**Lab # 03**

**Lab Tasks:**

**Task #1: W**rite **a program which takes an integer value (k) as input and prints the sequence of numbers from k to 0 in descending order.**

**Code:**

package dsa\_lab;

import java.util.Scanner;

public class DSA\_Lab\_3 {

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

System.out.print("Enter Value to print Descending Sequence: ");

int k =sc.nextInt();

kToZero(k);

System.out.println();

sc.close();

}

public static void kToZero(int k){

if(k<0){

return;

}else{

System.out.print( k +" ");

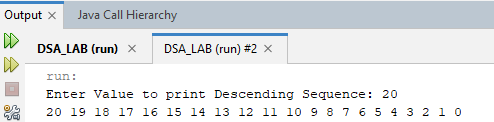
kToZero(k-1);

}

}

}

**Output:**



**Task #2: Write a program to reverse your full name using Recursion.**

**Code:**

package dsa\_lab;

import java.util.Scanner;

public class DSA\_Lab\_3 {

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

System.out.print("Enter Your Full Name: ");

String j =sc.nextLine();

reverse(j);

System.out.println();

}

public static void reverse(String s){

if(s.isEmpty()){

return;

}else{

reverse(s.substring(1));

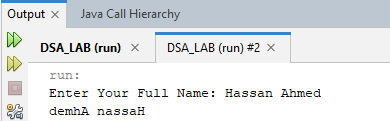
System.out.print(s.charAt(0));

}

}

}

**Output:**



**Task #3: Write a program to calculate the sum of numbers from 1 to N using recursion. N should be user input.**

**Code:**

package dsa\_lab;

import java.util.Scanner;

public class DSA\_Lab\_3 {

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

System.out.print("Enter Value of N to get its sum from 1: ");

int N =sc.nextInt();

int n=1;

System.out.println("Sum is:"+sumOneToN(n, N));

System.out.println();

sc.close();

}

public static int sumOneToN(int n,int N){

if(n==N){

return n;

}else{

return n+sumOneToN(n+1,N);

}

}

}

**Output:**

**A screenshot of a computer

Description automatically generated**

**Task #4: Write a recursive program to calculate the sum of elements in an array.**

**Code:**

package dsa\_lab;

import java.util.Scanner;

public class DSA\_Lab\_3 {

public static void main(String[] args){

int[] ar={1,5,7,45,12};

int sum=sumArray(ar,ar.length);

System.out.println("Sum of Array: "+sum);

}

public static int sumArray(int[] ary,int length){

if(length<=0){

return 0;

}else{

return ary[length-1]+sumArray(ary,length-1);

}

}

}

**Output:**

**A screenshot of a computer

Description automatically generated**

**Task #5: Write a recursive program to calculate the factorial of a given integer n**

**Code:**

package dsa\_lab;

import java.util.Scanner;

public class DSA\_Lab\_3 {

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

System.out.print("Enter Value of int to get its Factorial: ");

int no =sc.nextInt();

System.out.println(fact(no));

}

public static int fact(int n){

if(n==0){

return 1;

}else{

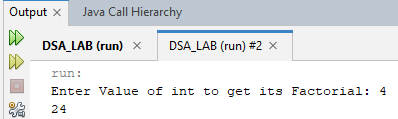
return n\*fact(n-1);

}

}

}

**Output:**



**Task #6: Write a program to count the digits of a given number using recursion. Code:**

package dsa\_lab;

import java.util.Scanner;

public class DSA\_Lab\_3 {

public static void main(String[] args){

int num=521391;

System.out.println(“Digit Count; “+countDigit(num));

}

public static int countDigit(int n){

if(n<=0){

return 0;

}else{

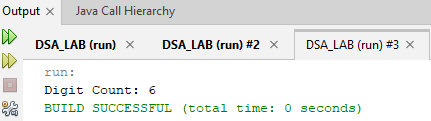
return 1+countDigit(n/10);//As it is int so last no become decimal which will be explited

}

}

}

**Output:**



**Home Tasks:**

**Task #1: Write a java program to find the N-th term in the Fibonacci series using Memoization. Code:**

import java.util.HashMap;

import java.util.Scanner;

public class FibonacciMemoization {

private static HashMap<Integer, Long> memo = new HashMap<>();

public static long fibonacci(int n) {

// Base cases

if (n <= 1) {

return n;

}

// Check if value is already computed

if (memo.containsKey(n)) {

return memo.get(n);

}

// Compute and store in the memo

long result = fibonacci(n - 1) + fibonacci(n - 2);

memo.put(n, result);

return result;

}

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the term number N to find in the Fibonacci series: ");

int n = scanner.nextInt();

long nthTerm = fibonacci(n);

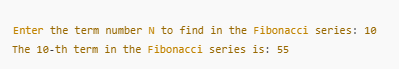
System.out.println("The " + n + "-th term in the Fibonacci series is: " + nthTerm);

scanner.close();

}

}

**Output:**

****

**Task #2: . Write a program to count the digits of a given number using recursion.**

**Code:**

package dsa\_lab;

import java.util.Scanner;

public class DSA\_Lab\_3 {

public static void main(String[] args){

int num=521391;

System.out.println(“Digit Count; “+countDigit(num));

}

public static int countDigit(int n){

if(n<=0){

return 0;

}else{

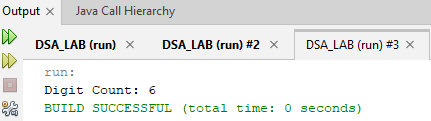
return 1+countDigit(n/10);//As it is int so last no become decimal which will be explited

}

}

}

**Output:**



**Task #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"**

**Code:**

package dsa\_lab;

import java.util.Scanner;

public class DSA\_Lab\_3 {

public static void main(String[] args){

String pl="BoB";

System.out.println(pl+" is palindrome: "+palinDrome(pl));

}

public static String palinDrome(String p){

if(p.length()<=1) {

return "Yes";

}

if(p.charAt(0)==p.charAt(p.length() -1)){

return palinDrome(p.substring(1, p.length() - 1));

}

else{

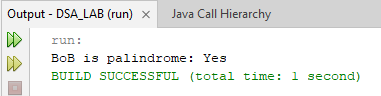
return "No";

}

}

}

**Output:**

****

**Task #4: Write a recursive program to find the greatest common divisor (GCD) of two numbers using Euclid's algorithm**

**Code:**

package dsa\_lab;

import java.util.Scanner;

public class DSA\_Lab\_3 {

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

System.out.println("Provide 2 numbers to get there gcd:");

int n1=sc.nextInt();

int n2=sc.nextInt();

System.out.println("gcd("+n1+","+n2+"): "+gcd(n1,n2)); }

public static int gcd(int n,int n1){

int max= Math.max(n, n1);

int min= Math.min(n, n1);

if(min==0){

return max;

}

int temp=min;

min=max%temp;

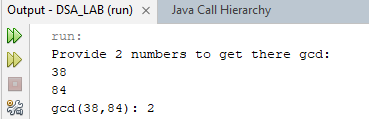
max=temp;

return (gcd(max,min));

}

}

**Output:**

****