Objective(s):

   To understand function programming, its types and function-call.

Program : Write a C program to add two numbers using used defined function

Code:
```c
#include <stdio.h>
int addNumbers(int a, int b);        // function prototype

int main()
{
   int n1,n2,sum;

   printf("Enters two numbers: ");
   scanf("%d %d",&n1,&n2);

   sum = addNumbers(n1, n2);      // function call
   printf("sum = %d",sum);

   return 0;
}

int addNumbers(int a, int b)        // function definition
{
   int result;
   result = a+b;
   return result;               // return statement
}
```

Program:

Write a program to add, subtract, multiply and divide two integers using user- defined
type function with return type.

Code:

```c
#include<stdio.h>

// functions declaration
int add(int n1, int n2);
int subtract(int n1, int n2);
int multiply(int n1, int n2);
```

```c
int divide(int n1, int n2);

// main function
int main()
{
  int num1, num2;

  printf("Enter two numbers: ");
  scanf("%d %d", &num1, &num2);

  printf("%d + %d = %d\n", num1, num2, add(num1, num2));
  printf("%d - %d = %d\n", num1, num2, subtract(num1, num2));
  printf("%d * %d = %d\n", num1, num2, multiply(num1, num2));
  printf("%d / %d = %d\n", num1, num2, divide(num1, num2));

  return 0;
}

// function to add two integer numbers
int add(int n1, int n2)
{
  int result;
  result = n1 + n2;
  return result;
}

// function to subtract two integer numbers
int subtract(int n1, int n2)
{
  int result;
  result = n1 - n2;
  return result;
}

// function to multiply two integer numbers
int multiply(int n1, int n2)
{
  int result;
  result = n1 * n2;
  return result;
}

// function to divide two integer numbers
int divide(int n1, int n2)
```

```
{
 int result;
 result = n1 / n2;
 return result;
}
```

Program: Write a program to calculate factorial of a number using recursion.

Code:

```
#include<stdio.h>
long factorial(int);                //Function declaration int
main()
{
    int num;
    long fact;
    printf("Enter a number to find factorial: \n"); scanf("%d", &num);
    if(num<0)
        printf("Factorial of negative no. is not defined. \n"); else
        {
            fact = factorial(num); printf("%d!=%d \n",
            num, fact);
        }
return 0;
}
//Function definition long
factorial(int num)
{
            if(num==0)
                     return 1;
    else
                return(num*factorial(num-1));
}
```

*(Students are to code the following programs in the lab and show the output
to instructor/course Teacher)*

Instructions

- *Write comment to make your programs readable.*
- *Use descriptive variables in your programs(Name of the variables should show their purposes)*

Programs List

1. Write a program to add, subtract, multiply and divide two integers using user-defined type function with return type.
2. Write a C program to find maximum and minimum between two numbers using functions.
3. Write a C program to check whether a number is even or odd using functions.
4. Write a C program to check whether a number is prime, Armstrong or perfect number using functions.
5. Write a C program to find power of any number using recursion.
6. Write a program to calculate sum of first 20 natural numbers using recursive function.
7. Write a program to generate Fibonacci series without and with recursive function.
8. Write a program to swap two integers using call by value and call by reference methods of passing arguments to a function.
9. Write a program to find sum of digits of the number using Recursive Function.
10. Write a program to read an integer number and print the reverse of that number using recursion.