Bahria University,

Karachi Campus

A picture containing text, room

Description automatically generated

LAB EXPERIMENT NO.

\_\_\_\_**06**\_\_\_\_\_

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| 1 | Write a code which prints the following series:  2 4 8 - - - - n |
| 2 | Write a program which calculates the square of a number using odd number series implemented with the help of recursion concept. |
| 3 | 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). |
| 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. |
| 6 | Implement file code |

Submitted On

09/11/2022

(Date: DD/MM/YY)

**Task No. 1 : Write a code which prints the following series:**

**2 4 8 - - - - n**

**Solution:**

**Main Class:**

static int Recursion(int n,int data){

data \*= 2;

Console.WriteLine("The Square of 2 is "+data);

return (n <= 1) ? 0 : Recursion(n - 1, data);}

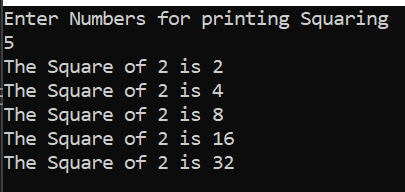
static void Main(string[] args){

Console.WriteLine("Enter Numbers for printing Squaring");

int n = int.Parse(Console.ReadLine());

Recursion(n, 1);}

**Output:**



**Task No. 2:** **Write a program which calculates the square of a number using odd number series implemented with the help of recursion concept.**

**Solution:**

**Main Class:**

static int cubic(int n,int count,int data){

data = count\*count;

Console.WriteLine(" "+count + " => "+data);

count += 2;

return n <= 1 ? 0 : cubic(n - 1, count,data);}

static void Main(string[] args){

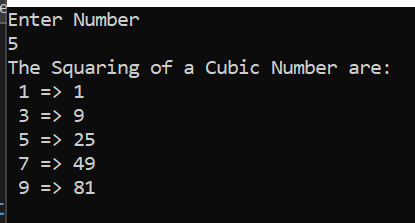
Console.WriteLine("Enter Number ");

int n = int.Parse(Console.ReadLine());

Console.WriteLine("The Squaring of a Cubic Number are: ");

cubic(n, 1,1);}

**Output:**

****

**Task No. 3:** **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:**

**Main Class:**

static int recursion(int n,int sum){

sum = n % 10 + sum;

return (n == 0) ? sum : recursion(n / 10, sum);}

static void Main(string[] args){

Console.WriteLine("Enter Number ");

int n = int.Parse(Console.ReadLine());

Console.WriteLine("The Sum of {0} Number is ",n);

Console.WriteLine(recursion(n, 0));}

**Output:**

**Text

Description automatically generated**

**Task No. 4:** **Calculation of number of moves for N number of disk in Tower of Hanoi problem using recursion.**

**Solution:**

**Main Class:**

static void towerOfHanoi(int n, char from\_rod, char to\_rod, char aux\_rod, ref int moves){

if (n == 1){

moves += 1;

Console.WriteLine("Move disk 1 from rod " + from\_rod + " to rod " + to\_rod);

return;}

towerOfHanoi(n - 1, from\_rod, aux\_rod, to\_rod, ref moves);

moves += 1;

Console.WriteLine("Move disk " + n + " from rod " + from\_rod + " to rod " +

to\_rod);

towerOfHanoi(n - 1, aux\_rod, to\_rod, from\_rod, ref moves);}

static void Main(string[] args){

int moves = 0;

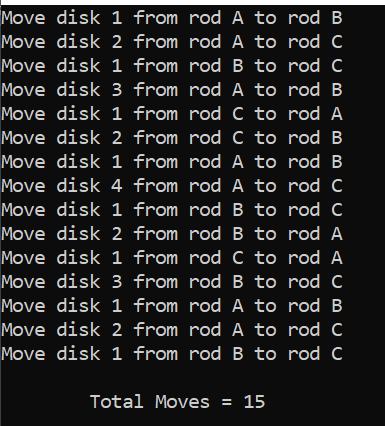
int disk = 4;

towerOfHanoi(disk, 'A', 'C', 'B', ref moves);

Console.WriteLine("\n\tTotal Moves = " + moves);

Console.ReadLine();}

**Output:**

****

**Task No. 5:** **Write a program to calculate H.C.F of two numbers, using recursion.**

**Solution:**

**Main Class:**

static int recursion(int num1,int num2){

if(num2==0 || num1 == num2){

return num1;}

else if (num1 == 0){

return num2;}

else if (num1 > num2){

return recursion(Math.Abs(num1 - num2), num2);}

else{

return recursion(num2,Math.Abs(num1 - num2));}}

static void Main(string[] args){

Console.WriteLine("Enter First Number ");

int num1 = int.Parse(Console.ReadLine());

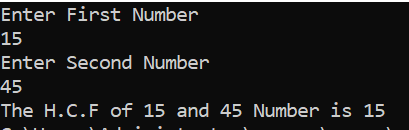
Console.WriteLine("Enter Second Number ");

int num2 = int.Parse(Console.ReadLine());

Console.Write("The H.C.F of {0} and {1} Number is ",num1,num2);

Console.Write(recursion(num1, num2));}

**Output:**

****

**Task No. 6:** **Implement file code**.

**Solution:**

**Main Class:**

static void Main(string[] args){

File file = new File();

file.SearchForFiles(@"C:\Intel");

foreach(string filename in file.result){

Console.WriteLine(filename);}}

public Dictionary<string, string> errors = new Dictionary<string, string>();

public List<string> result = new List<string>();

public void SearchForFiles(string path){

try{

foreach (string fileName in Dictionary.GetFiles(path)){

result.Add(fileName);}

foreach (string directory in Directory.GetDirectories(path))

SearchForFiles(directory);}}

catch (System.Exception ex){

errors.Add(path, ex.Message);}}

**Output:**

