Name: **Shaheer Javaid** Roll no: **2023F-BSE-238** Section: **E**

LAB NO 3

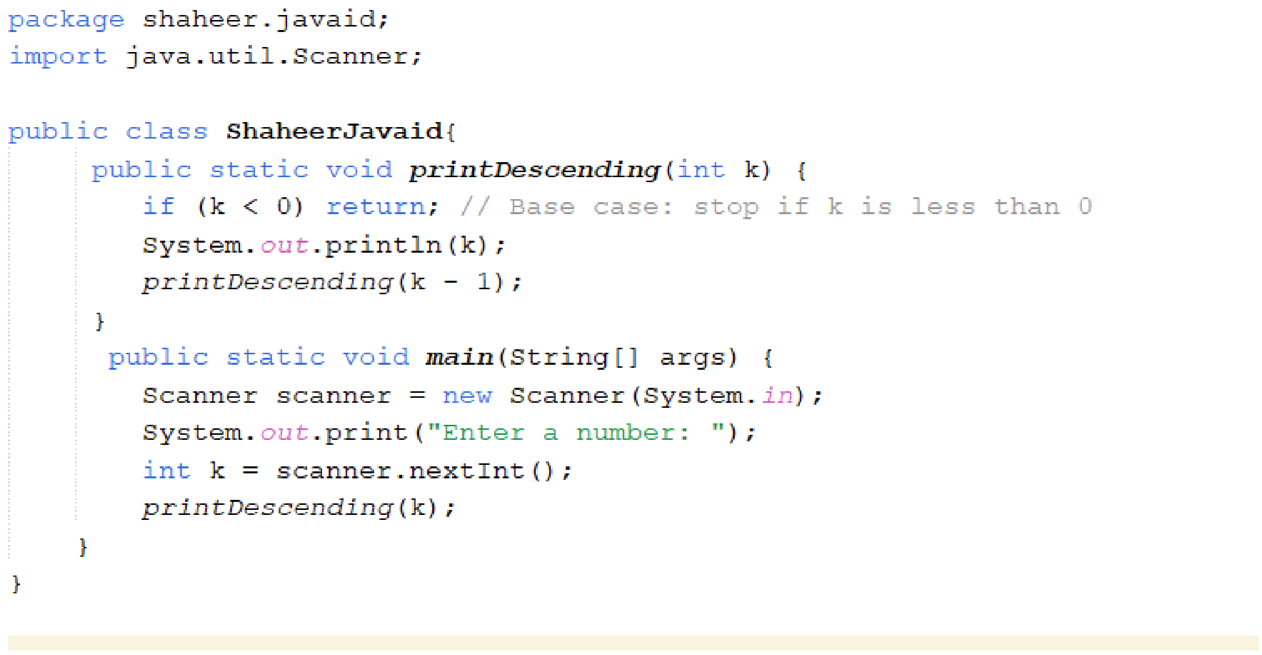
DATA STRUCTURES AND ALGORITHMS

OBJECTIVE: To understand the complexities of the recursive functions and a way to reduce these complexities.

TASK NO 1:

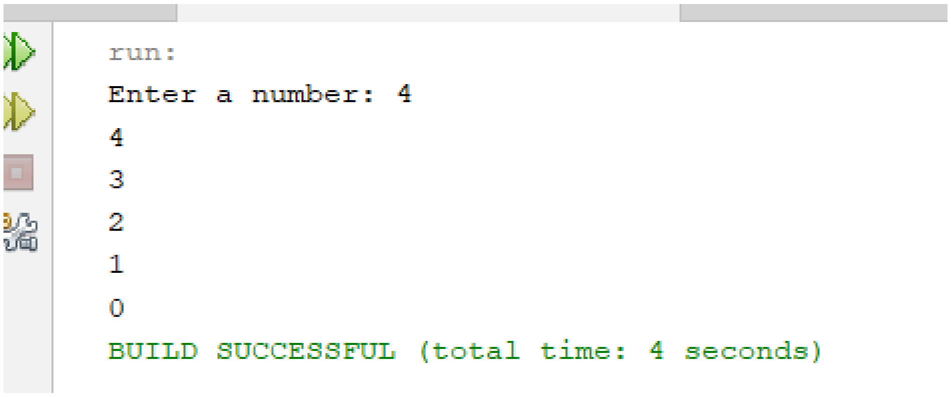
Write a program which takes an integer value (k) as input and prints the sequence of numbers from k to 0 in descending order.

INPUT:



```
package shaheer.javaid;
import java.util.Scanner;

public class ShaheerJavaid{
     public static void printDescending(int k) {
        if (k < 0) return; // Base case: stop if k is less than 0
        System.out.println(k);
        printDescending(k - 1);
     }
      public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int k = scanner.nextInt();
        printDescending(k);
    }
}
```

OUTPUT:

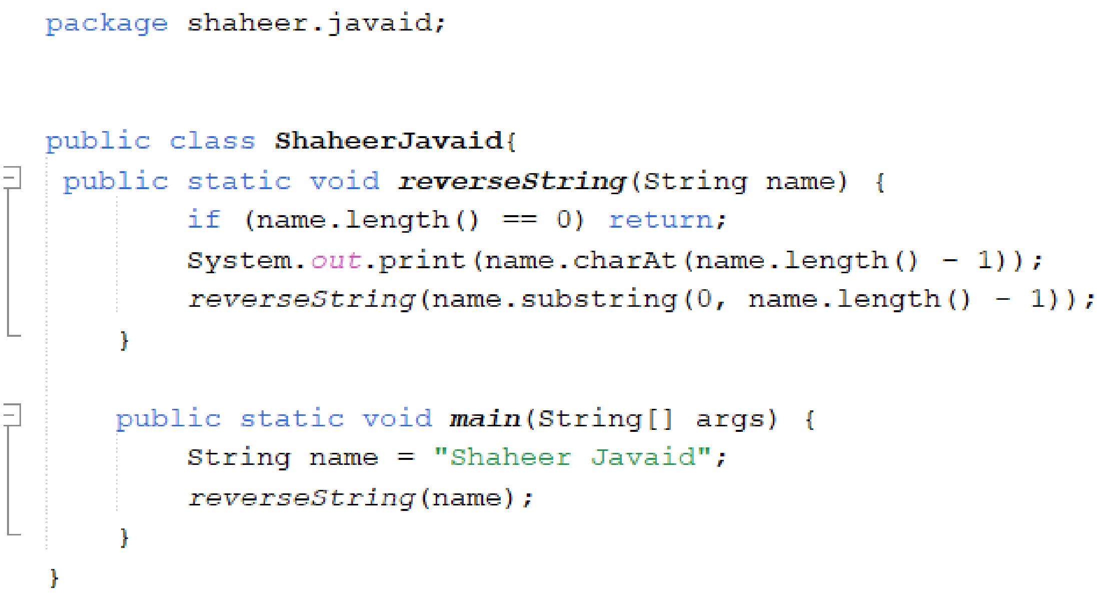


```
run:
Enter a number: 4
4
3
2
1
0
BUILD SUCCESSFUL (total time: 4 seconds)
```

TASK NO 2:

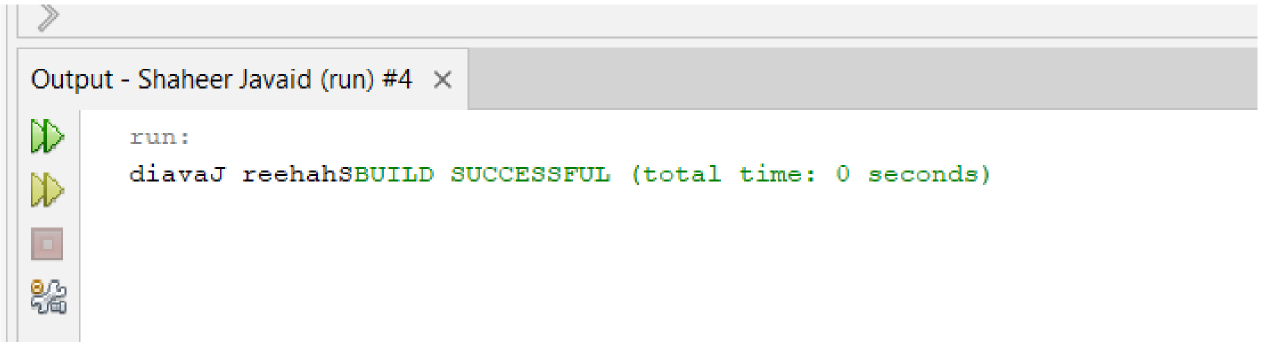
. Write a program to reverse your full name using Recursion.

INPUT:



```java
package shaheer.javaid;


public class ShaheerJavaid{
 public static void reverseString(String name) {
        if (name.length() == 0) return;
        System.out.print(name.charAt(name.length() - 1));
        reverseString(name.substring(0, name.length() - 1));
    }

    public static void main(String[] args) {
        String name = "Shaheer Javaid";
        reverseString(name);
    }
}
```

OUTPUT:

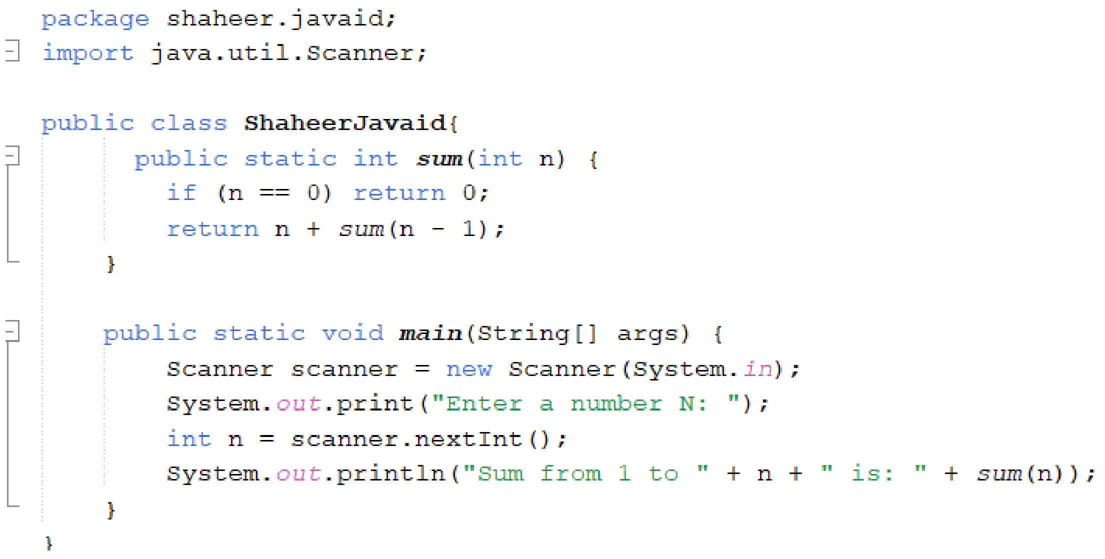


```
run:
diavaJ reehahSBUILD SUCCESSFUL (total time: 0 seconds)
```

TASK NO 3:

