CDAC MUMBAI

OOPJ

Assignment-2

1. Arithmetic & Assignment Operators

Q1: Write a program to swap two numbers without using a third variable and without using arithmetic operators like + or - . Hint : Use bitwise XOR ^ operator.

```
public class SwapUsingXOR {
   public static void main(String[] args) {
      int a = 5, b = 7;
      System.out.println("Before swapping: a = " + a + ", b = " + b);

      a = a ^ b;
      b = a ^ b;
      a = a ^ b;

      System.out.println("After swapping: a = " + a + ", b = " + b);
   }
}
```

```
Microsoft Windows [Version 10.0.26100.3323]
(c) Microsoft Corporation. All rights reserved.

C:\Users\abhiv>cd C:\Users\abhiv\Desktop\CDACKH\OOPJ

C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac SwapUsingXOR.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java SwapUsingXOR
Before swapping: a = 5, b = 7
After swapping: a = 7, b = 5
```

Q2: Write a program to check whether a given number is even or odd using only bitwise operators . Hint : Use n & 1 to check.

```
public class EvenOddCheck {
   public static void main(String[] args) {
      int n = 7;

      if ((n & 1) == 0) {
         System.out.println(n + " is even");
      } else {
         System.out.println(n + " is odd");
      }
```

```
  }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac EvenOddCheck.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java EvenOddCheck
7 is odd
```

Q3: Implement a program that calculates the sum of digits of an integer using modulus (%) and division (/) operators .

```
public class SumOfDigits {
  public static void main(String[] args) {
    int num = 1234;
    int sum = 0;

    while (num > 0) {
      sum += num % 10;
      num /= 10;
    }

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

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac SumOfDigits.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java SumOfDigits
Sum of digits: 10
```

Q4: Write a program to find whether a given number is divisible by 3 without using the modulus (%) or division (/) operators. Hint : Use subtraction and bitwise shifts .

```
import java.util.Scanner;
public class DivisibleByThree {
  public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    System.out.print("Enter a number: ");
    int num = scanner.nextInt();
    scanner.close();

    if (isDivisibleBy3(num)) {
      System.out.println(num + " is divisible by 3");
    } else {
```

```
            System.out.println(num + " is not divisible by 3");
        }
    }

    static boolean isDivisibleBy3(int num) {
        // Convert negative numbers to positive
        num = Math.abs(num);

        while (num > 0) {
            int sum = 0;
            while (num > 0) {
                sum += (num & 1);
                num -= 2;
            }
            num = sum;
        }

        return (num == 0 || num == 3);
    }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac DivisibleByThree.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java DivisibleByThree
Enter a number: 30
30 is divisible by 3
```

Q5: Write a Java program to swap two numbers using the += and -= operators only.

```
public class SwapUsingAddSub {
    public static void main(String[] args) {
        int a = 10, b = 20;
        System.out.println("Before swapping: a = " + a + ", b = " + b);

        a += b;
        b = a - b;
        a -= b;

        System.out.println("After swapping: a = " + a + ", b = " + b);
    }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac SwapUsingAddSub.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java SwapUsingAddSub
Before swapping: a = 10, b = 20
After swapping: a = 20, b = 10
```

2. Relational & Logical Operators

Q6: Write a program to find the largest of three numbers using only the ternary operator (? :) .

```
import java.util.Scanner;

public class LargestNumber {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter three numbers: ");
      int a = scanner.nextInt();
      int b = scanner.nextInt();
      int c = scanner.nextInt();
      scanner.close();

      int largest = (a > b) ? (a > c ? a : c) : (b > c ? b : c);
      System.out.println("The largest number is: " + largest);
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac LargestNumber.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java LargestNumber
Enter three numbers: 12 45 35
The largest number is: 45
```

Q7: Implement a Java program that checks whether a given year is a leap year or not using logical (&& , ||) operators .

```
import java.util.Scanner;

public class LeapYearCheck {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a year: ");
      int year = scanner.nextInt();
      scanner.close();

      boolean isLeap = (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);
      System.out.println(year + (isLeap ? " is a leap year." : " is not a leap year."));
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac LeapYearCheck.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java LeapYearCheck
Enter a year: 2025
2025 is not a leap year.
```

Q8: Write a program that takes three boolean inputs and prints true if at least two of them are true . Hint : Use logical operators (&& , ||).

```
import java.util.Scanner;

public class AtLeastTwoTrue {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter three boolean values (true/false): ");
      boolean a = scanner.nextBoolean();
      boolean b = scanner.nextBoolean();
      boolean c = scanner.nextBoolean();
      scanner.close();

      boolean result = (a && b) || (b && c) || (a && c);
      System.out.println("At least two are true: " + result);
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac AtLeastTwoTrue.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java AtLeastTwoTrue
Enter three boolean values (true/false): true false true
At least two are true: true
```

Q9: Implement a Java program that checks if a number is within a specific range (20 to 50) without using if-else . Hint : Use logical AND (&&) in a print statement

```
import java.util.Scanner;

public class NumberInRange {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = scanner.nextInt();
      scanner.close();

      System.out.println((num >= 20 && num <= 50) ? "Within range" : "Out of range");
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac NumberInRange.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java NumberInRange
Enter a number: 54
Out of range

C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac NumberInRange.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java NumberInRange
Enter a number: 38
Within range
```

