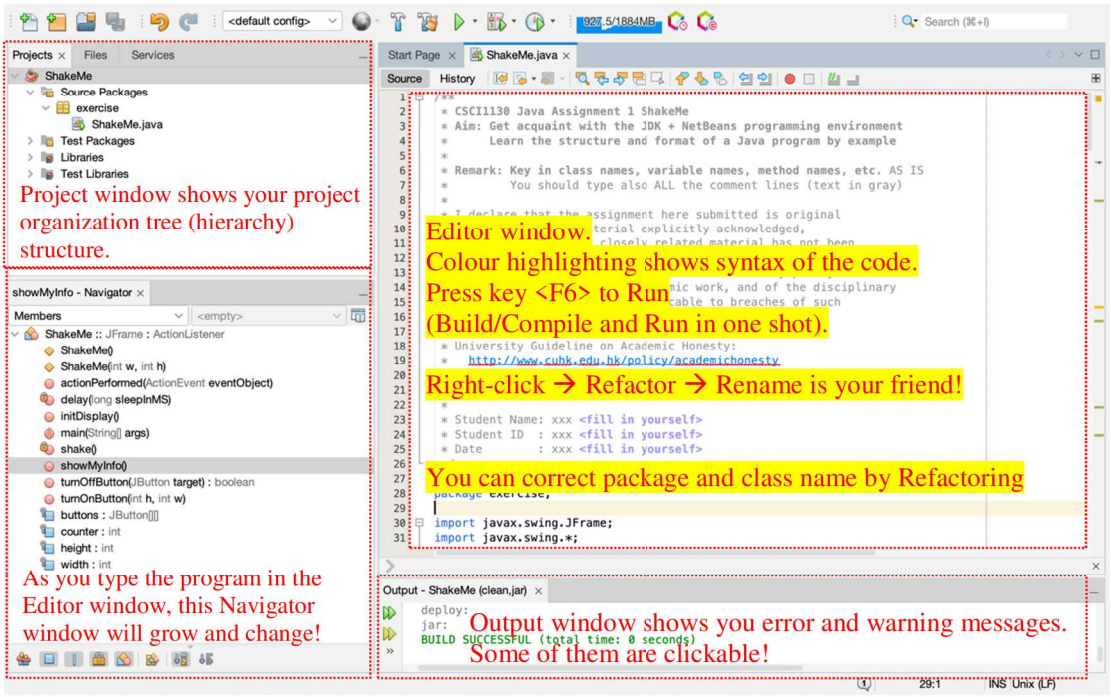
Aims: 1. Get acquaint with the NetBeans IDE, our Java programming environment.
2. Learn the structure and format of a Java program by example.

Procedure:

1. Use a computer with *both* JDK and NetBeans installed. Better though, install them yourself! Check our lecture and tutorial notes on Blackboard for details.
To begin with, start NetBeans.
Our course adopts JDK 17-19 (including minor update versions) and Apache NetBeans 14/ 15.

2. Under NetBeans, create a New Project [**Java with Ant**] → [Java Application]. Name the project **`ShakeMe`** and put it under the folder **`H:\JAVA_ASG1`** or some other location you prefer. Tick the box "Create Main Class" with name **`exercise.ShakeMe`**. Click Finish.

3. If you haven't ticked the box, *no regret*! You may create a New File [Java] → [Java Class]. Name the class **`ShakeMe`** and put it under the project **`ShakeMe`**. Let the Location be Source Packages and put it in package **`exercise`**. Click Finish.



