Sri Lanka Institute of Information Technology

B.Sc. Special Honours Degree
in
Information Technology

Final Examination Year 2,

Semester 1(2017)

IT200 - Software Technology II

Duration: 3 Hours

October 2017

Instruction to Candidates:

- This paper contains 5 questions.
- Answer all questions.
- Provide the answers in the booklets given.
- Total marks for the paper will be 100.
- Mark for each question is mentioned in the paper. Return the question paper with the registration number, name and the group id written on the cover page.

Question 1 (20 marks)

a) ***"Java is not a robust language"***

Do you agree with this statement? Justify your answer. (2 marks)

b) Explain how postfix and prefix increment operator (++) works with a suitable example with expected output. (3 marks)

c) Convert following if-else into ternary operator. (2 Marks)

```
int price() {
   if (isPremiumMember()) {
      return 80;
   }
   else {
      return 100;
   }
```

d) Write a program in Java to find out if an user input number is prime. (Ex:input 7: output true , input 9 : output false) A number is called prime if it is divisible by either itself or 1. The input must be taken as keyboard input. (5 marks)

Hint: You can use nextInt() method in Scanner class

e) Write a Java method called **"maxOccuredChar(String str)"** to print the characters repeated more than once in the sentence and their number of occurrences. This method must be called from **main(String args[])** method. The input is taken as a command line argument.

Hint : use to charArray() method (8 marks)

Question 2 **(20 marks)**

Assume that you have been asked to develop software for a Vehicle Rental company using Java programming language. The following is the scenario:

The system manages information related to the Vehicles owned by this company. Mainly two categories of *Vehicles* by rented in this company: *Trucks* and *Cars*. A vehicle has make, model, rental rate, number of days vehicle not available and income as its properties. Users should be able to calculate rental cost and number of days vehicle not available regardless of the type of the vehicle. The calculate rental cost method must update the income attribute when number of travelling days are given as the input parameter. The method that calculates the number of days the vehicle is not available takes number of travelling days as the input parameter and updates the relevant attribute. If the days vehicle not available exceeds 365, value need to be set to zero. The check availability operation should return the number of days vehicle not available.

A Car is a vehicle type associated with number of passengers. For each passenger Rs 100 charged and this amount is added to rental cost. Number of passengers must be less than 5. Car has make, model, rental rate, number of days vehicle not available, income and number of passengers as its properties. Car has a method called addPassengerCost() that calculates the passenger cost and updates the rental income.

A Truck is a vehicle category with tonnage. A fee is charged for tonnage transported. Truck has make, model, rental rate, number of days vehicle not available, income, tonnage transported (initially zero) and truck capacity. There should be a track of number of tons transported by a vehicle by truck. For *any* Truck rental, the company charges Rs 10000 if the transported tons are less than half of the truck capacity. However, if transported capacity is more than the half of the capacity, then Rs 100 is charged for each additional ton. To implement this operation Truck should have a method called capacityCost(). capacityCost() method should calculate the amount charged for capacity transported of each trip. Also note that once the cost is calculated the amount needs to be added to the rental cost as an additional charges.

i. Implement the Vehicle class. Apply data hiding capabilities to the defined class. Implement calculateRentalCost(), checkAvaialability() and calculateDaysNotAvailable() methods.
Include a constructor that will take the values relating to the make, model, rental rate, number of days vehicle not available and income. Overload the constructor in a way it accepts only the rental rate, number of days vehicle not available and income.
Hint : use this, this() keywords appropriately (6 marks)

ii. Implement Car class and implement addPassengerCost() method.
Include the constructor that will take the values relating to the make, model, rental rate, number of days vehicle not available, income and number of passengers.
Hint : use this, super() keywords appropriately (5 marks)

iii. Implement the Truck class and capacityCost() method.
Override the calculateRentalCost() method with appropriate changes.
Include the constructor that will take the values relating to the make, model, rental rate, number of days vehicle not available, income, tonnage transported and truck capacity. When a Truck is created the tonnage transported is initially zero.
Hint : use this, super, super() keywords appropriately (9 marks)

Question 3 (20 marks)

a) What will be the output of the following code segment when you try to compile and execute it? State reasons for your answer. (3 marks)

```
public class Person {
   String name;
   int age;

   public Person(String name) {
      this.name = name;
   }

}

public class Student extends Person{
   String studentId;

   public Student(String s) {
     this.studentId =s;
   }
}
```

b) What will be the output of the following code segment when you try to compile and execute it? State reasons for your answer. (3 marks)

```
public class Person {
   String name;
   final int age;

   public void setName(String name) {
      this.name = name;
   }

}

public class Student extends Person{
   String studentId;

   public Student(String s) {
     this.studentId =s;
   }

   public void printDetails(){
      System.out.println(this.studentId);
      System.out.println(this.name);
      System.out.println(this.age);
   }
}
```

c) What will be the output of the following code segment when you try to compile and execute it? State reasons for your answer. (3 marks)

```
abstract public class Person {
   String name;
   double salary;

   public void setName(String name) {
      this.name = name;
   }

   abstract public void calcSalary(){

   }

}

public class Student extends Person{
   String studentId;

   public Student(String s) {
     this.studentId =s;
   }

   public void calcSalary() {
      System.out.println("250000");
   }
```

d) What will be the output of the following code segment when you try to compile and execute it? State reasons for your answer. (3 marks)

```
class Parent {

   private int x;
   private int y;

   double addition() {
      return (x + y);
   }
}

public class Child extends Parent {
        String name;
        String address;

       Child ( String n , String address, int x){
              this.name = n;
              address = a;
              this.x=x;
       }
   }
}
```

e) What will be the output of the following code segment when you try to compile and execute it? State reasons for your answer. (4 marks)

```
interface Employee {

   String company = "ABC Orgnization";

   public double calcSalary();

}

abstract class PartTimeEmployee implements Employee {

   double salary;

   abstract public double calcSalary();

   void getDetails() {
      System.out.println("My Information " + salary);
   }
}

public class Test {
   public static void main(String[] args) {
      Employee p = new Employee ();

      PartTimeEmployee s = new PartTimeEmployee();



   }
}
```

f) What will be the output of the following code segment when you try to compile and execute it? State reasons for your answer. (4 marks)

```
class Student {

   private String name;
   private String regNumber;
   private double gpa;

  public void testMethod(){
       System.out.println("The Student");
  }

}

class PartTimeEmployee extends Student {

   double salary;

  public void testMethod(){
       System.out.println("The Part Time Student");
  }


}

class Demo {

   public static void main(String[] args) {
       Student st = new PartTimeEmployee();
       st.testMethod();
   }
}
```

Question 4 (20 marks)

a) What is the difference between an exception and error? Give examples (4 marks)

b) What exception will be thrown when the following code block is executed? State the reason.

(2 marks)

```
Integer[][] ints = { { 1, 2, 3 }, { null }, { 7, 8, 9 } };
System.out.println("value = " + ints[1][1].intValue());
```

c) What is the difference between run() and start() in Thread? (2 marks)

d) What is daemon thread? Can daemon threads used as user threads? (3 marks)

e) When should a thread must be interrupted? (1 marks)

f) Write a Java Program to handle `RuntimeException` thrown when executing threads.

(8 marks)

Question 5 (20 marks)

a) Describe Enterprise level application. Write 3 characteristics of an enterprise level application. (4 marks)

b) Explain 3-tiered architecture in detail (6 marks)

c) Explain Entity Bean. (3 marks)

d) Explain the purpose of EJB Container. (2 Marks)

e) Explain Message Driven Beans (2 Marks)

f) Explain 3 differences between Servlets and JSP (3 Marks)