OOPJ: Assignment No-2

Name:Shital Hiray_JH

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.

Ans:.

```
import java.util.Scanner;

public class SwapBitwiseOp {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter two numbers: ");
        int x = sc.nextInt();
        int y = sc.nextInt();

        System.out.println("Before swapping: x = " + x + ", y = " + y);

        x = x ^ y;
        y = x ^ y;
        x = x ^ y;

        System.out.println("After swapping: x = " + x + ", y = " + y);

        sc.close();
    }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac SwapBitwiseOp.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java SwapBitwiseOp
Enter two numbers: 3
6
Before swapping: x = 3, y = 6
After swapping: x = 6, y = 3
```

Q2: Write a program to check whether a given number is **even or odd** using only **bitwise operators**.

Hint: Use `n & 1` to check.

Ans.:

```
import java.util.Scanner;

public class EvenOddBitwiseOp {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter two numbers: ");
                int n= sc.nextInt();
                if( (n&1)==0)
                {
                 System.out.print(n+ " is Even: ");
```

```
            }
            else{
                System.out.print(n+ " is Odd: ");
            }
            sc.close();
        }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java EvenOddBitwiseOp
Enter two numbers: 6
6 is Even:
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac EvenOddBitwiseOp.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java EvenOddBitwiseOp
Enter two numbers: 3
3 is Odd:
```

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

Ans.:

```
import java.util.Scanner;

public class SumOfDigits {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter an integer: ");
        int num = sc.nextInt();
        int sum = 0;
        int temp = Math.abs(num);
        while (temp > 0) {
            sum += temp % 10;
            temp /= 10;
        }
        System.out.println("Sum of digits: " + sum);

        sc.close();
    }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac SumOfDigits.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java SumOfDigits
Enter an integer: 3456
Sum of digits: 18
```

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**.

Ans.:

```
import java.util.Scanner;

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

```
        if (num < 0) num = -num;
        int sum = 0;

        while (num > 0) {
            sum += (num & 7);
            num = num >> 3;
        }
        if (sum == 0 || sum == 3 || sum == 6 || sum == 9) {
            System.out.println("The number is divisible by 3.");
        } else {
            System.out.println("The number is NOT divisible by 3.");
        }
                    sc.close();
    }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac DivisibleByThree.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java DivisibleByThree
Enter a number: 27
The number is divisible by 3.
```

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

```
import java.util.Scanner;

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

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

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

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

        System.out.println("After swapping:");
        System.out.println("First number (a): " + a);
        System.out.println("Second number (b): " + b);
    sc.close();
          }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java SwapNumbers
Enter first number (a): 5
Enter second number (b): 6
After swapping:
First number (a): 6
Second number (b): 5
```

---

2. Relational & Logical Operators

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

Ans.:

```
import java.util.Scanner;



public class LargestOfThree {

    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();

        System.out.print("Enter third number: ");

        int c = sc.nextInt();

        Int largest = (a > b) ? ((a > c) ? a : c) : ((b > c) ? b : c);

        System.out.println("The largest number is: " + largest);

                    sc.close();

    }

}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac LargestOfThree.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java LargestOfThree
Enter first number: 6
Enter second number: 9
Enter third number: 2
The largest number is: 9

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>
```

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

Ans.:

```
import java.util.Scanner;

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

```
        int year = sc.nextInt();
        if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {
            System.out.println(year + " is a leap year.");
        } else {
            System.out.println(year + " is not a leap year.");
        }
        sc.close();
    }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java LeapYearCheck
Enter a year: 2024
2024 is a leap year.

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac LeapYearCheck.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java LeapYearCheck
Enter a year: 2023
2023 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 (`&&`, `||`).

Ans:.

```
import java.util.Scanner;

public class AtLeastTwoTrue {
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.print("Enter first boolean value (true/false): ");

            boolean a = sc.nextBoolean();
        System.out.print("Enter second boolean value (true/false): ");

                boolean b = sc.nextBoolean();
        System.out.print("Enter third boolean value (true/false): ");

            boolean c = sc.nextBoolean();
        boolean result = (a && b) || (b && c) || (a && c);

        System.out.println("At least two values are true: " + result);
        sc.close();
    }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac AtLeastTwoTrue.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java AtLeastTwoTrue
Enter first boolean value (true/false): true
Enter second boolean value (true/false): false
Enter third boolean value (true/false): true
At least two values are true: true