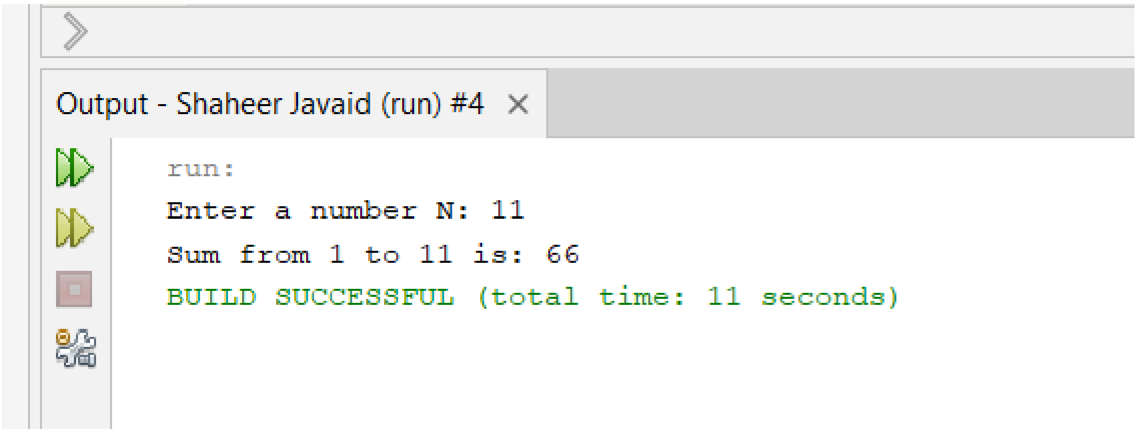
Write a program to calculate the sum of numbers from 1 to N using recursion. N should be user input.

INPUT:



```java
package shaheer.javaid;
import java.util.Scanner;

public class ShaheerJavaid{
      public static int sum(int n) {
        if (n == 0) return 0;
        return n + sum(n - 1);
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number N: ");
        int n = scanner.nextInt();
        System.out.println("Sum from 1 to " + n + " is: " + sum(n));
    }
}
```

OUTPUT:

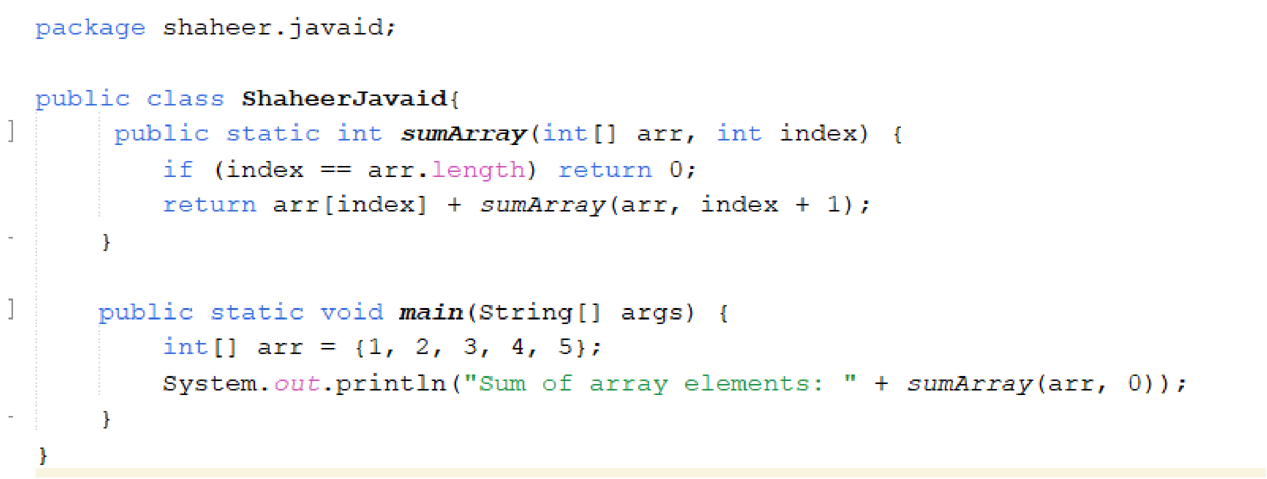


```
run:
Enter a number N: 11
Sum from 1 to 11 is: 66
BUILD SUCCESSFUL (total time: 11 seconds)
```

TASK NO 4:

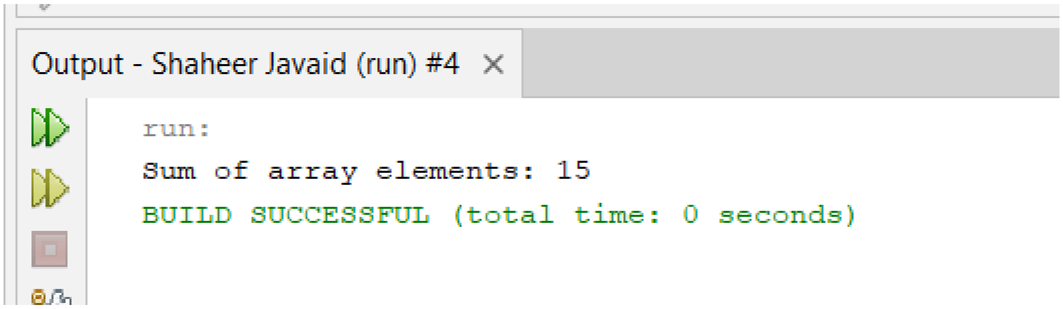
Write a recursive program to calculate the sum of elements in an array.