4. Key in the following program ***with given comment and proper style/ indentation*** *carefully*. Fill in also your own student ID, name and date. Change also the method **`showMyInfo()`**. *Although you may delete some of the default code/ comment generated by NetBeans,* ***you MUST key-in and keep ALL given comment on this assignment specification.***

```java
/**
 * CSCI1130 Java Assignment 1 ShakeMe
 * Aim: Get acquaint with the JDK + NetBeans programming environment
 *      Learn the structure and format of a Java program by example
 *
 * Remark: Key in class names, variable names, method names, etc. AS IS
 *         You should type also ALL the comment lines (text in gray)
 *
 * I declare that the assignment here submitted is original
 * except for source material explicitly acknowledged,
 * and that the same or closely related material has not been
 * previously submitted for another course.
 * I also acknowledge that I am aware of University policy and
 * regulations on honesty in academic work, and of the disciplinary
 * guidelines and procedures applicable to breaches of such
 * policy and regulations, as contained in the website.
 *
 * University Guideline on Academic Honesty:
 *   http://www.cuhk.edu.hk/policy/academichonesty
 * Faculty of Engineering Guidelines to Academic Honesty:
 *   https://www.erg.cuhk.edu.hk/erg/AcademicHonesty
 *
 * Student Name: xxx <fill in yourself>
 * Student ID  : xxx <fill in yourself>
 * Date        : xxx <fill in yourself>
 */

package exercise;

import javax.swing.JFrame;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.util.concurrent.TimeUnit;

/**
 * ShakeMe Game
 * Java Assignment
 * @author Michael FUNG
 * @since  9 September 2022
 */
public class ShakeMe extends JFrame implements ActionListener {

    /** Instance fields indicating size width x height. */
    protected int width, height;
    /** 2D array of JButton object references. */
    protected JButton buttons[][];
    /** Number of YELLOW (turned-on) buttons. */
    protected int counter = 0;

    /** Default constructor. */
    public ShakeMe()
    {
        width = 20;
        height = 10;
        initDisplay();
    }

    /**
     * Constructor with given width and height of the ShakeMe object.
     * @param w is number of boxes left-to-right
     * @param h is number of boxed top-to-bottom
     */
    public ShakeMe(int w, int h)
    {
        width = w;
        height = h;
        initDisplay();
    }
```

```
/** Initialize the ShakeMe window. */
public final void initDisplay() {
    try {
        UIManager.setLookAndFeel(UIManager.getCrossPlatformLookAndFeelClassName());
    } catch (ClassNotFoundException |
             IllegalAccessException |
             InstantiationException |
             UnsupportedLookAndFeelException exceptionObject) {
    }

    setTitle("Java Shake Me");
    setLayout(new GridLayout(height, width));
    buttons = new JButton[height][width];
    for (int row = 0; row < height; row++)
        for (int col = 0; col < width; col++)
        {
            buttons[row][col] = new JButton(row + ", " + col);
            buttons[row][col].setMargin(new Insets(1, 1, 1, 1));
            buttons[row][col].addActionListener(this);
            add(buttons[row][col]);
            if (row == height - 1)
                buttons[row][col].setForeground(Color.RED);
        }
    setSize(width * 40, height * 40);
    setVisible(true);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

/**
 * ActionPerformed on user clicking a target button, try to turn off the target.
 * On clearing all "YELLOW" targets, shake and show Done and Bye
 */
@Override
public void actionPerformed(ActionEvent eventObject) {
    JButton target = (JButton) (eventObject.getSource());
    target.setForeground(Color.BLUE);
    if (!turnOffButton(target))
        // wrong target, shake()!
        shake();
    else if (counter == 0) {
        shake();
        JOptionPane.showMessageDialog(null, "Done!  Bye!");
        shake();
        System.exit(0);
    }
}

/** Slow down this process by sleeping this thread. */
private void delay(long sleepInMS) {
    try {
        TimeUnit.MILLISECONDS.sleep(sleepInMS);
    } catch (InterruptedException exceptionObject) {
        Thread.currentThread().interrupt();
    }
}

/** Shake the window. */
private void shake() {
    Point windowLocation = getLocation();

    double round = 5, max_radius = 10, step = 100;

    double limit = 2 * Math.PI * round;
    double angle_increment = limit / step;
    double radius_increment = max_radius / step;

    for (double angle = 0,   radius = 0;
         angle < limit;
         angle += angle_increment,   radius += radius_increment)
```

```
        {
            setLocation((int) (Math.cos(angle) * radius) + windowLocation.x,
                        (int) (Math.sin(angle) * radius) + windowLocation.y);
            this.delay(6);
        }
        this.setLocation(windowLocation);
    }

    /**
     * Turn on a button and increase the counter, taking two coordinates.
     * @param h is the top-to-bottom coordinate (row index)
     * @param w is the left-to-right coordinate (column index)
     */
    public void turnOnButton(int h, int w) {
        if (Color.YELLOW != buttons[h][w].getBackground()) {
            buttons[h][w].setBackground(Color.YELLOW);
            counter++;
        }
    }

    /**
     * Turn off a button "object" and decrease the counter.
     * @param target is a JButton object reference
     * @return false if the target is NOT on; true after finishing the action
     */
    public boolean turnOffButton(JButton target) {
        if (Color.YELLOW != target.getBackground())
            return false;
        target.setBackground(null);
        counter--;
        return true;
    }

    /**
     * TO DO: students should customize this method.
     * to show the last FIVE digits of your SID in YELLOW in BIG Buttons; AND
     * to show first 10-char of your SURNAME 1-by-1 RED-ON-YELLOW on the bottom
     */
    public void showMyInfo()
    {
        // example digit: 7 in YELLOW
        turnOnButton(1, 4);
        turnOnButton(2, 4);
        turnOnButton(3, 4);
        turnOnButton(4, 4);
        turnOnButton(5, 4);
        turnOnButton(6, 4);
        turnOnButton(7, 4);
        turnOnButton(1, 1);
        turnOnButton(1, 2);
        turnOnButton(1, 3);

        // example surname with 4-char: N A M E
        int c = 0;
        buttons[height - 1][c].setText("N");
        turnOnButton(height - 1, c++);
        buttons[height - 1][c].setText("A");
        turnOnButton(height - 1, c++);
        buttons[height - 1][c].setText("M");
        turnOnButton(height - 1, c++);
        buttons[height - 1][c].setText("E");
        turnOnButton(height - 1, c++);
        // add more lines as you need here
    }

    /**
     * main() method, starting point of the Java application.
     * @param args are command line arguments in a String array
     */
```

```
    public static void main(String[] args) {
        ShakeMe myObj;
        // create a ShakeMe object of different size 30 x 10
        myObj = new ShakeMe(30, 10);
        myObj.showMyInfo();
    }
}
```

