INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)
Dundigal, Hyderabad – 500 043

LABORATORY WORK SHEET

Date: 28/06/2022

Roll No: 2195IA6754 Name: P. JYOTHI PRASANNA

Exp No: 03 Experiment Name: CONTROL STRUCTURES

DAY TO DAY EVALUATION:

	Preparation	Algorithm / Performance in the Lab	Source Code / Calculations and Graphs	Program Execution / Results and Error Analysis	Viva	Total
Max. Marks	4	4	4	4	4	20
Obtained	4	4	4	4	4	20

Signature of Lab I/C

START WRITING FROM HERE:

AIM: Design and develop an algorithm to find the reverse of an integer number N and check whether it is palindrome or not. Implement a C program for the developed algorithm that takes an integer number as input and output the reverse of the same with suitable messages Ex: N: 2020, Reverse: 0202, Not a palindrome

PROGRAM:

```
#include <stdio.h>
void main()
{
  int n, rev=0, rem, m;
  printf("Enter a number:");
  scanf("%d", &n);
  m=n;
  if (n<=999||n>9999)
  {
     printf("Not a 4-digit number");
     exit(0);
  }
  while(n!=0)
```

```
{
rem = n%10;
n = n/10;
rev = rev*10 + rem;
}
if (m==rev)
{
   printf ("The given number %d is Palindrome", m);
}
else
{
   printf ("The given number %d is not Palindrome", m);
}
}
```

INPUT: Enter a number: 121

OUTPUT: The given number 121 is a palindrome

b) AIM: Draw the flowchart and write c program to compute sin(x) using Taylor series approximation given by $\sin(x) = x - (x^3/3!) + (x^5/5!) - (x^7/7!) + \ldots$ Compare the result with the built-in library function and print both the results with appropriate messages.

PROGRAM:

```
#include<stdio.h>
#include<math.h>
float mysine (float x, int n);
int fact (int n);
int main ()
{
int x, n;
float rad, res;
printf ("Enter the degree and number of terms:");
scanf ("%d%d", &x, &n);
rad = (x*3.14)/180;
```

```
res = mysine (rad, n);
printf("my defined sine function (%d) = %f", x, res);
printf("using library function sine (%d) = %f", x, sin(rad));
//getch();
}
float mysine (float x, int n);
{
int i, sign = 1;
float sum = 0;
for (i=1; i<=n; i+=2)
{
sum = sum + sign*(pow(x,i) / fact(i));
sign* = -1;
/* int count, n=1, sign=1;
  for (count=1; (n<=10); count+=2)
  {
   *res += sign* (pow(num, count)/factorial(count));
   n+=1;
   sign* = -1;
  }*/
}
return sum;
}
int fact(int n);
{
if (n==0)
return 1;
return n*fact(n-1);
}
```

INPUT: Enter the degree and number of terms: 30
2

OUTPUT: my defined sine function (30) = 0.523333.
using library function sine(30) = 0.499770

c) AIM: Design and develop and algorithm and flowchart to read a 3 digit number and check whether the given number is Armstrong or not. Write a C program to implement the same & also display the armstrong no.'s between 1 to 1000.

PROGRAM:

```
# include <stdio.h>
int main()
{
int num, originalnum, remainder, result = 0;
printf ("Enter a three-digit integer: ");
scanf ("%d", &num);
original_num = num;
while (original_num != 0){
{    // remainder contains the last digit
remainder = original_num % 10;
result += remainder * remainder * remainder;
// remaining last digit from the original number
original_num /= 10;
}
if (result == num)
    printf ("%d is an Armstrong number.", num);
else
    printf ("%d is not an Armstrong number", num);
return 0;
}
```

INPUT: Enter a three digit number : 153

OUTPUT: 153 is an Armstrong number.

d. Design and develop an algorithm for evaluating the polynomial $f(x) = a_4x^4 + a_3x^3 + a_2x^2 + a_1x^1 + a_0$, for a given value of x as its co-efficients using Horner's method. Implement a c program for the same & execute the program for different sets of values of co-efficients & x.

```
// Preprocessor directions
#include <stdio.h>
// Main program
main () {
float coeff [10], x, fx = 0;
int i;
printf ("Program to evaluate the given polynomial f(x) =
                    a4x^4 + a3x^3 + a2x^2 + a1x^1 + a0 for ");
printf ("Given value of x and the co-efficients using Horner's
                                                    method);
printf ("Enter the co-efficients (integer / float) of a given polynomial
                                    (i.e., a0, a1, a2, a3, a4 ... "));
for (i=0; i<=4; i++) {
printf ("coeff [%d] = ", i);
scanf ("%f", &coeff [i]);
}
printf ("Enter the value of x (integer or float) ... : ");
scanf ("%f", &x);
for (i=4; i>=0; i--) {
fx = fx * x + coeff [i];
printf ("f(%f) = %f ", x, fx);
}
}
```

INPUT: Enter the value of x (integer or float) a: 2

OUTPUT: f(2.000000) = 0.000000