INPUT:



```
package shaheer.javaid;

public class ShaheerJavaid{
        public static int sumArray(int[] arr, int index) {
            if (index == arr.length) return 0;
            return arr[index] + sumArray(arr, index + 1);
        }

        public static void main(String[] args) {
            int[] arr = {1, 2, 3, 4, 5};
            System.out.println("Sum of array elements: " + sumArray(arr, 0));
        }
}
```

OUTPUT:



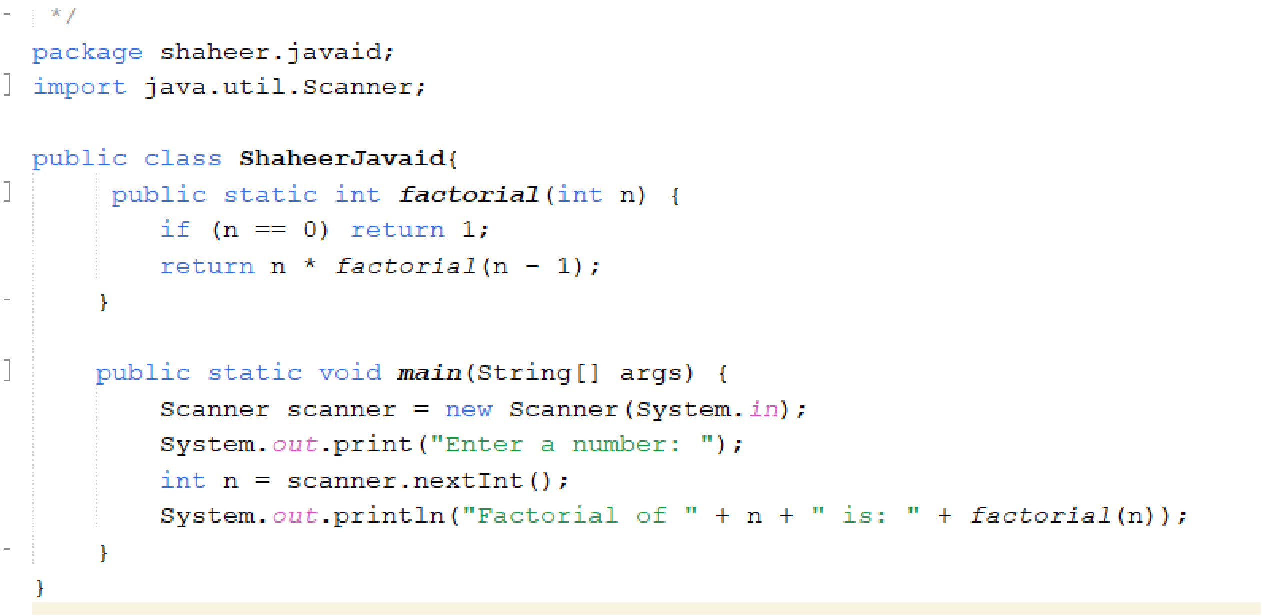
```
run:
Sum of array elements: 15
BUILD SUCCESSFUL (total time: 0 seconds)
```

:

TASK NO 5

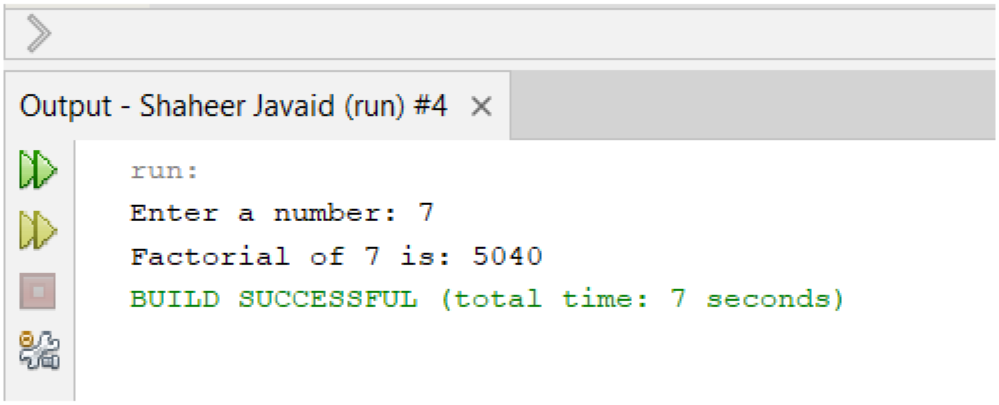
Write a recursive program to calculate the factorial of a given integer n

INPUT:



```
 */
package shaheer.javaid;
import java.util.Scanner;

public class ShaheerJavaid{
     public static int factorial(int n) {
        if (n == 0) return 1;
        return n * factorial(n - 1);
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int n = scanner.nextInt();
        System.out.println("Factorial of " + n + " is: " + factorial(n));
    }
}
```

OUTPUT:

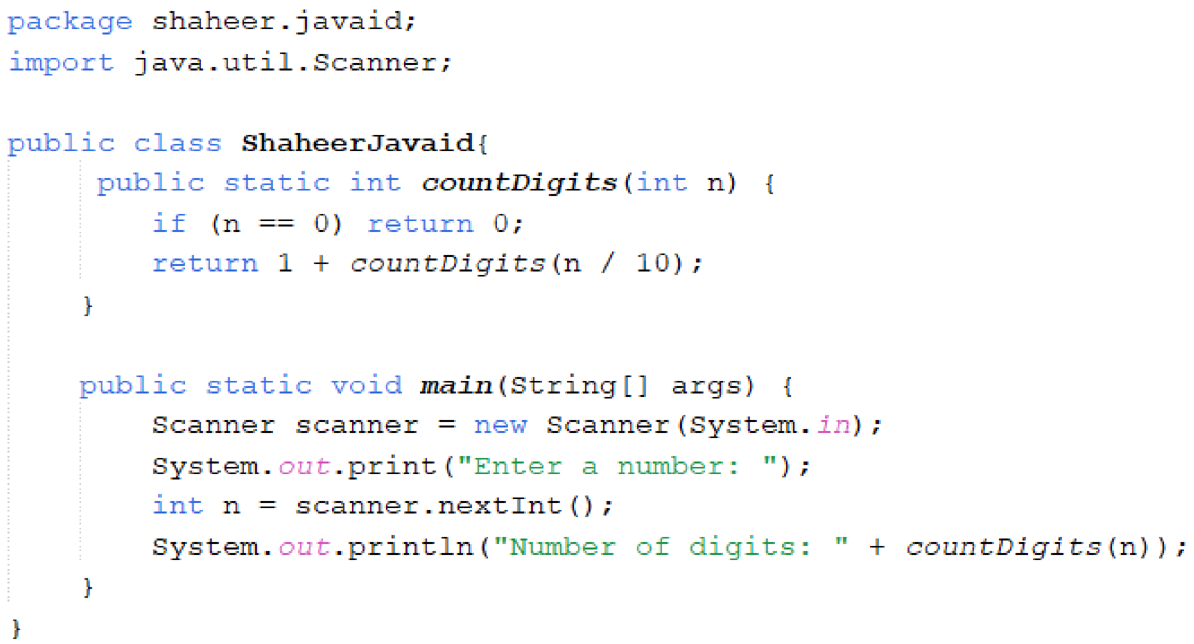


```
run:
Enter a number: 7
Factorial of 7 is: 5040
BUILD SUCCESSFUL (total time: 7 seconds)
```

