

# **TASK 1: BASIC CONDITIONAL STATEMENTS AND LOOPING PROGRAMS**

## **(A) COUNT EVEN AND ODD NUMBERS**

### **AIM**

To write a Java program that counts how many numbers are even and odd from five given integers.

### **ALGORITHM**

1. Start the program.
2. Read 5 numbers from the user.
3. Initialize even and odd counters to 0.
4. Check each number:
  - If divisible by 2 → increment even.
  - Otherwise → increment odd.
5. Display the counts.
6. Stop.
- 7.

### **PROGRAM**

```
import java.util.Scanner;

class EvenOddCount {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int even = 0, odd = 0;

        System.out.println("Enter 5 numbers:");
        for(int i=0; i<5; i++){
            int num = sc.nextInt();

            if(num % 2 == 0)
                even++;
            else
                odd++;
        }
    }
}
```

```
        System.out.println("Even numbers = " + even);
        System.out.println("Odd numbers = " + odd);
    }
}
```

## **OUTPUT**

Enter 5 numbers:

1 2 3 4 5

Even numbers = 2

Odd numbers = 3

## **RESULT**

Thus, the program successfully counts the even and odd numbers.

## **(B) SUM OF LAST DIGIT OF TWO NUMBERS**

### **AIM**

To write a Java program that extracts the last digits of two numbers and finds their sum.

### **ALGORITHM**

1. Start.
2. Read two integers.
3. Find last digit using number % 10.
4. Add the last digits.
5. Display the result.
6. Stop.

### **PROGRAM**

```
import java.util.Scanner;

class LastDigitSum {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter first number: ");
        int a = sc.nextInt();

        System.out.print("Enter second number: ");
        int b = sc.nextInt();

        int sum = (a % 10) + (b % 10);

        System.out.println("Sum of last digits = " + sum);
    }
}
```

### **OUTPUT**

```
Enter first number: 27
Enter second number: 45
Sum of last digits = 12
```

### **RESULT**

Thus, the program successfully calculates the sum of the last digits.

## **(C) CHECK WHETHER A NUMBER IS PRIME**

### **AIM**

To write a Java program to determine if a given number is prime.

### **ALGORITHM**

1. Start.
2. Read the number n.
3. If  $n \leq 1$ , it is not prime.
4. Check divisibility from 2 to  $\sqrt{n}$ .
5. If divisible  $\rightarrow$  Not Prime.
6. Otherwise  $\rightarrow$  Prime.
7. Display result and stop.

### **PROGRAM**

```
import java.util.Scanner;

class PrimeCheck {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter a number: ");
        int n = sc.nextInt();

        boolean isPrime = true;

        if(n <= 1){
            isPrime = false;
        } else {
            for(int i = 2; i <= Math.sqrt(n); i++){
                if(n % i == 0){
                    isPrime = false;
                    break;
                }
            }
        }

        if(isPrime)
```

```
        System.out.println("Prime Number");
    else
        System.out.println("Not a Prime Number");
    }
}
```

## **OUTPUT**

Enter a number: 7

Prime Number

## **RESULT**

Thus, the program correctly checks whether the number is prime.

## **(D) Factorial of a Number**

### **AIM**

To write a Java program that calculates the factorial of a given number.

### **ALGORITHM**

1. Start.
2. Read integer n.
3. Initialize fact = 1.
4. Multiply numbers from 1 to n.
5. Display factorial.
6. Stop.

### **PROGRAM**

```
import java.util.Scanner;

class Factorial {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter a number: ");
        int n = sc.nextInt();

        long fact = 1;

        for(int i=1; i<=n; i++){
            fact *= i;
        }

        System.out.println("Factorial = " + fact);
    }
}
```

### **OUTPUT**

```
Enter a number: 5
Factorial = 120
```

### **RESULT**

Thus, the program successfully computes the factorial.

## **(E) NTH FIBONACCI NUMBER**

### **AIM**

To write a Java program that finds the Nth term in the Fibonacci sequence.

### **ALGORITHM**

1. Start.
2. Read n.
3. Initialize a = 0, b = 1.
4. Compute next terms using  $c = a + b$ .
5. Repeat until the Nth term is reached.
6. Display the result.
7. Stop.

### **PROGRAM**

```
import java.util.Scanner;

class Fibonacci {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter N: ");
        int n = sc.nextInt();

        int a = 0, b = 1, c = 0;

        if(n == 0)
            System.out.println("Fibonacci = 0");
        else if(n == 1)
            System.out.println("Fibonacci = 1");
        else {
            for(int i = 2; i <= n; i++){
                c = a + b;
                a = b;
                b = c;
            }
        }
    }
}
```

```
        System.out.println("Fibonacci = " + c);
    }
}
}
```

### **OUTPUT**

Enter N: 6

Fibonacci = 8

### **RESULT**

Thus, the program successfully finds the Nth Fibonacci number.