Full Name: _____

Student ID#: _____

# CSE 1325 OBJECT-ORIENTED PROGRAMMING

**PRACTICE 1 -=#=- Exam #2 –=# 1 M #=– Java -=#=- PRACTICE 1**

## Instructions

1. Students are allowed pencils, erasers, and beverage only.
2. All books, bags, backpacks, phones, **smart watches**, and other electronics, etc. must be placed along the walls. **Silence all notifications.**
3. PRINT your name and student ID at the top of this page **and every additional pastel coding sheet**, and verify that you have all pages.
4. **Read every question completely before you start to answer it.** If you have a question, please raise your hand. You may or may not get an answer, but it won't hurt to ask.
5. If you leave the room, you may not return.
6. You are required to SIGN and ABIDE BY the following Honor Pledge for each exam this semester.

## Honor Pledge

On my honor, I pledge that I will not attempt to communicate with another student, view another student's work, or view any unauthorized notes or electronic devices during this exam. I understand that the professor and the CSE 1325 Course Curriculum Committee have zero tolerance for cheating of any kind, and that any violation of this pledge or the University honor code will result in an automatic grade of zero for the semester and referral to the Office of Student Conduct for scholastic dishonesty.

Student Signature: _____

# Vocabulary

Write the word or phrase from the Words list below to the left of the definition that it best matches. Each word or phrase is used at most once, but some will not be used. {15 at 2 points each}

***Vocabulary***

| Word | Definition |
|---|---|
| 1 | An encapsulated bundle of data and code |
| 2 | A special class member that cleans up when an object is deleted (not supported by Java) |
| 3 | A named scope |
| 4 | A method that changes the value of a private variable |
| 5 | A subclass replacing its superclass' implementation of a method |
| 6 | A style of programming focused on the use of classes and class hierarchies |
| 7 | A method that returns the value of a private variable |
| 8 | Reuse and extension of fields and method implementations from another class |
| 9 | A template encapsulating data and code that manipulates it |
| 10 | A subclass or interface inheriting members from two or more superclasses or interfaces |
| 11 | An instance of a class containing a set of encapsulated data and associated methods |
| 12 | A class member variable |
| 13 | A data type that includes a fixed set of constant values called enumerators |
| 14 | A managed memory technique that tracks the number of references to allocated memory, so that the memory can be freed when the count reaches zero |
| 15 | A special class member that creates and initializes an object from the class |

***Word List***

| | | | | |
|---|---|---|---|---|
| Abstraction | Class | Constructor | Destructor | Encapsulation |
| Enumerated type | Field | Garbage Collector | Getter | Inheritance |
| Instance | Method | Multiple Inheritance | Namespace | Object |
| Object-Oriented Programming (OOP) | Operator | Override | Package | Primitive type |
| Reference Counter | Setter | Subclass | Superclass | Variable |

Note: ALL vocabulary words may be on the actual exam!

## Multiple Choice

Read the full question and every possible answer. Choose the one best answer for each question and write the corresponding letter in the blank next to the number. {15 at 2 points each}

1. ____ **To display bold text in JLabel label, write**

   A. `label.bold(); label.setText("I am Groot");`

   B. `label.setText("<html><b>I am Groot</b></html>");`

   C. `label.setText("I am Groot").bold();`

   D. `label.setText("<b>I am Groot</b>");`

2. ____ **The statement**
   `button.addActionListener(event -> System.out.println("Click!\n"));`

   A. displays "Click!" on the button each time it is clicked

   B. will not compile, because no type is specified for `event`

   C. prints "Click!" to the console each time `button` is clicked

   D. displays "Click!" in a dialog box each time `button` is clicked

3. ____ **To respect encapsulation while saving and loading program data,**

   A. create an I/O class in the package and set all fields to package-private

   B. delegate the reading and writing of each class's fields to that class

   C. use `Object.pickle(object)` to automatically save each object's fields to the file

   D. add a `Struct save()` method to each class that returns a struct of its private data

4. ____ **Swing represents a color using**

   A. a 3-byte `int` with the bytes representing red, green, and blue

   B. a 4-byte `int` with the bytes representing red, green, blue, and alpha

   C. a `Color` class

   D. None of these

5. ____ **Which of these statements about Swing is FALSE?**

   A. Swing is implemented using Java classes with multiple inheritance

   B. A Swing program may include both a menu bar and a tool bar in a main window

   C. Swing programs are portable, and run on Linux, Windows, or OS X

   D. All of these statements are true

6. ____ **Which of the following declares a Swing main window class for our application?**