TASK NO 6:

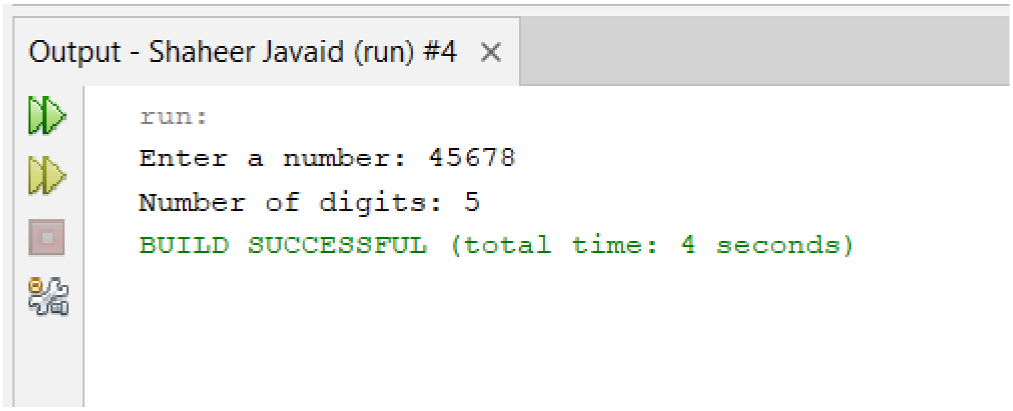
Write a program to count the digits of a given number using recursion.

INPUT :



```
package shaheer.javaid;
import java.util.Scanner;

public class ShaheerJavaid{
    public static int countDigits(int n) {
        if (n == 0) return 0;
        return 1 + countDigits(n / 10);
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int n = scanner.nextInt();
        System.out.println("Number of digits: " + countDigits(n));
    }
}
```

OUTPUT:



```
run:
Enter a number: 45678
Number of digits: 5
BUILD SUCCESSFUL (total time: 4 seconds)
```

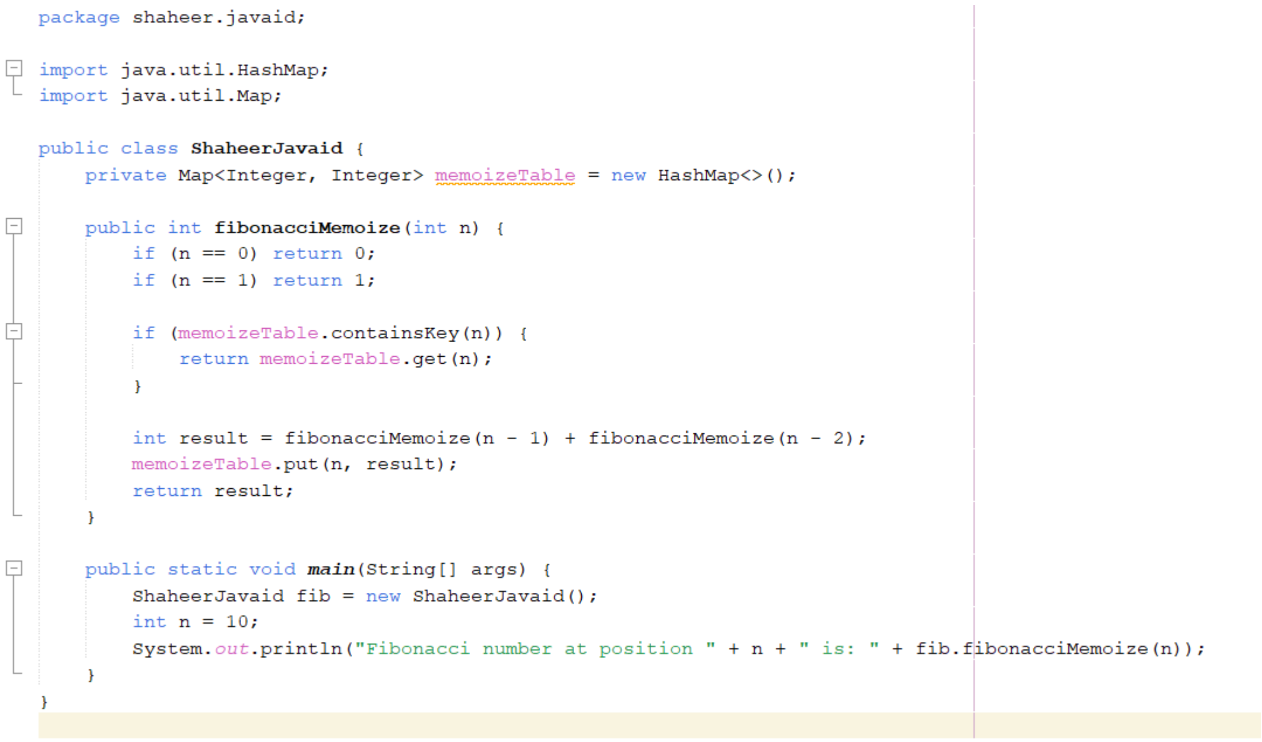
<u>HOME TASKS</u>

TASK NO 1:

Write a java program to find the N-th term in the Fibonacci series using Memoization.

