COMP2511

Object-Oriented Design and Programming

Generic Types and Polymorphism

Wayne Wobcke

w.wobcke@unsw.edu.au

Today's Lecture

- UML Sequence Diagrams
- Java Type System
  - Abstract Classes and Interfaces
  - Arrays and Lists
- Generic Types
  - Set<E>

UML Sequence Diagram

- Mainly for tricky interactions
  - Authentication, payments
- Validate use cases and walkthroughs

Walkthrough

1. EnrolSystem asks Course for list
2. User selects course
3. Offering asks Course for prereqs
4. Offering asks Student if passed
5. Offering asks Session for free list
6. User chooses Session
7. EnrolSystem asks Session to enrol student
8. EnrolSystem creates Enrolment

Java Type System

- Everything is an Object . . .
  - Except int, float, double, etc., and null
- Assignment b = a is valid if actual type of a is a subtype of declared type of b
- Parameters are references and are called by value

Abstract Classes and Interfaces

- Abstract class cannot have instances
  - e.g. Shape, Car, Fruit
  - Concrete subclasses have instances
- Interface defines functionality (methods)
  - e.g. Cloneable, List<E>
  - Can declare variables as interface types

Arrays and Lists

- Arrays covariant (if $s \le t$ then $s[] \le t[]$)
  - Fruit[] f = new Apple[10] – OK
  - f[0] = new Orange() – ArrayStoreException
- Lists not covariant
  - ArrayList<Fruit> f; ArrayList<Apple> a;
  - Neither f = a nor a = f compiles
  - f.add(new Orange())?

Generic Types

- Types with parameter E, T, etc.
  - Compiler determines actual type
- Java static type checking, but . . .
  - Compiler cannot guarantee type safety
- Implemented by Type Erasure
  - Runtime access only to “raw” types
- Much more complicated than here

Assignment 1

- Design before coding – follow the object-oriented design process
- Submit a UML design document, not automatically generated
- UML class diagram should include fields and methods
- UML class diagram should conform to code
- Don't look for solutions on github, pastebin or rent-a-coder
- Only use Stack Overflow with caution: many explanations wrong
- Make sure you read from a file args[0], write to System.out
- Make sure main() is in CinemaBookingSystem.java
- Don't use a package – use only **the** default package
- Make sure the submitted files include java files and a pdf file

Next Week

- Graphs and Basic Search
- Design Patterns