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac AtLeastTwoTrue.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java AtLeastTwoTrue
Enter first boolean value (true/false): false
Enter second boolean value (true/false): false
Enter third boolean value (true/false): true
At least two values are true: false
```

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**.

Ans.:

```
import java.util.Scanner;

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

        System.out.print("Enter a number: ");
        int number = sc.nextInt();
        System.out.println("Is the number in the range 20 to 50? " + (number >= 20 && number
<= 50));
        sc.close();
    }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac NumberInRange.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java NumberInRange
Enter a number: 25
Is the number in the range 20 to 50? true
```

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

Ans.:

```
import java.util.Scanner;

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

        System.out.print("Enter a character: ");
        char ch = sc.next().toLowerCase().charAt(0); // Convert to lowercase to handle
uppercase input

        String result = (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u') ? "Vowel"
```

```
                : ((ch >= 'a' && ch <= 'z') ? "Consonant" : "Invalid input");

        System.out.println("The character is: " + result);

        sc.close();
    }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java VowelOrConsonant
Enter a character: abc
The character is: Vowel

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac VowelOrConsonant.jav
a

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java VowelOrConsonant
Enter a character: p
The character is: 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.

Ans.: import java.util.Scanner;

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



        if (n > 0 && (n & (n - 1)) == 0) {
            System.out.println(n + " is a power of 2.");
        } else {
            System.out.println(n + " is not a power of 2.");
        }
                    sc.close();
    }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac PowerOfTwo.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java PowerOfTwo
Enter a number: 5
5 is not a power of 2.

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac PowerOfTwo.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java PowerOfTwo
Enter a number: 4
4 is a power of 2.

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>
```

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

```
import java.util.Scanner;

public class MultiplyByEight {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int n = sc.nextInt();
        int result = n << 3;

        System.out.println("Result: " + result);
                    sc.close();
```

```
    }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac MultiplyByEight.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java MultiplyByEight
Enter a number: 5
Result: 40
```

Q13: Implement a Java program to find the **absolute value** of an integer using bitwise operators.

Hint: `mask = num >> 31; abs = (num + mask) ^ mask;`

Ans.:

```
import java.util.Scanner;

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


        int mask = num >> 31;  // Extracts the sign bit (0 for
positive, -1 for negative)
        int abs = (num + mask) ^ mask; // Computes absolute value

        System.out.println("Absolute value: " + abs);
     sc.close();
     }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac AbsoluteValue.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java AbsoluteValue
Enter a number: 5
Absolute value: 5

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac AbsoluteValue.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java AbsoluteValue
Enter a number: -5
Absolute value: 5
```

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)`.

Ans.:

```
import java.util.Scanner;

public class CountSetBits {
   public static int countOnes(int n) {
      int count = 0;
      while (n > 0) {
         n = n & (n - 1);
         count++;
      }
      return count;
   }
```

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


      int onesCount = countOnes(num);
      System.out.println("Number of 1s in binary: " + onesCount);
    sc.close();
        }
}
```

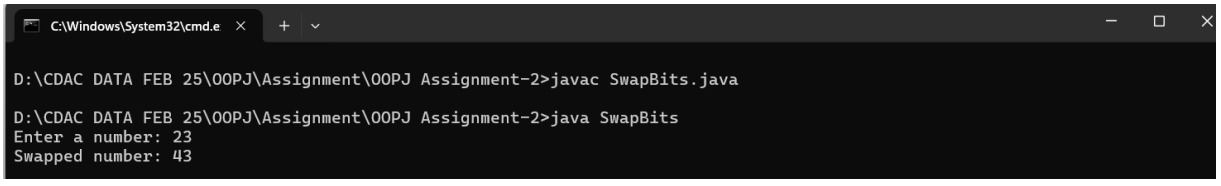
```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac CountSetBits.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java CountSetBits
Enter a number: 13
Number of 1s in binary: 3
```

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.`

Ans.:

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


        int swapped = ((n & 0xAAAAAAAA) >> 1) | ((n & 0x55555555) << 1);
        System.out.println("Swapped number: " + swapped);
                sc.close();
   }
}
```



---

4. Ternary Operator Challenges

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

Ans.:

```
import java.util.Scanner;

public class NumberCheck {

    public static void main(String[] args) {
```

```
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter a number: ");

        int num = sc.nextInt();

        String result = (num > 0) ? "Positive" : (num < 0) ? "Negative" : "Zero";

        System.out.println("The number is: " + result);

                sc.close();

         }

}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac NumberCheck.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java NumberCheck
Enter a number: 10
The number is: Positive

