Lab 2

TOPIC :RECURSIVE

Objective:

Student should be able to develop the programs using recursive concept.

Exercise 1:

Sum three number using recursive

Algorithm

Step 1: Start

Step 2: Declare variables array and n and assign variable n to 3

Step 3: Input array

Step 4: Call function sum(a, n)

Step 5: Display output from function sum(a, n)

Step 6: Stop

Function sum (a[],n) = $\begin{cases} return\ 0\ if\ n = 0 \\ a[i] + sum(a, n-1) \end{cases}$

Answer;

```
#include <iostream.h>

int sum (int[], int);

int main(){

   int Array[3] = {16, 10, 5};
   int n = 2;

   cout<<sum(Array, n);
return 0;
}

int sum (int A[], int x){

   if (x==0)
      return A[0];
     else
```

```
        return A[x]+sum(A,x-1);
}
```

Exercise 2: Print in binary, computer represent integers as sequences of bits, A bit is a single digit in the binary numbers system and have only the value 0 or 1. Write the program to convert a binary to decimal numbers.

Algorithm

Step 1: Start

Step 2 : Declare a variable n as binary number

Step 3: Input n

Step 4: Call function BiCoDe(n)

Step 5: Display output from function BiCoDe(n)

Step 6: Stop

$$\text{Function BiCoDe(n)} = \begin{cases} 0 \; if \; n = 0 \\ n\%10 + 2 * BiCoDe(\frac{n}{10}) \end{cases}$$

Answer;

```
#include <iostream.h>

int BiCoDe (int);

int main(){

   int n;
   cout<< "Input binary numbers: ";
   cin>>n;
   cout<< BiCoDe(n);
}

int BiCoDe (int x) {

   if (x==0)
      return 0;
     else
      return (x%10 + 2 * BiCoDe (x/10));
}
```

TOPIC :POINTER

Objective:

Student should be able to develop the programs using pointer.

Exercise 1: Write a program to find a larger number between two numbers using pointers.

Algorithm:

Step 1: Start the program

Step 2: Declare variables number1, number2, *ptr1, *ptr2

Step 3: Assign pointers to appropriate variables

Step 4: Get values for first number and second number

Step 5: Compare two pointer values

Step 6: Print the larger number

Step 7: Stop

Answer;

```
#include <iostream>
using namespace std;

int main()
{
  int number1, number2;
  int *xptr, *yptr;

  xptr = &number1;
  yptr = &number2;

  cout<<"Enter first number = ";
  cin>>number1;
  cout<<"Enter second number = ";
  cin>>number2;

     if(*xptr > *yptr)
       {
        cout<<"\nFirst number is the largest";
       }

     else
       {
        cout<<"\nSecond number is the largest";
       }

     return 0;
}
```

Exercise 2: Write a program to print out the characters of the reversed string using a pointer

Algorithm:

Step 1: Start the program.

Step 2: Declare variables str1, revstr, *strPtr, *revPtr

Step 3: Get a string value for str1.

Step 4: Reverse the string

Step 5: Print the reversed string

Step 7: Stop

Exercise 3: Write a program to find the largest element for five numbers using Dynamic Memory Allocation

Algorithm:

Step 1: Start the program.

Step 2: Declare the variables *element, n, i.

Step 3: Get the value for total number of elements.

Step 4: Allocate the memory for all elements

Step 5: Display the result

Step 7: Stop