Q10: Write a program to determine if a character is a vowel or a consonant using the ternary operator.

```
import java.util.Scanner;

public class VowelOrConsonant {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a character: ");
      char ch = scanner.next().charAt(0);
      scanner.close();

      ch = Character.toLowerCase(ch);
      String result = (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u')
                  ? "Vowel" : "Consonant";

      System.out.println(ch + " is a " + result);
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac VowelOrConsonant.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java VowelOrConsonant
Enter a character: i
i is a Vowel

C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac VowelOrConsonant.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java VowelOrConsonant
Enter a character: b
b is a Consonant
```

3. Bitwise Operators

Q11: Write a program to check if a given number is a power of 2 using bitwise operators. Hint : n & (n - 1) == 0 for positive numbers.

```
import java.util.Scanner;

public class PowerOfTwo {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = scanner.nextInt();
      scanner.close();

      boolean isPowerOfTwo = (num > 0) && ((num & (num - 1)) == 0);
      System.out.println(num + (isPowerOfTwo ? " is a power of 2." : " is not a power of
2."));
   }
```

}

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac PowerOfTwo.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java PowerOfTwo
Enter a number: 24
24 is not a power of 2.
```

Q12: Write a Java program to multiply a number by 8 without using * or / operators. Hint : Use bitwise left shift (<<).

```
import java.util.Scanner;

public class MultiplyByEight {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = scanner.nextInt();
      scanner.close();

      int result = num << 3; // Left shift by 3 is equivalent to multiplying by 8
      System.out.println("Result after multiplying by 8: " + result);
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac MultiplyByEight.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java MultiplyByEight
Enter a number: 35
Result after multiplying by 8: 280
```

Q13: Implement a Java program to find the absolute value of an integer using bitwise operators. Hint : mask = num >> 31; abs = (num + mask) ^ mask;

```
import java.util.Scanner;

public class AbsoluteValue {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = scanner.nextInt();
      scanner.close();

      int mask = num >> 31; // All 1s if negative, all 0s if positive
      int absValue = (num + mask) ^ mask;
      System.out.println("Absolute value: " + absValue);
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac AbsoluteValue.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java AbsoluteValue
Enter a number: 28
Absolute value: 28
```

Q14: Write a program to count the number of 1s (set bits) in a binary representation of a number using bitwise operations. Hint : Use n & (n - 1) .

```
import java.util.Scanner;

public class CountSetBits {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = scanner.nextInt();
      scanner.close();

      int count = 0;
      while (num > 0) {
         num &= (num - 1); // Remove the rightmost set bit
         count++;
      }

      System.out.println("Number of 1s in binary representation: " + count);
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac CountSetBits.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java CountSetBits
Enter a number: 15
Number of 1s in binary representation: 4
```

Q15: Implement a program to swap odd and even bits of a number using bitwise operators. Hint : Use masks: (x & 0xAAAAAAAA) >> 1 | (x & 0x55555555) << 1

```
import java.util.Scanner;

public class SwapOddEvenBits {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = scanner.nextInt();
      scanner.close();

      int evenBits = num & 0xAAAAAAAA; // Mask even bits (10101010...)
      int oddBits = num & 0x55555555;  // Mask odd bits  (01010101...)

      int swapped = (evenBits >> 1) | (oddBits << 1);
      System.out.println("Number after swapping odd and even bits: " + swapped);
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac SwapOddEvenBits.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java SwapOddEvenBits
Enter a number: 150
Number after swapping odd and even bits: 105
```

4. Ternary Operator Challenges

Q16: Write a program that determines whether a given number is positive, negative, or zero using only the ternary operator .

```
import java.util.Scanner;
public class NumberCheck {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = scanner.nextInt();
      scanner.close();

      String result = (num > 0) ? "Positive" : (num < 0) ? "Negative" : "Zero";
      System.out.println("The number is: " + result);
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>java NumberCheck
Enter a number: 6
The number is: Positive

C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac NumberCheck.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java NumberCheck
Enter a number: -4
The number is: Negative
```

Q17: Implement a Java program that finds the minimum of four numbers using nested ternary operators.

```
import java.util.Scanner;
public class MinimumOfFour {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter four numbers: ");
      int a = scanner.nextInt();
      int b = scanner.nextInt();
      int c = scanner.nextInt();
      int d = scanner.nextInt();
      scanner.close();

      int min = (a < b) ? ((a < c) ? ((a < d) ? a : d) : ((c < d) ? c : d))
                  : ((b < c) ? ((b < d) ? b : d) : ((c < d) ? c : d));

      System.out.println("The smallest number is: " + min);
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac MinimumOfFour.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java MinimumOfFour
Enter four numbers: 2 5 7 9
The smallest number is: 2
```

Q18: Given a student's percentage, print "Pass" if the percentage is 40 or above; otherwise, print "Fail" , using only the ternary operator.

