

COMP 268 – Introduction to Programming in Java

Midterm Quiz

Maximum 100 marks. Minimum 50 marks to pass the exam. Take home exam. Each question carries 10 marks. Answer at least any 10 questions.

1. Write a program to determine if a given substring exists within a larger string. Both inputs should be entered through console.
2. Given:

```
public class Boxer1 {  
    Integer i;  
    int x;  
    public Boxer1(int y) {  
        x=i+y;  
        System.out.println(x);  
    }  
    public static void main(String[] args) {  
        new Boxer1(new Integer(4));  
    }  
}
```

What is the result? Complete the program to confirm the results.

- A. The value "4" is printed at the command line.
 - B. Compilation fails because of an error in line 5.
 - C. Compilation fails because of an error in line 9.
 - D. A NullPointerException occurs at runtime.
 - E. A NumberFormatException occurs at runtime.
 - F. An IllegalStateException occurs at runtime.
3. A news magazine wants to conduct a survey across the four research universities in Alberta. For each university, the magazine wants to gather the number of departments, the number of students

doing research in each department, courses taken by each research student, and marks scored by these students in courses. Design, develop, and implement a Class to capture this information.

4. Under “Interfaces and Inheritance in Java” in Unit 4 you would have studied about a concept called Polymorphism. How does the Bicycle-MountainBike-RoadBike example shown under the link [“http://docs.oracle.com/javase/tutorial/java/landl/polymorphism.html”](http://docs.oracle.com/javase/tutorial/java/landl/polymorphism.html) exemplifies polymorphism. Extend the MountainBike class to create another class called DownhillMountainBike. Downhill bikes are a specialised type of mountain bike with a very strong frame, much stronger than a typical mountain bike. Create the DownhillMountainBike class with getFrameStrength(), setFrameStrength(int strength), and printDescription() methods. The frame strength is to be a value in between 1 and 10, inclusive. Test the DownhillMountainBike class using the TestBikes class from the same web link provided above.
5. Distinguish between buffered and unbuffered I/O in Java with example programs. Explain the difference between the two.
6. Implement a Class called ‘Electric Car’. Identify and explain its key States and Behaviour. Also test the functionality of the class using a wrapper class.
7. Describe a Class called ‘People’. Identify and explain some of its key States and Behaviour. Discuss if it is appropriate to embed all States and all Behaviour about people in a single Class? Implement the People Class in Java and test its functionality.
8. What are the different types of inheritance? Create a hierarchy of Classes for a top Class called “Computer”. Explain various States and Behaviour inherited between Classes in this hierarchy. Implement the Class Computer in Java.
9. Write an Interface named ‘Java Programming Language’ and discuss its implications in the context of different types of Operating Systems and Hardware. Create a Class in Java to implement this Interface.
10. Write a program to encode and decode a string using StringTokenizer. The give text will be “Java Programming”. When u find letter ‘a’ in the string, it should be replaced by @. When you find a space character, it should be replaced by the character ~, 3 times. So, the output of encode will be J@v@~~~Progr@mming. Decode should return the original string “Java Programming” given the encoded string as input. Test the program with various inputs.

11. What are bitwise operators? Explain with an example. Write a program to print 1's complement and 2's complement of a binary number.

12. Write a program to print the first 20 number of the Fibonacci series using recursion.

13. Write a program to print a triangle as shown below, given an input number. This triangle is printed when the input is 3. There are 5 spaces between elements in the triangle.

```
      1
    2  0  2
  3  0  0  0  3
```

14. The coin-counting program in the study guide targets Canadian money. Modify and implement the program to count coins from a different country of your choice outside of North America.

15. The following code snippets, each, find all substrings within a given string. That is, given a string of length 'n' each of the following code snippets will return ' $n * (n + 1) / 2$ ' substrings. Walk through the code snippets by commenting each line of the code and explain how each snippet differs from the rest in computing substrings.

```
a) String str = "12345";
   ArrayList<String> subs = new ArrayList<String>();
   while (str.length() > 0){
       subs.add(str);
       for (int i=1;i<str.length();i++) {
           subs.add(str.substring(i));
           subs.add(str.substring(0,i));
       }
       str = str.substring(1, Math.max(str.length()-1, 1));
   }
```

```
b) String str = "12345";
   ArrayList<string> al = new ArrayList<string>();
   for(int i = 1; i < str.length(); i++){
       for(int k = 0; k < str.length(); k++){
```

```

        if(i + k > str.length()) break;
        al.add(str.substring(k, k + i));
    }
}

```

c) `String str = "12345";`
`ArrayList<string> al = new ArrayList<string>();`
`for(int i = 1; i < str.length() + 1; i ++)`
`{`
 `for(int k = 0; k < str.length(); k++)`
 `{`
 `int a = k;`
 `int b = k + i;`
 `int c = str.length() - 1 - k - i;`
 `int d = str.length() - 1 - k;`
 `al.add(str.substring(a,b));`
 `if(a < c)al.add(str.substring(c,d));`
 `}`
`}`

d) `String str = "12345";`
`StringBuilder sb = new StringBuilder();`
`for(int i = 1; i < str.length() + 1; i ++)`
`{`
 `for(int k = 0; k < str.length(); k++)`
 `{`
 `int a = k;`
 `int b = k + i;`
 `int c = str.length() - 1 - k - i;`
 `int d = str.length() - 1 - k;`
 `if(i > 1 && k > 0)sb.append(",");`
 `sb.append(str.substring(a,b));`
 `if(a < c){`
 `sb.append(",");`
 `sb.append(str.substring(c,d));`
 `}`
 `}`
`}`
`string str = sb.toString();`
`String[] sArray = str.split("\\\\,");`

