



COMSATS University Islamabad  
Department of Computer Science  
Terminal Examination - FALL 2025

Class: BCS/BSE/BAI/BDS/BCT.  
Subject: CSC103-Programming Fundamentals  
Instructor:

Marks: 50  
Time allowed: 3 Hours  
Date: 27-12-2025

Note: Return question paper along with answer book.

[CLO-1]-Demonstrate the fundamental concepts of programming.

Question# 1: Show the output/Errors of the following code snippets.

(10 marks)

1) Show the output

```
int sum = 0, y = 0;
do
{ for(int x=1; x<5; x++) {
    sum += x;
    y++;
} while(y<3);
System.out.print(sum);
```

error

2) Give the final output.

```
Suppose table is a 2D array:
for(int i=0; i<3; i++){
    for(int j=0; j<3; j++){
        System.out.print(table[i][j]+" ");
        break;
    }
}
```

|    |    |    |
|----|----|----|
| 10 | 20 | 30 |
| 40 | 50 | 60 |
| 70 | 80 | 90 |

3) What does the following program print?

```
String s = "H1";
for (int i=1; i<10; i++) s+=s;
System.out.print(i);
```

4) What will be the output

```
int i=0;
while (i++<5); {
    System.out.print(i);
}
```

5) Consider token = "123". What will be returned?

```
public static boolean isNumeric(String token) {
    try { double num = Double.parseDouble(token);
        return true;
    } catch (java.lang.NumberFormatException ex) {
        return false;
    }
}
```

6) What does the following program print?

```
String str = "Java_is_fun";
for(int i=0; i<str.length(); i++) {
    if("aeiou".indexOf(str.charAt(i)) > -1){
        str=str.substring(0,i)+"*";
    }
} System.out.print(str);
```

7) What does the following program print?

```
public static String r(int m){
    if(m > 0)
        return "+" + r(m-1);
    else
        return "-";
}
public static String s(int m, int n){
    if(n > 0)
        return "-" + s(m, n-1) + "-";
    else
        return r(2*m);
} System.out.print(s(3,2));
```

8) What does the following program print?

```
int sum[][] = new int[4][4];
for(int k=0; k<4; k++){
    sum[k][0] = 1;
}
for(int k=0; k<4; k++){
    sum[0][k] = 1;
}
for(int m=1; m<4; m++)
    for(int n=1; n<4; n++)
        sum[m][n] = sum[m-1][n-1] + sum[m][n-1];
for(int n=1; n<4; n++)
    System.out.print(sum[3][n] + " ");
```

9) Which catch block will be executed?

```
public class Test {
    public static void main(String[] args) {
        try { method();
            System.out.println("After the method call");
        } catch (ArithmeticException ex) {
            System.out.println("ArithmeticException");
        } catch (RuntimeException ex) {
            System.out.println("RuntimeException");
        } catch (Exception e) {
            System.out.println("Exception");
        }
    }
    static void method() throws Exception {
        System.out.println(1 / 0);
    }
}
```

10) Which catch block will be executed?

```
public class Test {
    public static void main(String[] args) {
        try {
            int [] list = new int[10];
            System.out.println("list[10] is" +list[10]);
        } catch (ArithmeticException ex) {
            System.out.println("ArithmeticException");
        } catch (RuntimeException ex) {
            System.out.println("ArrayIndexOutOfBoundsException");
        } catch (Exception ex) {
            System.out.println("Exception");
        }
    }
}
```

[CLO-2]- Employ programming constructs using a programming language.

Question# 2

(7 marks)

Write a method `public static String removeSpam (String str, String [] spam)` that takes two parameters: an input string and a list spam words. The method should return a new string where all instances of the spam words have been removed. The removal should be case-insensitive.

Sample Run:

String str: "This is a test string with some spam words like spam and junk."  
String [] spam = {"spam", "junk"}  
Return string: "This is a test string with some words like and."  
Suppose 'spam' and 'junk' are the spam words in the above case.

Question# 3

(5 marks)

A company is designing a Payroll System that calculates the salaries of different types of employees. The system must support three cases:

1. Hourly employees, where the salary is computed using  $hoursWorked \times hourlyRate$ ;
2. Salaried employees, whose salary is simply their  $monthlySalary$ ;
3. Commission-based employees, whose salary is  $baseSalary + (salesAmount \times commissionRate)$ .

You are only required to write three methods `calculateSalary()` with complete method headers to implement all three calculations. Do not write main method.

Question# 4

(8 marks)

Write a Java program that uses recursion to calculate the sum of all elements in an integer array. You must define a recursive method with the header `public static int sumArray(int[] arr, int index)` that returns the sum of the array starting from the given index.

In the main method, the student must read the array elements from the user, call the recursive method, and finally display the total sum produced by the recursion.

Question# 5

(10 marks)

A restaurant chain, FoodieExpress, is developing a software system to manage its daily orders across multiple branches. Each branch records the number of orders received for each menu item every day. The management wants to analyze this data to determine the total orders for each item, branch-wise sales, and overall sales.

There are 5 menu items: Burger, Pizza, Pasta, Salad, and Sandwich.

There are 3 branches: Downtown, Uptown, and Suburban.

The system should be able to:

1. Input the daily orders for each branch in a 2D array (rows = branches, columns = menu items).
2. Calculate the total orders for each menu item across all branches.
3. Calculate the total orders for each branch.
4. Identify the menu item with the highest overall orders.

[CLO-3]- Handle programs utilizing exception and file I/O.

Question# 6

(10 marks)

You are required to develop a Java program to maintain PSL match records. The league consists of six teams: Islamabad United, Lahore Qalandars, Multan Sultans, Quetta Gladiators, Peshawar Zalmi, and Karachi Kings. Whenever a match is played between any two teams, your program should store the match result in a text file named "PSL.txt".

The system should follow these rules for awarding points:

- 2- points for a Win,
- 1 point to each team for No Result (NR)
- 0 points for a Loss

Your program must implement the following options:

1. **Submit Result** – Accepts the names of the two Teams and the match outcome (Won, Lost, NR), then updates the Points column in PSL.txt for each Team accordingly.
2. **Display Status** – Reads the stored match data and displays the updated points table.
3. **Display Qualifiers** – Determines and displays the names of the top two teams based on points, indicating which teams have qualified for the final.

When the user requests the current standings, the program should display the following Points Table:

| POINTS TABLE      |        |     |      |    |        |
|-------------------|--------|-----|------|----|--------|
| Teams             | Played | Won | Lost | NR | Points |
| Islamabad United  | 0      | 0   | 0    | 0  | 0      |
| Lahore Qalandars  | 0      | 0   | 0    | 0  | 0      |
| Multan Sultans    | 0      | 0   | 0    | 0  | 0      |
| Quetta Gladiators | 0      | 0   | 0    | 0  | 0      |
| Peshawar Zalmi    | 0      | 0   | 0    | 0  | 0      |
| Karachi Kings     | 0      | 0   | 0    | 0  | 0      |

NOTE: Handle the program utilizing exception and file I/O.