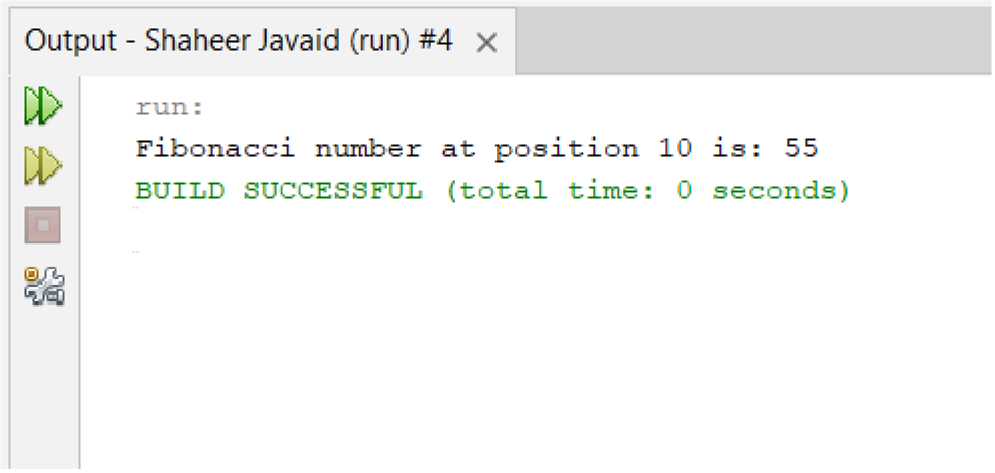
INPUT:

:



```
package shaheer.javaid;

import java.util.HashMap;
import java.util.Map;

public class ShaheerJavaid {
    private Map<Integer, Integer> memoizeTable = new HashMap<>();

    public int fibonacciMemoize(int n) {
        if (n == 0) return 0;
        if (n == 1) return 1;

        if (memoizeTable.containsKey(n)) {
            return memoizeTable.get(n);
        }

        int result = fibonacciMemoize(n - 1) + fibonacciMemoize(n - 2);
        memoizeTable.put(n, result);
        return result;
    }

    public static void main(String[] args) {
        ShaheerJavaid fib = new ShaheerJavaid();
        int n = 10;
        System.out.println("Fibonacci number at position " + n + " is: " + fib.fibonacciMemoize(n));
    }
}
```

OUTPUT:

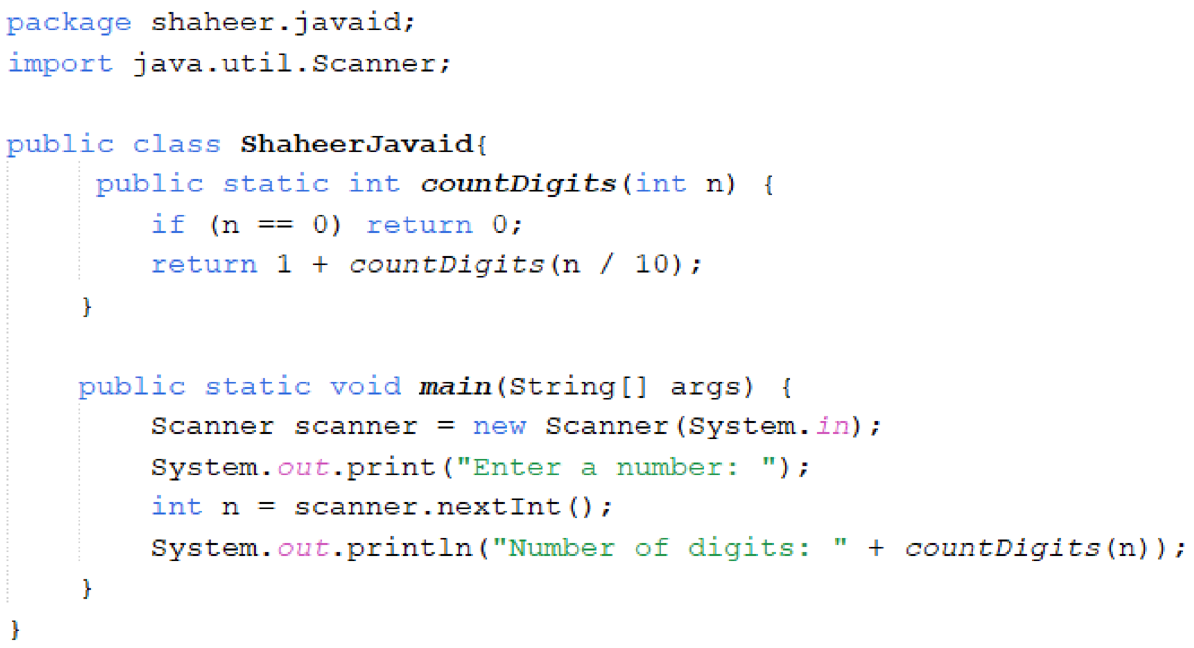


```
run:
Fibonacci number at position 10 is: 55
BUILD SUCCESSFUL (total time: 0 seconds)
```

