CSE 1720

Lecture 24

Review and Recap

High-Level Overview of the Course

L1-L6: Aggregation
- Ch8 (JBA)
- Lecture examples (to demonstrate Graphics2D class)

L7-L10: Inheritance, use `Set` and `List` from Collections Framework
- Ch9 (JBA)

L11-12: Exceptions
- Ch11 (JBA)

L13: Review and Recap for Midterm

L14: Midterm

L15-L16: Observer Pattern, Event Dispatching, Component Redrawing

L17: guest lecture

L18-L21: Model View Controller

L22: In-class quiz, evaluations, Final Exam Discussion

L23: use `Map` from Collections Framework

L24: Recap and Review

I. Aggregation

- when talking about aggregation, are we talking about objects or class definitions?
- If aggregation is a relationship, what are the entities that are participating in this relationship? **Use correct terminology**.
- what are some concrete examples of classes that define aggregations?
- which of these examples are *collections*?
- what distinguishes a *collection* as a special type of *aggregation*?
- What is a collection?
  - Describe in *functional* terms. It is a *thing* that….

I. Aggregation

- To what do the terms *alias*, *shallow copy* and *deep copy* apply?
  - In what context does it make sense to apply these terms?
  - What is the difference between an alias, a shallow copy and a deep copy?
- What is the relationship, if any, between `Graphics2D` and `Rectangle2D`?
- Is `Rectangle2D` an aggregate? Why or why not?
- Is `Graphics2D` an aggregate? Why or why not?
- Composition is a special type of aggregation. What makes it special?

I. Collections

- How do collections support iteration? What is an iterator?
- What is a set, a list, and a map? What are the basic operations on each?
- Given a design scenario and the task of saying whether to use a set, a list or map, how do you proceed?

II. Inheritance

- what relationship exists between an arbitrary subclass object and an arbitrary parent object?
  - do they have the same state? Explain
  - do they provide the same services? Explain
- Does inheritance apply to objects, class definitions, or both? Explain.
- In Java, are all classes subclasses? what is the root of Java's inheritance hierarchy?
- What do all objects inherit?

II. Inheritance

- what is the substitutability principle? when does it get applied? (3 example scenarios)
- what is early binding? what is late binding?
- when is the invocation signature established? early or late binding?
- what is meant by polymorphism?

II. Inheritance

- why do we have abstract classes? As clients, how can we make use of services provided by abstract classes?
- why do we have interfaces? As clients, how can we make use of services provided by interfaces?
- what is meant by a generic? What are some examples of generic classes? What are some special characteristics?

III. Exceptions

- what is a error (for a program, for services)? what is the specification? where do we find it?
- what are the sources of error?
- explain whether an exception signifies that an error has occurred or not.
- Are exceptions part of the precondition, postcondition, or neither?
- are exceptions objects? if so, what services do they offer?

III. Exceptions

- what are the language constructs for dealing with exception?
- what are the rules that come into play when dealing with exceptions?
- what is the difference between checked and unchecked exceptions?
- what are some examples of exceptions?
  - for these examples, which services may potentially throw them?
- are exceptions thrown only when using services?

IV. Observer Pattern, Event Dispatching, Component Redrawing

- What is the observer pattern?
- What are two demonstrations of this pattern in the MVC framework (e.g., in the app in L20_pkg)
- What is the Event Dispatching Thread (EDT)?
- How do we place a process on the EDT?
- In the context of GUI programming, how does a window get updated (e.g.,in response to a window resizing)

V. Model View Controller (MVC)

- What is the purpose of MVC? When is it used?
- What is meant by: a model? By a view? By a controller?
- What is the purpose of each of these components?
- How does the view get updated? (in response to a user input action)
- What is an example of a model? For this example, what is an example of an operation that the user might perform that would cause the model to change?
- How does the controller modify the model?
- Is it possible to have two views of the same model?
- Does the model "know" what views exist of it?

The Final Exam

- Final Lab Exam, 15%
  - 85 minutes
- Final Written Exam, 15%
  - 85 minutes

WED APR 4th, 2012

- Family Names: A-M
  - Lab Exam, CSE 1002, 9-10:25am
  - Written Exam, LAS B (formerly “CSE B”), 10:30-12pm
- Family Names: N-Z
  - Written Exam, LAS B (formerly “CSE B”), 9-10:25am
  - Lab Exam, CSE 1002, 10:30-12pm

- The location info is provided as a convenience to you. It is your responsibility to verify the location using the official source: https://w2prod.sis.yorku.ca/Apps/WebObjects/cdm.woa/wa/curexam

Final Written Exam

- 15% Aggregation concepts (Ch8)
- 15% Collections and Generics (Ch10)
- 15% Inheritance concepts (Ch9)
- 15% Exceptions (Ch11)
- 15% TBD
- 25% Event-Based, Model-View-Controller

Final Lab Exam

- Will be a version of labtest 5
- One question will require you to use a map