Homework 2: Answers to Questions

Jade Pearl

CMSC 335-7381 Object-Oriented and Concurrent Programming

Professor Andrew Seely

**Question 1**:

The diagrams defined by the UML Standard are:

1. ***Class Diagram***: Shows the attributes and methods of each class used in a project. Also helps us analyze relationships between classes such as inheritance.
2. ***Composite Structure Diagram***: Describes how structures of a software (e.g. classes, interfaces, etc.) behave in relation to their configuration.
3. ***Object Diagram***: Similar to class diagrams but shows instances of a class and its relationships at certain points in time.
4. ***Component Diagram***: Used to show the relationship between software systems and the external interfaces that they interact with. They are useful for designing/building complex systems according to GeeksForGeeks.
5. ***Deployment Diagram***: Shows the relationships between hardware components that exist and the software components that are run by the hardware components.
6. ***Package Diagram***: Depicts the organization of packages and their elements. The composition of multiple packages are also shown to organize class and use case diagrams (GeeksForGeeks 2017)
7. ***State Machine Diagram***: Or simply state diagram, a behavioral diagram that represents the behavior and conditions of a condition using finite state transitions.
8. ***Activity Diagrams***: can be used to model the flow of activities and control in a system which is often used for analyzing the control in a software process.
9. ***Use Case Diagram***: Describes the functionality of a system by mapping out when/where a system can be used and its interaction with external factors. Does not cover the actual implementation of the system.
10. ***Sequence Diagram***: Describes the order in which objects/elements of a system function.
11. ***Communication Diagram***: In a free form manner, rather then sequential like the sequence diagram, communication diagrams describe the messages exchanged and links between objects and their relationships.
12. ***Timing Diagram***: Another form of sequence diagram that depicts an object’s behavior over a specific time and duration.
13. ***Interaction Overview Diagram***: Can be a mix of sequence and activity diagrams because it involves the sequential actions taken in an exchange and breaks down complex interactions.

Here is an example of a UML diagram. A Class UML diagram:

A screenshot of a computer

Description automatically generated

**Question 2**:

The way that the toString methods in H2ClassA and H2ClassB should be written is as follows:

A screen shot of a computer program

Description automatically generated

It shows the proper output of “4 3 7 5 99 3” and it takes advantage of the natural toString method in H2ClassB. H2ClassA toString builds a string with the contents of the list and appends each H2ClassB item with a space between each number before returning it as a string result. The H2ClassB takes its own x field, which is already an integer type, into a string.

**Question 3**:

A screen shot of a computer program

Description automatically generated

The error that occurs in this code is that there is an undefined default constructor/explicit constructor for H2ClassC, which leads to one of the approaches to fixing this problem.

H2ClassD extends H2ClassC but Java will not properly process any here because the H2ClassD is not calling anything specific and there is no default constructor for Java to use. One way to fix this code is to add a non-argument constructor to H2ClassC; this makes the error disappear.

A screen shot of a computer program

Description automatically generated

For the second approach to this issue, you can leave H2ClassC untouched if desired by adding an explicit constructor in H2ClassD which uses the already defined constructor in the H2ClassC class.

A screen shot of a computer code

Description automatically generated

**Question 4**:

A screen shot of a computer program

Description automatically generated

The error from the above code is due to a compiler error. The constructor H2ClassE(int a) is not properly written as the constructor call “this(5, 12)” is not the first statement in the constructor as it is supposed to be. This is fixed by putting that statement first in the constructor:

A computer screen shot of a program

Description automatically generated

**Question 5**:

The declaration “public static final int myNumber = 17.36” is an incorrect declaration because it is trying to assign an integer to a decimal value which is syntactically incorrect. A way to fix this problem is to make the int into a double variable instead or to round the decimal to an integer by using forced integer conversion on the double number.

1. public static final double myNumber = 17.36;
2. public static final int myNumber = (int) 17.36;

**Question 6**:

In this code:

A screen shot of a computer program

Description automatically generated

The issue is that the default constructor does nothing to initialize x which is a final int variable, meaning it has to be initialized at least once. To fix this, error, you could change from x from being a final variable to just a private variable or assign x to a number, say 0, in the default constructor.

A screen shot of a computer code

Description automatically generatedA screen shot of a computer program

Description automatically generated

**Question 7**:

A screen shot of a computer code

Description automatically generated

This code has an interesting problem. Just like question 6, x should be initialized at least once in the code which one option is to make a default constructor where x is assigned:

A screen shot of a computer program

Description automatically generated

That option would be used if the programmer wanted to keep the int H2ClassH() method the way it was. X could have also been switched to a public int instead of a final. Lastly, the method int H2ClassH() could entirely be changed into a new method or a constructor:

A screen shot of a computer program

Description automatically generated

OR

A screen shot of a computer program

Description automatically generated

**Question 8**:

A screen shot of a computer code

Description automatically generated

The above code has an issue with x not being initialized correctly. The final variable cannot be assigned as h.x = 24. One way to fix this is to assign 24 to x in a default constructor and get rid of the h.x = 24 statement. I added a print statement to ensure the correct output.

A screen shot of a computer program

Description automatically generated

The other way I did was without a default constructor and just assign x to 24 in the class member variable declaration statement:

A screen shot of a computer program

Description automatically generated

**Question 9**:

**A screen shot of a computer program

Description automatically generated**

The given code misuses the MouseListener class when programming this scenario because the constructor tries to add a new MouseListener with another MouseListener.

The first way to fix this is to change MouseListener to the MouseAdapter in the addMouseListener (new MouseListener() … line:

A computer screen shot of text

Description automatically generated

The other method is to keep the MouseListener but to provide all the methods needed for MouseListener:

A screen shot of a computer program

Description automatically generated

**Question 10**:

A computer screen with text and images

Description automatically generated

The above FX GUI code incorrectly imports for JavaFX. It is trying to import a JavaFX library that is no longer supported and does not exist. The programmer needs to import the proper extensions/libraries that are currently used such as the javafx.stage.\*, javafx.scene.\*, javafx.application.\*, and javafx.event.\*. After that is implemented, the class H2ClassK needs to extend the Application class of the JavaFX library and use the overriden Application’s start method. The programmer also needs to set up buttons, event handlers, etc to make the program complete. After finishing the start method, they need to make a main method that will launch the program. The start method will consist of the label that needs to be added, the a button to control the label, a scene, a primaryStage, and some kind of pane to set the scene. Here is an example going off of this code that I implemented:

A screen shot of a computer program

Description automatically generated

A white background with black text

Description automatically generatedA screenshot of a computer

Description automatically generated

**Resources**:

GeeksforGeeks. (2017, October 27). *Unified Modeling Language (UML) | An Introduction - GeeksforGeeks*. GeeksforGeeks. <https://www.geeksforgeeks.org/unified-modeling-language-uml-introduction/>

*MouseAdapter (Java Platform SE 8 )*. (n.d.). Docs.oracle.com. Retrieved November 7, 2023, from <https://docs.oracle.com/javase/8/docs/api/java/awt/event/MouseAdapter.html>

*MouseListener (Java Platform SE 8 )*. (n.d.). Docs.oracle.com. Retrieved November 7, 2023,

from <https://docs.oracle.com/javase/8/docs/api/java/awt/event/MouseListener.html>

Pearl, J. (2023). *UML diagram Example* [Mermaid Chart Creator *UML diagram Example*].