

# COSC 121: Computer Programming II

## Practice Midterm Examination 1

*This is a practice exam for Midterm1. While the structure is similar to the actual exam, the length of each section might be a different. The questions themselves will also be different (obviously). As mentioned before, in order to prepare for this exam, you should be able to solve all questions from the lectures and labs on your own. No questions are more important than others.*

/5

### PART 1. Multiple Choice Questions (Circle correct answer)

(5 marks)

Write your choice (A, B, C, etc.) for each question in this table →

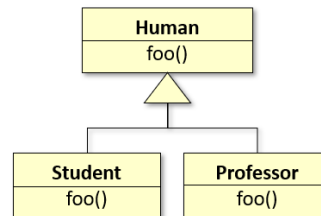
Question #	1	2	3	4	5
Your Choice					

Q1. Which of the following is not inherited?

- A. Instance variables
- B. Constructors
- C. Methods
- D. Both (B) and (C)
- E. None of the above

Q2. In order for the code below to be valid, what must the type of the reference variable **x** be?

```
????? x;  
x = new Student();  
x.foo();  
x = new Professor();  
x.foo();
```



- A. Human
- B. Student
- C. Professor
- D. Something else

Q3. Which class cannot be extended?

- A. superclass
- B. subclass
- C. final class
- D. Object class
- E. abstract class

Q4. Which of these is true about Java Interfaces?

- A. All interfaces share a single root, the Object interface
- B. Interfaces can have constructors, abstract/concrete methods and instance variables
- C. An interface can extend any number of interfaces
- D. An interface is a class

Q5. Which of these declarations is valid?

- A. class WorkingStudent extends Employee, Student implements Comparable
- B. class WorkingStudent extends Comparable
- C. class WorkingStudent extends Employee implements Student
- D. class WorkingStudent implements Comparable, Serializable

/ 10

## Part2. Code Analysis

(10 marks)

Q6a. [+4] The program below has problems, 1 problem per method or constructor. Identify these problems, explain them, then fix them.

```
public class Circle {
    private double radius;
    public Circle(double radius) {
        radius = radius;
    }
    public getArea() {
        return radius * radius * Math.PI;
    }
}

class B extends Circle {
    private double length;
    B(double radius, double length) {
        Circle(radius);
        this.length = length;
    }
    public double getArea() {
        return getArea() * length;
    }
}
```

Q6b. [+3] What is the output of the following program? Justify your answer (you must trace the code and list the order of execution of statements and constructors).

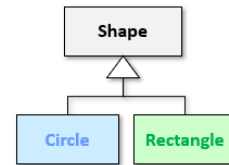
```
public class Main {
    public static void main(String[] args) {
        Furniture a = new Desk();
    }
}

public class Desk extends Furniture {
    public Desk() {
        this(1);
        System.out.print("A");
    }
    public Desk(int t) {
        System.out.print("B");
    }
}

public class Furniture {
    public Furniture() {
        System.out.print("C");
    }
}
```

Q6c. [+3] Assume that shape, circle and rectangle objects are created as follows:

```
Shape s = new Shape();  
Shape c = new Circle();  
Shape r = new Rectangle();
```



Are the following Boolean expressions true or false?

1. (s instanceof Circle)
2. (c instanceof Circle)
3. (r instanceof Circle)
4. (s instanceof Shape)
5. (c instanceof Shape)
6. (r instanceof Shape)

/ 10

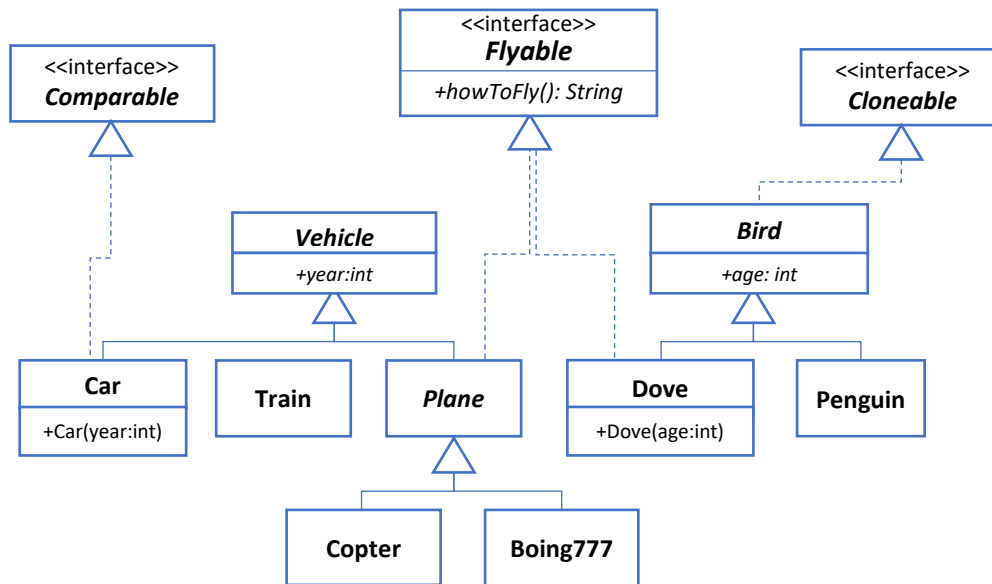
### Part3: Programming

(10 marks)

*Note: the question below is taken from exercise E03. The aim is to help you practice for the exam. I will ask you to model a class structure, but it will be different from this (e.g. include more/less interfaces, different method requirements, etc). The exam may or may not describe the class hierarchy using UML.*

Write Java code to implement the UML diagram below. Note the following:

- a) All italic classes are abstract.
- b) No need to include any method except those you have to include.
- c) The `howToFly()` method should print one line describing how an object flies. For example, a Copter will *"fly off the ground vertically using rotor blades."*
- d) Don't implement `howToFly()` in the Plane class (as it will make little sense because subclasses fly differently). Instead, implement `howToFly()` in both Copter and Boeing777.
- e) Cars are comparable based on their manufacturing year.
- f) You don't have to write the body of all classes (e.g. Train would be an empty class), but feel free to adjust the design and add any class members of your choice.



**Solution:**

**Part1**

Question #	1	2	3	4	5
Your Choice	B	A	C	C	D

**Q6-A** The fixed code is below. You should also explain these errors (if you are not sure what is happening here, put the code in eclipse and see the note about the errors).

```

public class Circle {
    private double radius;
    public Circle(double radius) {
        this.radius = radius;
    }
    public double getArea() {
        return radius * radius * Math.PI;
    }
}

class B extends Circle {
    private double length;
    B(double radius, double length) {
        super(radius);
        this.length = length;
    }
    public double getArea() {
        return super.getArea() * length;
    }
}

```

**Q6-B**

**Output:** CBA

**Tracing:**

- main() calls constructor Desk()
- Desk() calls Desk(int t)

- Desk(int t) calls Furniture()
- Furniture prints C then returns to Desk(int t)
- Desk(int t) prints B then returns to Desk()
- Desk prints A

*Q6(c) – try it in eclipse*

*Part3:*

*see solutions to exercise E03*