7.Write menu driven program to implement recursive Functions for following tasks. a) To find GCD and LCM b) To print n Fibonacci numbers c) To find reverse of number d) To solve 1 +2+3+4+........+(n- l )+n

**Code:-**

**import java.util.Scanner;**

**public class RecursiveFunctionsMenu {**

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

**Scanner scanner = new Scanner(System.in);**

**int choice;**

**do {**

**System.out.println("\nMenu:");**

**System.out.println("1. Find GCD and LCM");**

**System.out.println("2. Print n Fibonacci numbers");**

**System.out.println("3. Find reverse of a number");**

**System.out.println("4. Solve 1 + 2 + 3 + ... + n");**

**System.out.println("5. Exit");**

**System.out.print("Enter your choice: ");**

**choice = scanner.nextInt();**

**switch (choice) {**

**case 1:**

**System.out.print("Enter first number: ");**

**int num1 = scanner.nextInt();**

**System.out.print("Enter second number: ");**

**int num2 = scanner.nextInt();**

**System.out.println("GCD: " + findGCD(num1, num2));**

**System.out.println("LCM: " + findLCM(num1, num2));**

**break;**

**case 2:**

**System.out.print("Enter the number of Fibonacci numbers to print: ");**

**int n = scanner.nextInt();**

**System.out.println("Fibonacci Series:");**

**for (int i = 0; i < n; i++) {**

**System.out.print(fibonacci(i) + " ");**

**}**

**break;**

**case 3:**

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

**int numToReverse = scanner.nextInt();**

**System.out.println("Reverse: " + reverseNumber(numToReverse));**

**break;**

**case 4:**

**System.out.print("Enter the value of n: ");**

**int valueOfN = scanner.nextInt();**

**System.out.println("Sum: " + solveSum(valueOfN));**

**break;**

**case 5:**

**System.out.println("Exiting the program. Thank you!");**

**break;**

**default:**

**System.out.println("Invalid choice. Please try again.");**

**}**

**} while (choice != 5);**

**scanner.close();**

**}**

**// Recursive function to find GCD**

**private static int findGCD(int num1, int num2) {**

**if (num2 == 0) {**

**return num1;**

**}**

**return findGCD(num2, num1 % num2);**

**}**

**// Recursive function to find LCM**

**private static int findLCM(int num1, int num2) {**

**return (num1 \* num2) / findGCD(num1, num2);**

**}**

**// Recursive function to find Fibonacci series**

**private static int fibonacci(int n) {**

**if (n <= 1) {**

**return n;**

**}**

**return fibonacci(n - 1) + fibonacci(n - 2);**

**}**

**// Recursive function to find reverse of a number**

**private static int reverseNumber(int num) {**

**if (num < 10) {**

**return num;**

**}**

**int lastDigit = num % 10;**

**int remainingDigits = num / 10;**

**int reversed = reverseNumber(remainingDigits);**

**return (int) (lastDigit \* Math.pow(10, Math.floor(Math.log10(reversed)) + 1) + reversed);**

**}**

**// Recursive function to solve 1 + 2 + 3 + ... + n**

**private static int solveSum(int n) {**

**if (n == 1) {**

**return 1;**

**}**

**return n + solveSum(n - 1);**

**}**

**}**