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac NumberCheck.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java NumberCheck
Enter a number: -20
The number is: Negative
```

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

Ans.:
```
import java.util.Scanner;

public class MinOfFour {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter four numbers: ");

        int a = sc.nextInt();

        int b = sc.nextInt();

        int c = sc.nextInt();

        int d = sc.nextInt();

        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 minimum number is: " + min);

           sc.close();
```

```
  }

}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac MinOfFour.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java MinOfFour
Enter four numbers: 8369
23
58
123
The minimum number is: 23
```

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

Ans.:
```
import java.util.Scanner;

public class StudentResult {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter the student's percentage: ");

        double percentage = sc.nextDouble();



        String result = (percentage >= 40) ? "Pass" : "Fail";

        System.out.println("Result: " + result);

                sc.close();

   }

}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac StudentResult.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java StudentResult
Enter the student's percentage: 90
Result: Pass

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac StudentResult.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java StudentResult
Enter the student's percentage: 35
Result: Fail
```

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

Ans.:
```
import java.util.Scanner;

public class CharacterCheck {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
```

```
        System.out.print("Enter a character: ");
        char ch = sc.next().charAt(0);
        String result = (ch >= 'A' && ch <= 'Z') ? "Uppercase Letter" :
                        (ch >= 'a' && ch <= 'z') ? "Lowercase Letter" :
                        "Not a Letter";
        System.out.println("The character is: " + result);
    sc.close();

        }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac CharacterCheck.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java CharacterCheck
Enter a character: h
The character is: Lowercase Letter

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac CharacterCheck.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java CharacterCheck
Enter a character: B
The character is: Uppercase Letter
```

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

Ans.: import java.util.Scanner;

```
public class AbsoluteValue1 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = sc.nextInt();
        int absValue = (num < 0) ? -num : num;
        System.out.println("Absolute value: " + absValue);
     sc.close();
        }
```

```
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac AbsoluteValue1.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java AbsoluteValue1
Enter a number: 3
Absolute value: 3

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac AbsoluteValue1.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java AbsoluteValue1
Enter a number: -6
Absolute value: 6
```

5. Miscellaneous Operator Questions

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

```
import java.util.Scanner;

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

        System.out.println("Incremented value: " + incrementedValue);
     sc.close();
         }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac IncrementNumber.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java IncrementNumber
Enter a number: 5
Incremented value: 6
```

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

```
public class Calculator {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter first number: ");
        double num1 = scanner.nextDouble();

        System.out.print("Enter an operator (+, -, *, /): ");
        char operator = scanner.next().charAt(0);

        System.out.print("Enter second number: ");
        double num2 = scanner.nextDouble();

        double result;

        switch (operator) {
            case '+':
                result = num1 + num2;
                break;
            case '-':
                result = num1 - num2;
                break;
            case '*':
                result = num1 * num2;
                break;
```

```
            case '/':
                if (num2 != 0)
                    result = num1 / num2;
                else {
                    System.out.println("Error: Division by zero");
                    return;
                }
                break;
            default:
                System.out.println("Invalid operator");
                return;
        }

        System.out.println("Result: " + result);
        scanner.close();
    }
}
```

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java Calculator
Enter first number: 10
Enter an operator (+, -, *, /): *
Enter second number: 20
Result: 200.0

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac Calculator.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java Calculator
Enter first number: 6
Enter an operator (+, -, *, /): /
Enter second number: 8
Result: 0.75
```

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

Ans.:

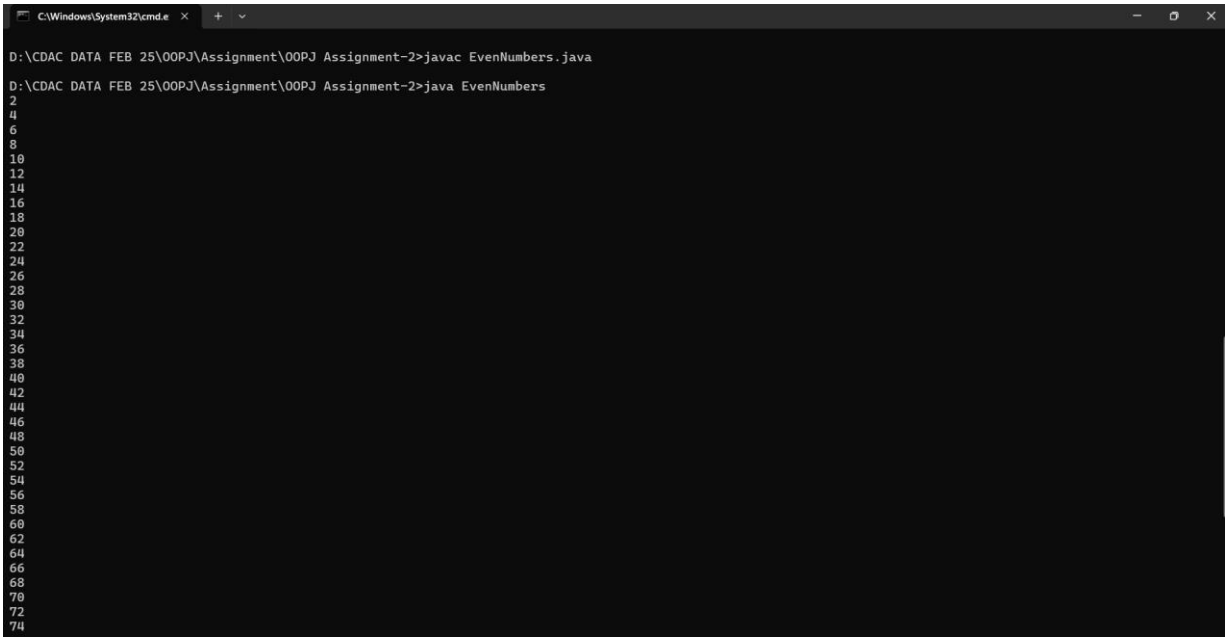
```
import java.util.Scanner;

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

        System.out.print("Enter a number: ");
        int num = scanner.nextInt();

        System.out.println((num & 1) == 0 ? "Even" : "Odd");

        scanner.close();
    }
}
```

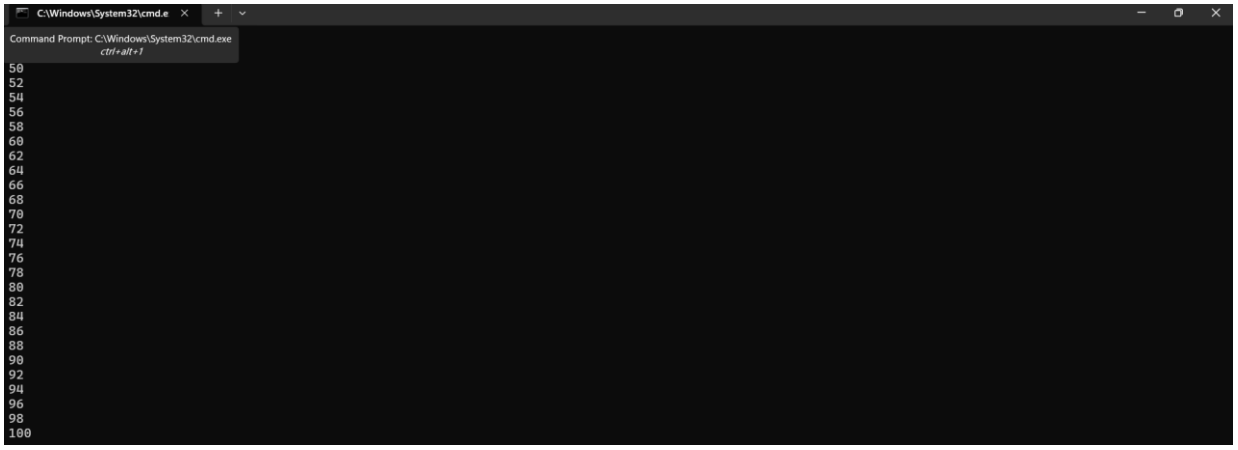
```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac OddEvenCheck.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java OddEvenCheck
Enter a number: 5
Odd

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac OddEvenCheck.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java OddEvenCheck
Enter a number: 8
Even
```

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

Ans.:

```
public class EvenNumbers {

    public static void main(String[] args) {

        for (int i = 1; i <= 100; i++) {

            if ((i & 1) == 0) {

                System.out.println(i);

            }

        }

    }

}
```





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; }`

Ans.:

```
import java.util.Scanner;

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

        System.out.print("Enter an integer: ");
        int num = scanner.nextInt();
        int rev = 0;
        while (num != 0) {
            rev = rev * 10 + num % 10;
            num /= 10;
        }

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

```
D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>javac ReverseInteger.java

D:\CDAC DATA FEB 25\OOPJ\Assignment\OOPJ Assignment-2>java ReverseInteger
Enter an integer: 63
Reversed number: 36
```