    A. `public class Mainwin extends JMainWindow`

    B. `public JFrame Mainwin`

    C. `public class Mainwin extends JFrame`

    D. `public class Mainwin implements Window`

7. ____ **To print a String on** `JPanel canvas` **with** `Graphics2D g`**, call**

    A. `g.println("Hi", 30, 30)`

    B. `g.drawString("Hi", 30, 30)`

    C. `g.translate("Hola", 30, 30)`

    D. `g.setFont("Hi", 30, 30)`

8. ____ **What is TRUE about painting the Swing JPanel widget?**

    A. To request that paintComponent be called by Swing, our code calls `repaint()`

    B. The `Graphics` context is usually duplicated (created) before modification using its create method

    C. The `paintComponent` method is called by Swing when the `JPanel` widget needs to be redrawn

    D. All of these are true

9. ____ **Which of the following is a Java anonymous class?** `Comparator comp =`

    A. `(String a, String b) -> a.length() - b.length();`

    B. `(a, b) -> a.length() - b.length();`

    C. `new Comparator() {public int compare(String a, String b) {return a.length() - b.l`

    D. None of these are a Java anonymous class

10. ____ **To effectively read or write a file, we need to know**

    A. its path, name, and size

    B. its path and name

    C. its name, access permissions, and inode

    D. its name and file format

11. ____ **To ensure clean handling of I/O errors, use**

    A. nested try / catch as required

    B. try / catch / finally

    C. try-with-resources

    D. ternary-try

12. ____ **The best dialog to enable an artist to select any color for drawing on a** `JPanel` **is**

    A. A menu on the menu bar with simple color names like "Red", "Green", "Blue", and "Orange"

    B. `JOptionPane.showInputDialog`

    C. `JDialog` with a `JSlider` each for red, green, blue, and alpha

    D. `JColorChooser`

13. ____ **To pass the int parameter '3' to JButton b's click observer onButtonClick, write**

    A. `b.addActionListener(event -> onButtonClick(3));`

    B. `b.addActionListener(event -> onButtonClick(), 3);`

    C. `b.addActionListener(3, event -> onButtonClick());`

    D. `b.addActionListener(event -> onButtonClick() -> 3);`

14. ____ **To open** `filename` **for reading, write**

    A. `File f = fopen(filename, 'r');`

    B. `FileReader f = new BufferedReader.createFileReader(filename);`

    C. `File f = filename.open('r');`

    D. `BufferedReader br = new BufferedReader(new FileReader(filename));`

15. ____ **The layout of widgets in a Swing window is determined by**

    A. Absolute coordinates and widget widths measured in pixels

    B. A layout manager object such as `GridBagLayout`

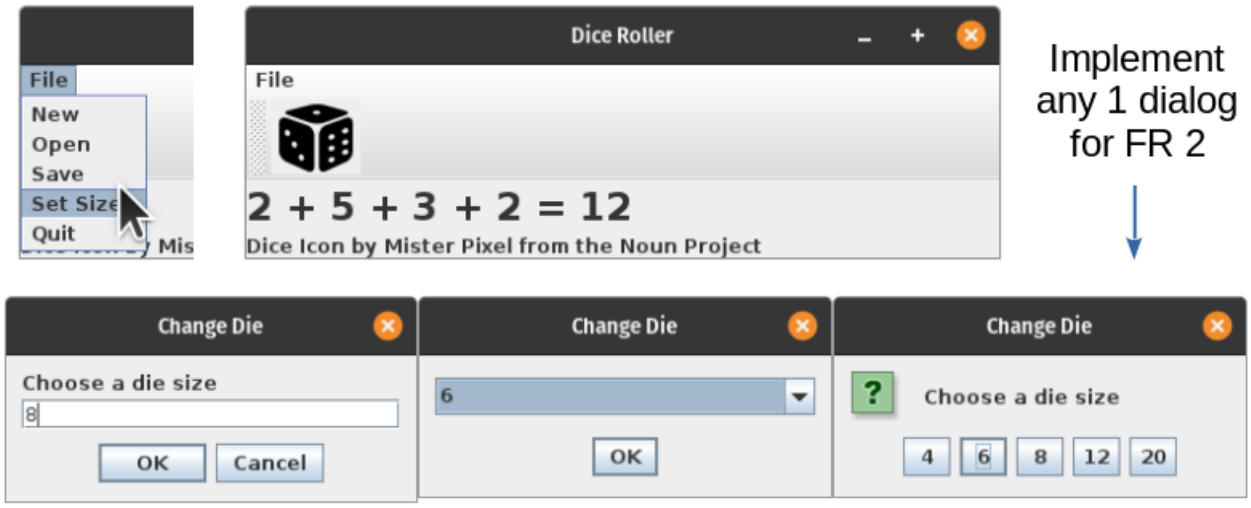
    C. Containers with predefined layout rules such as `Box` or `GridBag`