16. Given a number (maximum 99), write a program to determine and print the non-prime numbers, in the range from 0 to that input number, and store the prime-numbers in a singly-linked list. Finally, print the elements stored in the singly-linked list.
17. A school hallway has N number of lockers, N is to be input up to a maximum of 100. A student goes through the hallway opening every locker. Then, a second student goes down the hallway closing every other locker. A third student goes down the hallway, and at every third locker, opens it if it is closed and closes it if it was opened. This continues on for a total of N students opening and closing lockers. Write a program to compute how many lockers remain opened when N students are finished with this exercise.
18. What is the importance of finally block in exception handling? Explain with an example program.
19. Write a program that simulates a random walk of a robot in an empty room. The room is a (n x n) square and the robot starts in position (1 x 1). The robot can move east, move west, move north, or move south, at a time. When the robot hits a boundary and cannot proceed further it makes a random turn and attempts to continue its walk. Define an exit location in this room and allow the robot to walk until it finds the exit. Print the current location of the robot on the screen at every move. Stop the program if the robot failed to find the exit within 10000 moves. Each move should be randomly generated.
20. Write a program that accepts a sequence of numbers and determines whether the input numbers represent a magic square. A magic square is one in which the summation of both diagonals, all rows, and all columns add up to the same number. The program should print the total if the input numbers represent a 2D magic square. The program prints -1 if the rows do not all add up to the same number. The program prints -2 if the columns do not all add up to the same number. The program prints -4 if the diagonals do not all add up to the same number. The program prints -8 if the summation of all of the rows agree, the summation of all of the columns agree, and the summation of both diagonals agree, but these are not the same sums.
21. Write a program that exemplifies the difference between Data Streams and Object Streams.
22. Write a program that writes the frequency of all alphabets from an input file (e.g., 'a' – 25 times, 'b' – 17 times, and so on) into an output file. Also, the program should write the frequency of the size of

words (e.g., 1 letter words – x times, 2 letter words – y times, and so on). Use the program CountLetter.java (<http://docs.oracle.com/javase/tutorial/essential/io/QandE/CountLetter.java>) as a reference.

23. Write the following function that tests whether a two-dimensional list has four consecutive numbers of the same value, either horizontally, vertically, or diagonally.

```
public static boolean isConsecutiveFour(int[][] values)
```

Write a test program that prompts the user to enter the number of rows and columns of a two-dimensional list and then the values in the list and displays True if the list contains four consecutive numbers with the same value. Otherwise, display False. Here are some examples of the true cases:

0 1 0 3 1 6 1	0 1 0 3 1 6 1	0 1 0 3 1 6 1	0 1 0 3 1 6 1
0 1 6 8 6 0 1	0 1 6 8 6 0 1	0 1 6 8 6 0 1	0 1 6 8 6 0 1
5 6 2 1 8 2 9	5 5 2 1 8 2 9	5 6 2 1 6 2 9	9 6 2 1 8 2 9
6 5 6 1 1 9 1	6 5 6 1 1 9 1	6 5 6 6 1 9 1	6 9 6 1 1 9 1
1 3 6 1 4 0 7	1 5 6 1 4 0 7	1 3 6 1 4 0 7	1 3 9 1 4 0 7
3 3 3 3 4 0 7	3 5 3 3 4 0 7	3 6 3 3 4 0 7	3 3 3 9 4 0 7

24. Define the MyRectangle2D class that contains:
- Two double data fields named x and y that specify the center of the rectangle with get and set methods. (Assume that the rectangle sides are parallel to x- or y- axes.)
 - The double data fields width and height with get and set methods.
 - A no-arguments constructor that creates a default rectangle with (0, 0) for (x, y) and 1 for both width and height.
 - A constructor that creates a rectangle with the specified x, y, width, and height.
 - A method getArea() that returns the area of the rectangle.
 - A method getPerimeter() that returns the perimeter of the rectangle.
 - A method contains(double x, double y) that returns true if the specified point (x, y) is inside this rectangle. See Figure 1(a).
 - A method contains(MyRectangle2D r) that returns true if the specified rectangle is inside this rectangle. See Figure 1(b).
 - A method overlaps(MyRectangle2D r) that returns true if the specified rectangle overlaps with this rectangle. See Figure 1(c).

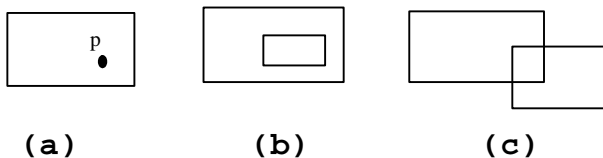


Figure 1

Figure 1(a) A point is inside the rectangle.

Figure 1(b) A rectangle is inside another rectangle.

Figure 1(c) A rectangle overlaps another rectangle.

25. What is a Java Virtual Machine (JVM)? How does JVM allow Java to run on almost any computer platform? Explain in detail.

26. Create a Circle.java class that uses the value of a circle's radius to compute the circle's area and circumference. Then, create another class called Cylinder.java. A cylinder of height 'h' can be thought of as 'h number of circles' stacked on top of each other. Implement the methods cylinderVolume() and cylinderSurfaceArea() in Cylinder.java that uses Circle's circleArea() and circleCircumference() methods, respectively.