Please check that this question paper contains 9 questions and 2 printed pages within first ten minutes.

[Total No. of Questions: 09] [Total No. of Pages: 3]

Uni. Roll No. …….

Program: B.Tech. (IT)
Semester: 5
Name of Subject: Programming in Java
Subject Code: PCIT-109
Paper ID: 16440

08-01-2022(E)

Time Allowed: 02 Hours **Max. Marks: 60**

NOTE:

1) Each question is of 10 marks. In case of question contains two sub-parts like 'a' and 'b' then each sub-part is of 5 marks.
2) Attempt any six questions out of nine
3) For programs it is expected that suitable assumptions are made and stated wherever micro-level requirement related to the code to be developed is not mentioned. All test cases with inputs and outputs should be explained along with the program. If concept to be used is not mentioned for developing a code, student is free to use any concept related to Java Programming.

Q1. a) With the help of a program, demonstrate **static** variables, methods, and blocks.

b) What goes behind the scene when your computer runs a Java program?

Q2. a) Design an applet based code to draw a bar chart for the runs scored by a cricket player in 6 one day matches.
b) With the help of a program, demonstrate command line arguments.

Q3. Design a menu driven code using methods to handle following:

1. Start counting from the input number incrementing the value by 1 for each iteration.

The counting stops once the current value is divisible by 10.

2. Find the count of factors for a given number. For example, the factors of 10 are 1,2,5,10 and count in this case is 4.

Q4. a) A person has to transfer Rs. X from his/her e-Wallet. Wallet app will accept transfer transactions if 'X' is a multiple of 5. For each transfer, Wallet Company charges Rs. 0.75. With the help of code calculate account balance in Wallet for the following assumed cases:

Case 1: Enough amount is in the Wallet.
Case 2: Amount to be transferred is not a multiple of 5.
Case 3: Enough amount is not in the Wallet.

b) Differentiate between method overloading and method overriding.

Q5. a) "Interfaces can be extended", substantiate this statement by developing a suitable code fragment.

b) Write a program to demonstrate inter-thread communication.

Q6. There are 'X' programmers who want to use a single high performance computing facility (equipment). In order to avoid clashes in the availability of equipment, a plan is informed:

- First programmer can use facility at time 0 and finish latest by T_1.
- Second programmer can use facility at time $T_1$and finish latest by T_2.
- And likewise.
- X-th programmer can use facility at time T_{x-1} and finish latest by T_x.

The i-th programmer requires K_i units of time. Not all programmers will be able to use facility. Write a program to find how many programmers will be able to use facility with the provided plan.

Q7. a) Compare recursion with iteration.

b) Design a code to handle **ArrayIndexOutOfBounds**, **ArithmeticException**, **NullPointerException** exceptions. There should also be an exception handler available for any user defined exception.

Q8. Develop code to perform the following task:

Suppose you have data containing two fields (roll_no, obtained_marks) for the subject Java where maximum marks are 60. Assume there are 65 students in the class and each one having has obtained some marks out of 60 i.e. no one is absent. Consider data in such a way that 40% of the maximum marks is the pass criteria and few students have obtained marks ranging 0 to 15 (Case-I) and others have obtained marks in the range of 16 to 60 (Case-II).

You are supposed to compute average of obtained_marks of the stated cases. Assume average of obtained_marks of students in Case-II is less than 46.

Now you can have additional fields called additional_marks and final_marks in your data.

Populate the data in such a way that you start assigning marks in the field additional _marks starting from 1 to all the students of Case-II(here final_marks=obtained_marks+additional_marks). Care must be taken that in no case final_marks be greater than maximum marks.

Perform the iterative process of addition of marks for all the students of Case-II and stop the process when average of final_marks of all students in Case-II is 46 (our target average). Ensure that all students of Case-II are given equal additional_marks except for students who have reached maximum limit.

Your program must be such that target average for Case-II must be read through appropriate prompt message. If target average for Case-II is already greater than or equal to the Case-II obtained marks of students, no action is required, just display an appropriate message in that case.

Q9. a) Design code to swap two numbers without using a third variable with the help of bitwise operators.

b) Design a package to contain the class Student that contains data members such as name, roll number and another package contains the interface Sports which contains some sports information. Import these two packages in a package called Report which process both Student and Sport and give the report.

****