    D. Callbacks registered with `JWindow` layout handlers

## Free Response

Provide clear, concise answers to each question. Each question implements a small portion of the Java Swing application whose GUI and introductory code is shown below for reference. Each question, however, is *completely independent* of the other questions, and is intended to test your understanding of one aspect of Java programming. **Write only the code that is requested.** You will NOT write the entire application!

The information on class Dice below applies to Free Response questions 1 through 3.




```
// Assume all needed imports

public class Dice extends JFrame {

   private ArrayList<Integer> rolls = new ArrayList<>();
   private int size = 6;

   private JLabel data;
   private JLabel msg;

   public static void main(String[] args) {
       Dice myApp = new Dice("Dice Roller");
       myApp.setVisible(true);
   }

   public Dice(String title) {
       super(title);
       setSize(480,160);

       // Continue writing the constructor as specified below
```

1.a. {2 points} In the Dice constructor, write code to configure the main window Dice to exit if the close button (the 'x' in the top corner of the window) is clicked.

1.b. {4 points} In the Dice constructor, write code to create and add to the main window a menu bar. Show code to add File > Set Size ONLY that calls onSetSizeClick() when selected. (Other menu items should NOT be coded.)

1.c. {4 points} In the Dice constructor, write code to create and add to the main window a toolbar just below the menubar. Show code to add a button with its icon from file "roll.png", with action command and tool tip both "Roll die", and that calls onRollDieClick() when clicked.

1.d. {2 points} In the Dice constructor, write code to create and add to the main window a label in the center section to display data, referenced by field `data` (shown in the code above).

1.e. {2 points} In the Dice constructor, write code to display and execute the main window.

2. {8 points} Write method Dice.onSetSizeClick(). This method should display a dialog allowing the user to select from integers 4, 6, 8, 12, or 20. You may use buttons, a combo box, or a text box for this selection. A JOptionPane static method (recommended) or custom JDialog is acceptable. Assign the selected integer to field `size`. If the user selects an invalid size (for text box only), display an error dialog. If the user selects Cancel or the close button (x), exit the dialog with no error dialog and with no change to `size`.

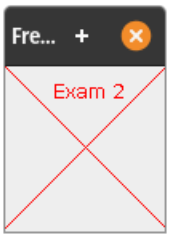
3. {10 points} Write method onOpenClick(). Open file "untitled.dice" for reading, which contains a sequence of previous die rolls, one per line. Read each line, converting the String to an int and adding that to ArrayList rolls. Use try-with-resources to display a message dialog "Unable to read untitled.dice" on an IOException.

4. {8 points} Give class X below, write package-private class Canvas. Canvas should draw a 100x100 pixel red X on the screen, and write "Exam 2" in red starting at (30,20) in 10-point bold sans serif font.

```
// Assume all needed imports

public class X extends JFrame {
   public X() {
       super("Free Response 4");
       setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
       setSize(100,140);
       Canvas canvas = new Canvas();
       add(canvas);
   }

   public static void main(String[] args) {
       X x = new X();
       x.setVisible(true);
   }
}
```




**Bonus:** {up to 4 points} List and briefly describe the two different backlogs used to plan work in a small Scrum project.

**Mark question number for solutions. Additional paper is available on request.**