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: 20 October 2022 (Thu) Assignment 3 Full mark: 100
Expected normal time spent: 5 hours

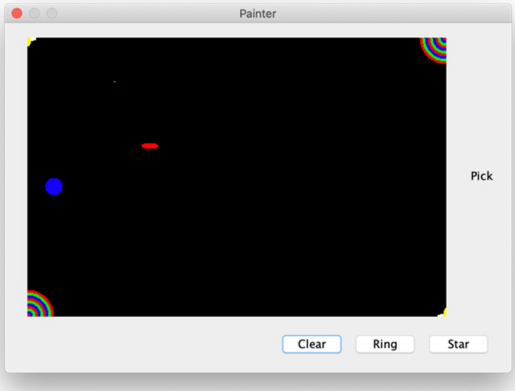
Painter

Aim:
1. declaring a class with constructor as well as instance fields and methods
2. declaring and invoking methods with parameters
3. random number generation
4. looping and branching
5. user interaction with JOptionPane

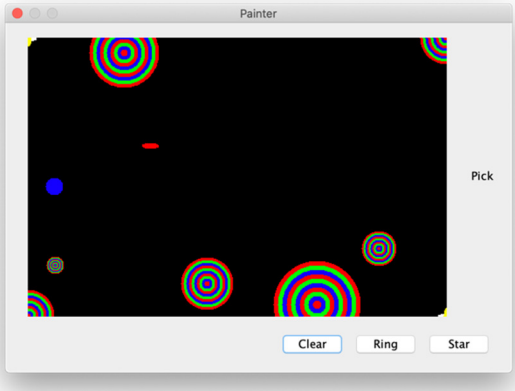
Task: Create a Java Painter program with JOptionPane dialogs and image display.

A program skeleton with sample commented code is given in a provided Java source file **`Painter.java`**. **Follow this specification AND the instructions in the comment**. Here are some screen shots for your reference:

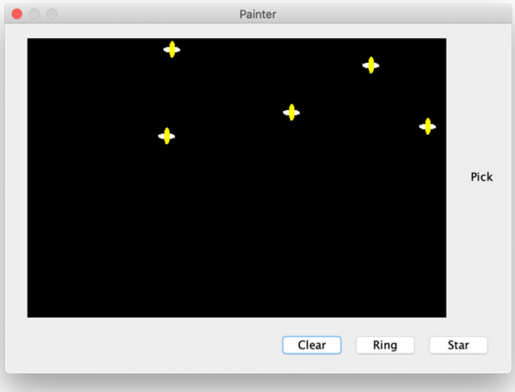
Initial dialog:



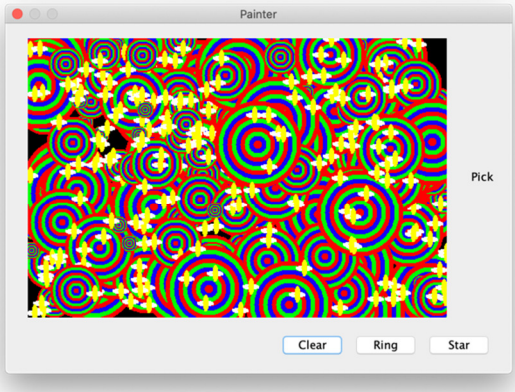
Then user clicked Ring:



User clicked Clear, then clicked Star:



Clicked Ring and Star buttons many times:



Note: the look-and-feel of the application dialog on Windows, macOS and other systems may vary. For example, the order and location of the three buttons Clear, Ring and Star may be different BUT it will NOT affect the functionality of the application.

Key Steps:

1. Create a new class **`Painter`** in a new NetBeans project **`Painter`** with package **`painter`**.

   a) Close your newly created, opened file Painter.java under NetBeans.

   b) Copy the given file Painter.java to replace yours under Windows Explorer/ mac Finder.

   c) Open the file Painter.java again under NetBeans.

   d) It imports/ makes use of a few Java API classes.

2. The initial given program can be compiled and run, however, it lacks many features.

   a) Close dialog button does not work.

   b) Buttons Star, Ring and Clear do not work.

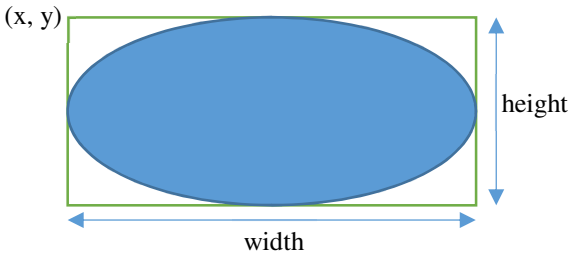
   c) Test patterns at the four corners are missing.

3. Screen/ image coordinates system: X-axis grows from left-to-right; Y-axis grows from top-to-bottom. Origin (0, 0) is at top-left corner. IMAGE_WIDTH and IMAGE_HEIGHT are given constant class fields, so bottom-right corner is at (IMAGE_WIDTH-1, IMAGE_HEIGHT-1).

4. There is a given instance field **`pen`** for drawing. We send the following message for drawing:
   ```
   pen.fillOval(x, y, width, height);
   ```

   a) Parameters x and y are referring to top-left corner of the oval, NOT the center.

   b) Parameters width and height indicate the dimensions of a bounding box of the oval.



5. The program should run in a given forever loop that detects user response repeatedly in a JOptionPane dialog. It delivers corresponding actions, **on top of the existing drawing**:

   a) Response = 0, i.e., Star, paints 5 WHITE-YELLOW cross-stars of dimensions 20 x 20, at random center locations in (0, 0) – (IMAGE_WIDTH-1, IMAGE_HEIGHT-1).

   b) Response = 1, i.e., Ring, paints 5 RGBRGBRGBR-concentric-rings of sizes 1, 2, 3, 4 and 5 respectively, translating to dimensions 20 x 20, 40 x 40, 60 x 60, 80 x 80 and 100 x 100, at random center locations in (0, 0) – (IMAGE_WIDTH-1, IMAGE_HEIGHT-1).

   c) Response = 2, i.e., Clear, clears the drawing in BLACK.

   d) Response = -1 or something else, return/ quit the forever loop and terminate the program.

6. Do NOT modify those given code indicated in the skeleton source file. Finish your work incrementally, method by method. Practise divide-and-conquer.

7. Edit/ (Build): Compile/ Run/ Debug your program. If you do something wrong, don't panic. Double-click on the first error message. Check it, correct it and re-compile. Remember that a single mistake may trigger dozens of error messages. Always tackle the first one first. Be reminded that the error message itself as well as the indicated line number may not be accurate.

8. Thoroughly Test Run your program (press the function key [F6] on the keyboard) with different input data sets such as extreme values and invalid inputs. Enjoy your work.

Submission:

1. Prepare the header comment block properly in your Java source file **`Painter.java`** to include academic honesty declaration and your personal particulars.

2. **Locate** your NetBeans project folder, e.g., **`H:\JAVA_ASG3\Painter`**.

3. ZIP the project folder **`Painter`**. Upload and Submit the file **`Painter.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. Comment/remark, indentation, style are under assessment in every programming assignments unless specified otherwise. Variable naming, proper indentation for code blocks and adequate comments are important.

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

3. 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.