

**CMSC132  
Fall 2005  
Midterm #1**

**Grader Use Only:**

#1		(20)
#2		(20)
#3		(20)
#4		(25)
#5		(15)
<b>Total</b>		<b>(100)</b>

**First Name:** \_\_\_\_\_

**Last Name:** \_\_\_\_\_

**Student ID:** \_\_\_\_\_

**Section time** \_\_\_\_\_ **TA:** \_\_\_\_\_

**I pledge on my honor that I have not given or received any unauthorized assistance on this examination.**

**Your signature:** \_\_\_\_\_

**General Rules (Read):**

- This exam is closed-book and closed-notes.
- If you have a question, please raise your hand.
- Total point value is 100 points.
- Answer essay questions concisely using 1 or 2 sentences. Longer answers are not necessary and are discouraged.
- **WRITE NEATLY.** If we cannot understand your answer, we will not grade it (i.e., 0 credit).

### **Problem 1 Software Development & Testing (20 pts)**

- a. (4 pts) What is the main reason many software projects fail?
  - a. Poorly trained programmers
  - b. Insufficient project funding
  - c. Complexity of projects
  - d. Slow computers
  - e. Insufficient computer memory
  
- b. (4 pts) What is the software life cycle?
  
- c. (4 pts) What is the first phase of the software life cycle?
  - a. Testing
  - b. Coding
  - c. Design
  - d. Specification
  - e. Documentation
  
- d. (4 pts) True or False  
According to the waterfall model...
  - a. Design all algorithms before coding
  - b. Write test cases before coding
  - c. Use prototype implementation to refine design
  
- e. (4 pts) True or False  
According to the unified model...
  - a. Design all algorithms before coding
  - b. Write test cases before coding
  - c. Use prototype implementation to refine design
  
- f. (4 pts) True or False  
Compared to program verification, empirical testing...
  - a. Handles larger programs
  - b. Always catches more errors
  - c. Ensures code is correct
  - d. Can be applied without examining code
  
- g. (4 pts) True or False
  - a. Black box testing requires good programmers
  - b. Code coverage is a measure of code testing
  - c. Pre-conditions and post-conditions are used for empirical testing

## **Problem 2 Object-Oriented Design (20 pts)**

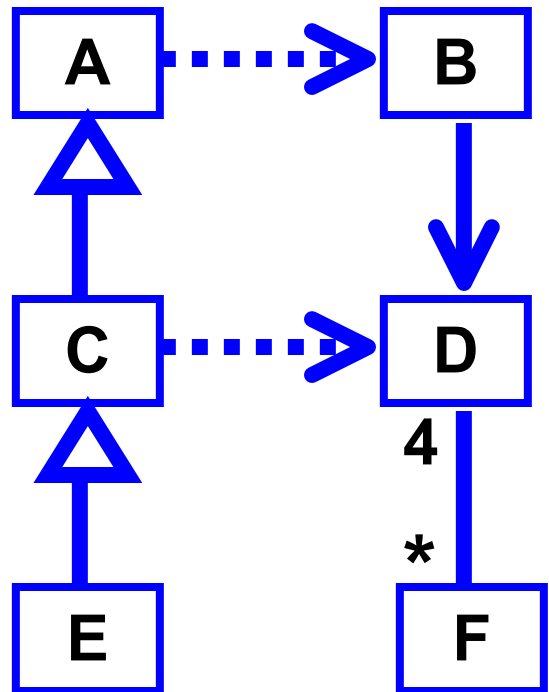
- h. (4 pts) State and behavior are two main qualities of objects in an object-oriented system.
  - a. What is the third quality?
  - b. What is it used for?
  - c. What is an example of its use in Java?
  
- i. (4 pts) True or False  
Object oriented design...
  - a. Produces faster programs
  - b. Produces smaller programs
  - c. Produces software without errors
  
- j. (4 pts) Abstraction and encapsulation are two principles of object-oriented design
  - a. Define abstraction
  - b. Define encapsulation
  - c. Describe how object-oriented design supports encapsulation
  
- k. (4 pts) Given the following problem description, produce an object-oriented solution. Include as many details as possible. Draw a UML class diagram (you may write code for Java classes ONLY if you don't know UML).

Design a simulation of a basketball conference. Each conference has 10 teams. Each team has 12 players. Each player has a specific height, speed, and accuracy. Players know which team they belong to. Some players are scholarship players. Scholarship players need to record their current grade-point average. Players may be transferred between teams. Teams play basketball games against other teams in the conference. The result of each game is determined using a function based on the height, strength, speed, and accuracy of the players on each team.

**Problem 3 Unified Modeling Language (20 pts)**

- l. (4 pts) Consider UML
  - a. What are class diagrams used for?
  - b. What is an association?
  - c. What is a dependency?
  
- m. (4 pts) Given the following Java code, draw a UML class diagram (you may write code for Java classes ONLY if you don't know UML).

```
public class Propeller {  
    public double thrust;  
    public int mileage;  
}  
public class Engine {  
    public double power;  
    public int mileage;  
}  
public class Plane {  
    public Propeller[] myPropellers;  
    public Engine myEngine;  
}  
public class Pilot {  
    public int flightHours;  
    public void fly(Plane p) {  
        ...  
    }  
}  
public class FighterPilot extends Pilot {  
    public int rank;  
}
```



- n. (4 pts) Consider the UML diagram on the right:
  - a. Which class contains class D?
  - b. Which class uses class D?
  - c. Which class may change if class D changes?
  - d. How many instances of class D does class F have?
  - e. Can class A be used wherever Class C is used?
  - f. Can class E be used wherever Class C is used?

#### **Problem 4 Java Programming (25 pts)**

- o. (4 pts) True or False
  - a. Using “==” and .equals() always return the same result
  - b. Variables of type Integer and int are both references
  - c. Autoboxing creates an Integer object from an int
  - d. Exceptions are used to capture run-time errors
  
- p. (4 pts) Write Java code for a Card class
  - a. Use an enumerated type for the suits in a card deck (Spades, Hearts, Diamonds, Clubs)
  - b. Implement the comparable interface for Card objects so that the suits are ranked in the order listed (Spades > Hearts > Diamonds > Clubs)
  
- q. (4 pts) Write Java code for a Deck class
  - a. Uses an ArrayList to store multiple Card objects
  - b. Use an anonymous inner class to generate an Iterator over Card objects in the Deck

#### **Problem 5 Graphic User Interfaces (15 pts)**

- r. (4 pts) In a graphics user interface
  - a. What is the model?
  - b. What is the view?
  - c. What is the controller?
  - d. Why should these be kept separate?
  
- s. (4 pts) Event driven programming
  - a. What are events?
  - b. Why use events for GUIs?
  - c. How are events handled in Java Swing?
  
- t. (4 pts) Given Java Swing code for a GUI, understand the different roles of each part