5. **Under NetBeans, pick menu [File] → [Project Properties] → Categories [Run]. Browse and pick `exercise.ShakeMe` as [Main Class]. Try toggling some options under Categories [Documenting] too. Click OK to dismiss the Project Properties dialog.**

6. **If you have many opened projects, click menu [Run] → [Set Main Project].**

7. Build the project (press the function key [F11] on the keyboard). If there are errors, don't panic. Double-click on the first error message in the Output window. Check the error, correct it and re-compile. Feel tired? Take a rest.

8. When you finish and there is no more error, you are ready to try out the program.

9. Run the application (press the function key [F6] on the keyboard). Enjoy your work.
   - Try clicking some buttons a few times and observe the effects.
   - You may vary the number of buttons shown to fit your surname; see method **`main()`**.
   - You may design the letter style and the size of each digit/ character of your own.
   - We do NOT expect you to understand the whole given program at this stage. However, you shall be able to figure out the coordinates system and "paint" your SID using button "blocks" and show your surname by following our example statements in the given example method **`showMyInfo()`**.

10. Click Run → Generate JavaDoc and browse the documentation.

Submission:

1. **Locate** your NetBeans project folder, e.g., **`H:\JAVA_ASG1\ShakeMe\`**.

2. NetBeans File Menu → Export Project → To ZIP… and save the ZIPPED Root Project folder **`ShakeMe\`** to a convenient "Build ZIP" location such as Desktop.

3. Submit the file **`ShakeMe.zip`** via our Online Assignment Collection Box on Blackboard **`https://blackboard.cuhk.edu.hk`**

Marking Scheme and Notes:

1. The submitted program should be free of any typing mistakes, compilation errors and warnings.

2. Comment/remark, indentation, style is under assessment in every programming assignments unless specified otherwise. This program gives you an example of a well-formatted source file. Variable naming, proper indentation for code blocks and adequate comments are important.

3. Remember to do your submission before 6:00 p.m. of the due date. No late submission would be accepted.

4. If you submit multiple times, **ONLY** the content and time-stamp of the **latest** one would be counted. You may delete (i.e. take back) your attached file and re-submit. We ONLY take into account the last submission.

University Guideline for Plagiarism

Attention is drawn to University policy and regulations on honesty in academic work, and to the disciplinary guidelines and procedures applicable to breaches of such policy and regulations. Details may be found at http://www.cuhk.edu.hk/policy/academichonesty/. With each assignment, students are required to submit a statement that they are aware of these policies, regulations, guidelines and procedures.

Faculty of Engineering Guidelines to Academic Honesty

MUST read: https://www.erg.cuhk.edu.hk/erg/AcademicHonesty

Things to TRY (but your need NOT submit your funny version!)

1. **Locate** your NetBeans project folder and find a file `ShakeMe.jar`.
   If you are using Oracle JDK, double click it!
   If you are using Open JDK and double-clicking does not work, try run the last command shown in the NetBeans Output window after you Build the project.
   If you can't find it, Build your project once.

2. **Change** the spiral speed and/ or range.

3. **Change** the game rules.

4. **Add** your own cute behaviors.