

Summer 2010 - COMP 1005/1405
Introduction to Object-Oriented Programming
Test #2

Fill in the blank (10 marks)

1. The ability to use a superclass method in a subclass is due to Java's **inheritance** mechanism.
2. A method that lacks a body is an **abstract** method.
3. An **interface** is like a class except that it contains only instance methods, no instance variables.
4. An object can refer to the class that it inherits from by using the **super** keyword.
5. One cannot create instances of **abstract** classes.

Multiple Choice (20 marks)

1. Suppose `s` is a string with the value "java". What will be assigned to `x` if you execute the following code?

```
1 String s = "java";  
2 char x = s.charAt(4);
```

- (a) 'a'
(b) 'v'
(c) **Nothing will be assigned to x, because the execution causes the runtime error `StringIndexOutOfBoundsException`.**
2. What is `Math.floor(3.6)`?
- (a) **3.0**
(b) 3
(c) 4
(d) 5.0
3. What is `Math.ceil(3.6)`?
- (a) 3.0
(b) 3
(c) **4.0**
(d) 5.0
4. Which of the following statements are true? (select all that apply).
- (a) **You use the keyword `throws` to declare exceptions in the method heading.**
(b) **A method may declare to throw multiple exceptions.**
(c) **To throw an exception, use the keyword `throw`.**
(d) **If a checked exception occurs in a method, it must be either caught or declared to be thrown from the method.**

5. Which method do you use to find the number of elements in an ArrayList named x?

- (a) x.length()
- (b) x.count()
- (c) x.counts()
- (d) **x.size()**
- (e) x.sizes()

6. Which method do you use to test if an element is in an ArrayList named x?

- (a) x.in(element)
- (b) x.contain(element)
- (c) **x.contains(element)**
- (d) x.include(element)

7. How many times will the following code print "Welcome to Java"?

```
1 int count = 0;
2 while (count < 10) {
3     System.out.println("Welcome to Java");
4     count++;
5 }
```

- (a) 8
- (b) 9
- (c) **10**
- (d) 11
- (e) 0

8. What is i after the following for loop?

```
1 int y = 0;
2 for (int i = 0; i<10; ++i) {
3     y += i;
4 }
```

- (a) 9
- (b) 10
- (c) 11
- (d) **undefined**

9. What is the output of running class Test?

```
1 public class Test {
2     public static void main(String[] args) {
3         new Circle9();
4     }
5 }
6
7 public abstract class GeometricObject {
8     GeometricObject() {
9         System.out.print("A");
10    }
11
12    GeometricObject(String color, boolean filled) {
13        System.out.print("B");
14    }
15 }
16
17 public class Circle9 extends GeometricObject {
18     Circle9() {
19         this(1.0);
20         System.out.print("C");
21     }
22
23     Circle9(double radius) {
24         this(radius, "white", false);
25         System.out.print("D");
26     }
27
28     Circle9(double radius, String color, boolean filled) {
29         super(color, filled);
30         System.out.print("E");
31     }
32 }
```

- (a) ABCD
- (b) BACD
- (c) CBAE
- (d) AEDC
- (e) **BEDC**

10. Which of the following is a correct interface ?

- (a) interface A { void print() { }; }
- (b) abstract interface A { print(); }
- (c) abstract interface A { abstract void print() { };}
- (d) **interface A { void print();}**

Programming Questions (70 marks)

1. Write a method void printTriangleOfStars(int n). Use loops. (Note: do not hardcode for the value of 3 or 5).

(20 marks)

TrianglePrinter.java file:

```
1      class TrianglePrinter {
2          public void printTriangleOfStars(int n) {
3              for(int i=1; i <= n; i++) {
4                  for(int j=1; j<=i; j++) {
5                      System.out.print("*");
6                  }
7                  System.out.println();
8              }
9          }
10     }
```

TriangleTester.java file:

```
1      class TriangleTester {
2          public static void main(String args[]) {
3              TrianglePrinter t = new TrianglePrinter();
4              System.out.println("Printing with n = 3");
5              t.printTriangleOfStars(3);
6              System.out.println("Printing with n = 5");
7              t.printTriangleOfStars(5);
8          }
9      }
```

the test code above will output:

```
1      Printing with n = 3
2          *
3          **
4          ***
5      Printing with n = 5
6          *
7          **
8          ***
9          ****
10         *****
```

2. A friend asks you to help him finish the code below. Override the calculateArea method in the subclasses with the appropriate implementation. Use the Math library functions and constants as much as you can. (Note: $\text{areaOfCircle} = \text{PI} * \text{radius} * \text{radius}$, $\text{areaOfSquare} = \text{sideLength} * \text{sideLength}$)

(20 marks)

Shape.java file:

```
1 abstract class Shape {
2
3     abstract double calculateArea();
4 }
```

Circle.java file:

```
1 import java.lang.Math;
2 class Circle extends Shape {
3     double radius;
4
5     Circle(double radius) {
6         this.radius = radius;
7     }
8
9     // your code here
10    public double calculateArea() {
11        return Math.PI * Math.pow(radius,2);
12    }
13 }
```

Square.java file:

```
1 import java.lang.Math;
2
3 class Square extends Shape {
4     double sideLength;
5
6     Square(double sideLength) {
7         this.sideLength = sideLength;
8     }
9
10    // your code here
11    public double calculateArea() {
12        return Math.pow(sideLength,2);
13    }
14
15 }
```

3. Given the following code write the implementation of the `findLongestBook` and `findByAuthor` methods.

- `findLongestBook` returns the with the longest book in the library (one with the most pages).
- `findLongestBook` returns `null` if there are no books in the library.
- `findByAuthor` returns all the books in the library that have been written by a given author.
- `findByAuthor` returns an empty `ArrayList` if there are no books in the library that have been written by the given author.

(30 marks)

```
1 class Book {
2     String title;
3     String author;
4     int numPages;
5
6     public Book(String title, String author, int numPages) {
7         this.title = title;
8         this.author = author;
9         this.numPages = numPages;
10    }
11 }
```

```
1 import java.util.*;
2
3 class Library {
4     ArrayList<Book> books;
5
6     public Library(ArrayList<Book> libraryBooks) {
7         if (libraryBooks == null) { this.books = new ArrayList<Book>();
8         } else { this.books = libraryBooks; }
9     }
10    public Book findLongestBook() {
11        if (this.books.size() == 0) { return null; }
12        Book longestBook = books.get(0);
13        for(Book book : books) {
14            if (longestBook.numPages < book.numPages) { longestBook = book; }
15        }
16        return longestBook;
17    }
18
19    public ArrayList<Book> findByAuthor(String authorName) {
20        ArrayList<Book> foundBooks = new ArrayList<Book>();
21        for(Book book: books) {
22            if (book.author.equals(authorName) { foundBooks.add(book); }
23        }
24        return foundBooks;
25    }
26 }
```

Bonus (2 marks)

1. What is the most popular sport in the world? **Soccer/Football**