```
import java.util.Scanner;
public class PassOrFail {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter the student's percentage: ");
      int percentage = scanner.nextInt();
      scanner.close();

      String result = (percentage >= 40) ? "Pass" : "Fail";
      System.out.println("Result: " + result);
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>java PassOrFail
Enter the student's percentage: 67
Result: Pass

C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac PassOrFail.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java PassOrFail
Enter the student's percentage: 24
Result: Fail
```

Q19: Write a Java program that checks whether a character is uppercase, lowercase, or not a letter using only the ternary operator.

```
import java.util.Scanner;
public class CharacterCheck {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a character: ");
      char ch = scanner.next().charAt(0);
      scanner.close();

      String result = (ch >= 'A' && ch <= 'Z') ? "Uppercase"
                  : (ch >= 'a' && ch <= 'z') ? "Lowercase"
                  : "Not a letter";

      System.out.println("Character type: " + result);
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>java CharacterCheck
Enter a character: G
Character type: Uppercase

C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac CharacterCheck.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java CharacterCheck
Enter a character: h
Character type: Lowercase
```

Q20: Implement a Java program that returns the absolute value of a given number using the ternary operator (without using Math.abs()

```
import java.util.Scanner;
public class AbsoluteValueTernary {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = scanner.nextInt();
      scanner.close();

      int absValue = (num < 0) ? -num : num;
      System.out.println("Absolute value: " + absValue);
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac AbsoluteValueTernary.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java AbsoluteValueTernary
Enter a number: 47
Absolute value: 47
```

5. Miscellaneous Operator Questions

Q21: Write a program that increments a number without using + or ++ operators.
Hint : Use bitwise - (~x) .

```
public class IncrementWithoutPlus {
   public static void main(String[] args) {
      int x = 5;
      int incremented = -~x;
      System.out.println("Incremented value: " + incremented);
   }
}
```

```
Microsoft Windows [Version 10.0.26100.3323]
(c) Microsoft Corporation. All rights reserved.

C:\Users\abhiv>cd C:\Users\abhiv\Desktop\CDACKH\OOPJ

C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac IncrementWithoutPlus.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java IncrementWithoutPlus
Incremented value: 6

C:\Users\abhiv\Desktop\CDACKH\OOPJ>
```

Q22: Implement a calculator that takes two numbers and an operator (+ , - , * , /) as input and prints the result using only switch-case .

```
import java.util.Scanner;

public class CalculatorSwitch {
  public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.print("Enter first number: ");
    int num1 = sc.nextInt();
    System.out.print("Enter operator (+, -, *, /): ");
    char op = sc.next().charAt(0);
    System.out.print("Enter second number: ");
    int num2 = sc.nextInt();
    sc.close();

    switch (op) {
      case '+': System.out.println("Result: " + (num1 - (-num2))); break;
      case '-': System.out.println("Result: " + (num1 + (~num2 + 1))); break;
      case '*': System.out.println("Result: " + (num1 * num2)); break;
      case '/':
        if (num2 != 0) System.out.println("Result: " + (num1 / num2));
        else System.out.println("Cannot divide by zero!");
        break;
      default: System.out.println("Invalid operator!");
    }
  }
}
```

```
Microsoft Windows [Version 10.0.26100.3323]
(c) Microsoft Corporation. All rights reserved.

C:\Users\abhiv>cd C:\Users\abhiv\Desktop\CDACKH\OOPJ

C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac IncrementWithoutPlus.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java IncrementWithoutPlus
Incremented value: 6

C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac CalculatorSwitch.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java CalculatorSwitch
Enter first number: 8
Enter operator (+, -, *, /): +
Enter second number: 24
Result: 32
```

Q23: Given a number, find whether it is odd or even using the & bitwise operator and print the result without using if-else .

```
public class OddEvenBitwise {
  public static void main(String[] args) {
    int num = 7;
    System.out.println(num + " is " + ((num & 1) == 0 ? "Even" : "Odd"));
  }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac OddEvenBitwise.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java OddEvenBitwise
7 is Odd
```

Q24: Write a program that prints all even numbers from 1 to 100 using only bitwise AND (&) and for loop.

```
public class EvenNumbersBitwise {
   public static void main(String[] args) {
      for (int i = 1; i <= 100; i++) {
         if ((i & 1) == 0) {
            System.out.println(i);
         }
      }
   }
}
```

```
C:\Users
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
44
46
48
50
52
54
56
58
60
62
64
66
68
70
72
74
76
78
80
82
84
86
88
90
92
94
96
98
100
```

Q25: Implement a program that reverses an integer number without using string conversion (StringBuilder or toCharArray). Hint : Use while(n!=0) { rev = rev * 10 + n % 10; n /= 10; }

```
import java.util.Scanner;

public class ReverseInteger {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = sc.nextInt();
      sc.close();

      int rev = 0;
      while (num != 0) {
         rev = rev * 10 + num % 10;
         num /= 10;
      }

      System.out.println("Reversed number: " + rev);
   }
}
```

```
C:\Users\abhiv\Desktop\CDACKH\OOPJ>javac ReverseInteger.java

C:\Users\abhiv\Desktop\CDACKH\OOPJ>java ReverseInteger
Enter a number: 45
Reversed number: 54
```