TASK NO 2

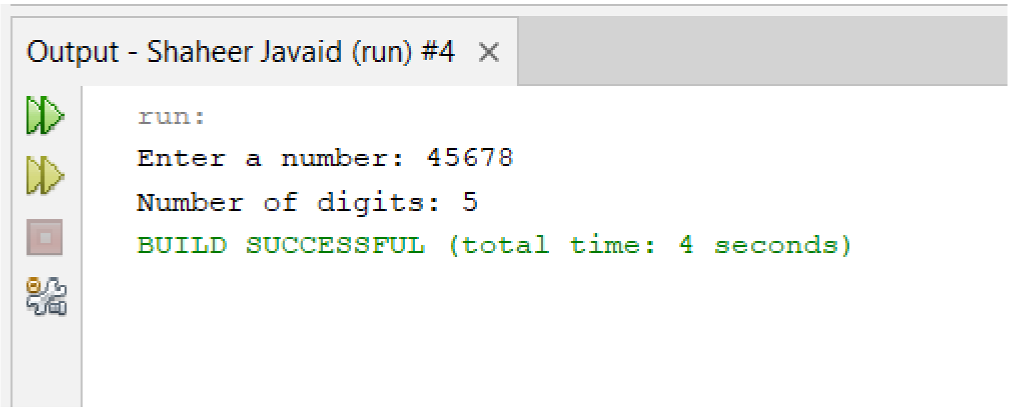
Write a program to count the digits of a given number using recursion.

INPUT:



```
package shaheer.javaid;
import java.util.Scanner;

public class ShaheerJavaid{
    public static int countDigits(int n) {
        if (n == 0) return 0;
        return 1 + countDigits(n / 10);
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int n = scanner.nextInt();
        System.out.println("Number of digits: " + countDigits(n));
    }
}
```

OUTPUT:



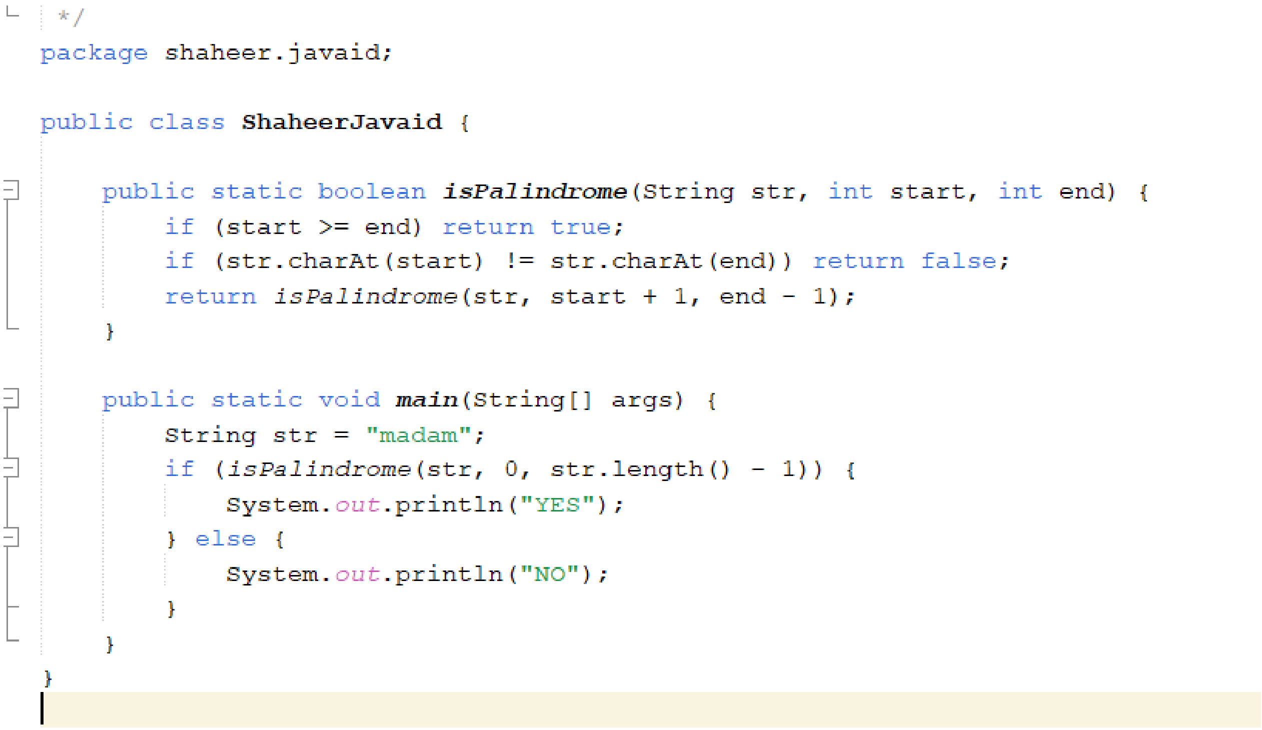
```
run:
Enter a number: 45678
Number of digits: 5
BUILD SUCCESSFUL (total time: 4 seconds)
```

TASK NO 3:

Write a java program to check whether a given string is a palindrome or not. A palindrome is a string that reads the same forwards and backwards.Print "YES" if the string is a palindrome, otherwise print "NO".

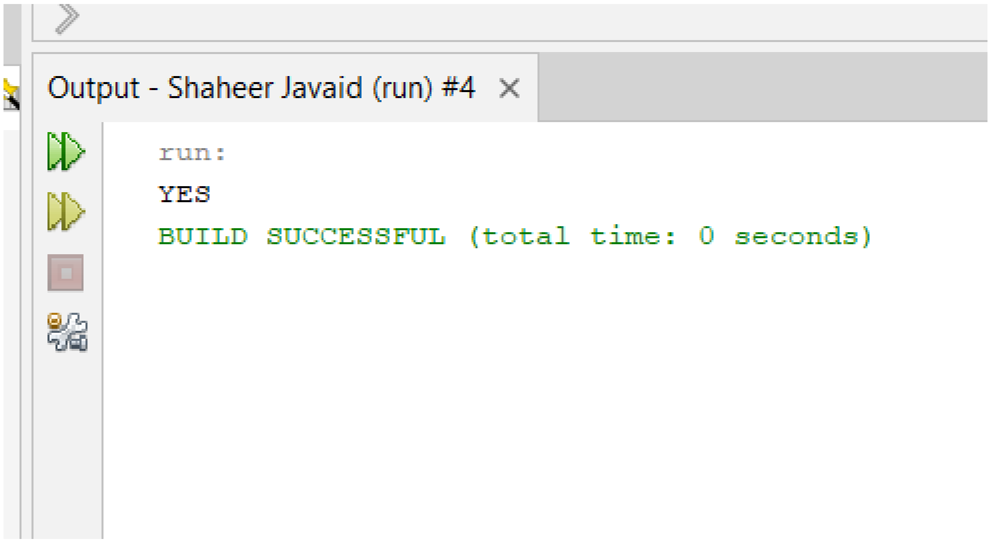
.INPUT:

