

\_\_\_\_abdul Quddos\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_69984\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(

Reg. No.

)

Sign

~~e~~

d

Remarks

~~:~~

Sc

~~o~~

~~r~~

~~e~~

~~:~~

I

L

S

1

27

/

1

2

/

7

2

B~~a~~hr~~ia~~ Un~~i~~vers~~i~~ty,



K~~a~~r~~a~~chi C~~a~~mpus

LAB EXPERIMENT NO.

\_\_\_\_\_\_\_1\_\_\_\_\_\_\_

|  |  |
| --- | --- |
| **TASK NO** | **OBJECTIVE** |
| **1** | Write a program which calculates the square of a number using odd number series implemented with the help of recursion concept |
| **2** | Write a program which takes input of an integer number and returns the sum of all numbers. i.e., if input is 3453 then the output should be 15 (3+4+5+3). |
| **3** | Write a program to calculate binomial coefficients of any given number using recursion. |
| **4** | Calculation of number of moves for N number of disk in Tower of Hanoi problem using recursion. |
| **5** | Write a program to calculate H.C.F of two numbers, using recursion. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Submitted On~~:~~

# \_\_\_\_\_14/10/2022\_\_\_\_\_\_

(Date~~:~~ DD/MM/YY)

**Task# 01: - Write a program which calculates the square of a number using odd number series implemented with the help of recursion concept**

**Solution:** static void Main(string[] args)

{

Console.WriteLine("enter limit for printing the series"); int max = int.Parse(Console.ReadLine());

Console.WriteLine("odd series : "+ oddseries(1,max));

Console.ReadKey();

}

public static string oddseries(int n,int max= 100)

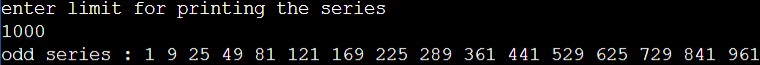
{ if (n\*n>max)

{

return " ";

} return n \* n + " " + oddseries(n + 2,max); }

**Output: -**



**Task# 02: - Write a program which takes input of an integer number and returns the sum of all numbers. i.e., if input is 3453 then the output should be 15 (3+4+5+3).**

**Solution:** static int sum\_of\_digit(int num)

{ if (num == 0)

return 0;

return (num % 10 + sum\_of\_digit(num / 10));

}

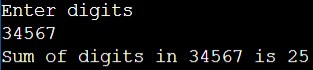
public static void Main()

{

Console.WriteLine("Enter digits"); int number = int.Parse(Console.ReadLine()); int res = sum\_of\_digit(number);

Console.WriteLine("Sum of digits in " + number + " is " + res); }

**Output: -**



**Task# 03: - Write a program to calculate binomial coefficients of any given number using recursion.**

**Solution:** static int binomialCoeff(int num, int kth)

{

if (kth == 0 || kth == num)

return 1;

return binomialCoeff(num - 1, kth - 1) + binomialCoeff(num - 1, kth);

}

public static void Main()

{

int num = 10, kth = 4;

Console.Write("Value of C(" + num + "," + kth + ") is " + binomialCoeff(num, kth));

}

**Output: -**



**Task# 04: - Calculation of number of moves for N number of disk in Tower of Hanoi problem using recursion.**

**Solution:** static void Main(string[] args)

{ Console.WriteLine("Enter No of discs"); int dscs = int.Parse(Console.ReadLine());

Console.WriteLine("no of moves are : "+moves(dscs));

}

public static int moves(int dsc)

{

if (dsc == 1)

{

return 1;

}

else

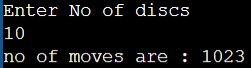
{

return 2 \* moves(dsc - 1) + 1;

}

}

**Output: -**



**Task# 05: - Write a program to calculate H.C.F of two numbers, using recursion.**

**Solution:** static void Main(string[] args)

{

Console.WriteLine("Enter two integers"); long num1 = Convert.ToInt64(Console.ReadLine()); long num2 = Convert.ToInt64(Console.ReadLine()); long hcf = gcd(num1, num2);

Console.WriteLine("HCF of {0} and {1} = {2}\n", num1, num2, hcf);

}

static long gcd(long num1, long num2)

{ if (num2 == 0) { return num1; }

else {

return gcd(num2, num1 % num2);

}

}

**Output: -**