:



```
 */
package shaheer.javaid;

public class ShaheerJavaid {

    public static boolean isPalindrome(String str, int start, int end) {
        if (start >= end) return true;
        if (str.charAt(start) != str.charAt(end)) return false;
        return isPalindrome(str, start + 1, end - 1);
    }

    public static void main(String[] args) {
        String str = "madam";
        if (isPalindrome(str, 0, str.length() - 1)) {
            System.out.println("YES");
        } else {
            System.out.println("NO");
        }
    }
}
```

OUTPUT:

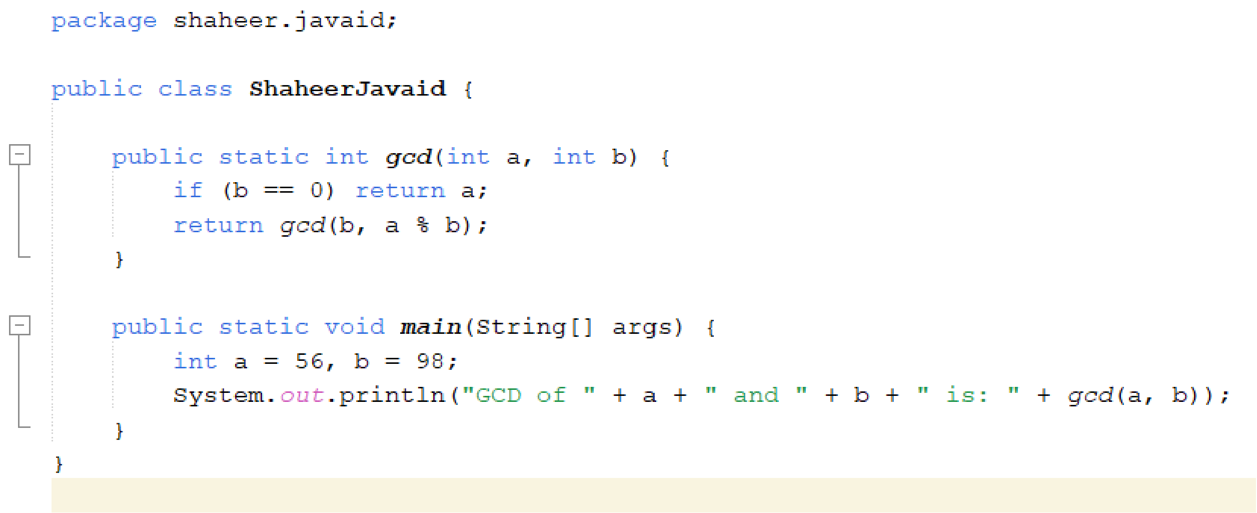


```
run:
YES
BUILD SUCCESSFUL (total time: 0 seconds)
```

TASK NO 4:

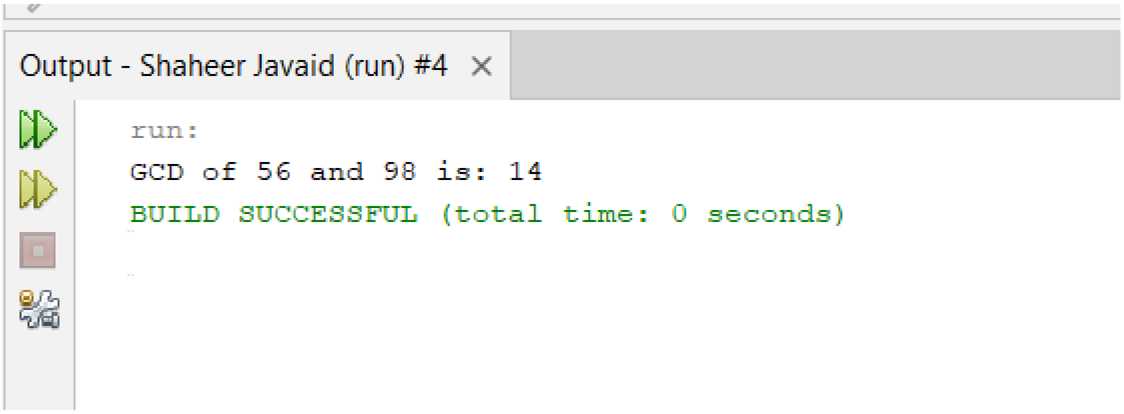
Write a recursive program to find the greatest common divisor (GCD) of two numbers using Euclid's algorithm.

INPUT:



```java
package shaheer.javaid;

public class ShaheerJavaid {

    public static int gcd(int a, int b) {
        if (b == 0) return a;
        return gcd(b, a % b);
    }

    public static void main(String[] args) {
        int a = 56, b = 98;
        System.out.println("GCD of " + a + " and " + b + " is: " + gcd(a, b));
    }
}
```

OUTPUT:



```
run:
GCD of 56 and 98 is: 14
BUILD SUCCESSFUL (total time: 0 seconds)
```