RAMAIAH
Institute of Technology

Department of Computer Science Engineering

LABORATORY MANUAL

SUBJECT CODE: PLC144

SUBJECT: INTRODUCTION TO C++ PROGRAMMING LAB

PREPARED BY: DR. ELAVAAR KUZHALI S, EIE, RIT
DR. CHANDRIKA, CSE, RIT

APPROVED BY: DR. ANNAPOORNA PATIL

REVISION: 001

Dec 2022- March 2023

M S Ramaiah Nagar, MSRIT Post, Bangalore 560 054
T+ 080 2360 3122 / 0822

Introduction to C++Programming- PLC144

Lab Manual

Term: Dec 2022- March 2023

USN:

Name:

CERTIFICATE

This is to certify that Mr./Ms. _____ has satisfactorily completed the course of experiment's in practical of _____ prescribed by the Department of _____ , Ramaiah Institute of Technology for year 20___ ___ - 20 ___ ___

Component	Maximum Marks	Marks Obtained
Record		
Lab Test		
Viva Voce		
CIE Marks		

Signature of the Student with Date

Name and Signature of the Faculty with Date

Signature of Head of the Department

RAMAIAH INSTITUTE OF TECHNOLOGY

(Autonomous Institute, Affiliated to VTU)

MSR Nagar, Bangalore – 560 054

Content Sheet

INDEX

Sl. No	List of Experiments	Date of Experiment	Marks Obtained	Signature of the Faculty
1.				
2.				
3.				
4.				
5.				
6.				
7.				
8.				
9.				
10.				
11.				

Lab Question 1

1 a) Write a C++ program to find the area and circumference of a circle

```
#include <iostream>
using namespace std;
#define PI 3.141
 int main()
{
   float radius, area;
   cout << "Enter radius of circle\n";
   cin >> radius;
   // Area of Circle = PI x Radius X Radius
   area = PI*radius*radius;
   cout << "Area of circle : " << area;
   return 0;
}
```

OUTPUT

Lab Question 1

b) Write a C++ program to find the simple interest

```
#include<iostream>

using namespace std;
int main()
{
  float p, r, t, si;
  cout<<"Enter Principle Amount: ";
  cin>>p;
  cout<<"Enter Rate of Interest: ";
  cin>>r;
  cout<<"Enter Time Period: ";
  cin>>t;
  si = (p*r*t)/100;
  cout<<"\nSimple Interest Amount: "<<si;
  cout<<endl;
  return 0;
}
```

OUTPUT

Lab Question 1

c) Write a C++ program to find the area of a triangle given its sides

```
// C++ program to find the area of a triangle
#include<iostream>
#include<math.h>
using namespace std;
int main()

{
float s1,s2,s3;
cout<<" enter the three sides of a triangle\n";
cin>>s1>>s2>>s3;

float s = (s1 + s2 + s3)/2;
float area = sqrt(s*(s - s1)*(s - s2)*(s - s3));

cout <<"Area of a triangle given its sides are= " << endl<<area;

}
```

OUTPUT:

Lab Question 1

d) Write a C++ program to get the name, age and salary of a person and display the same

```
// C++ program to display person's information
#include <iostream>
using namespace std;

int main()
{
        char name[10];
        int age;
       float salary;

        cout<<"Enter name of the person:\n ";
        cin>>name;
        cout<<"Enter age: \n";
        cin>>age;
        cout<< "Enter salary\n";
       cin>>salary;
        cout<<"Name: "<<name<<endl;
        cout<<"Age: "<<age<<endl;
       cout<<"Salary:"<< salary<<endl;

        return 0;
}
```

OUTPUT:

Lab Question 2

2 a) Write a C++ program to find the factorial of a number

```
// C++ program to display factorial of a number
#include <iostream>
using namespace std;

int main()

   int n;
   long factorial = 1.0;

   cout << "Enter a positive integer: ";
   cin >> n;

   if (n < 0)
      cout << "Error! Factorial of a negative number doesn't exist.";


 else
{
      for(int i = 1; i <= n; ++i)

 {
         factorial *= i;
      }

cout << "Factorial of " << n << " = " << factorial;


}
```

OUTPUT:

Lab Question 2

b) Write a C++ program to find whether the entered number is palindrome or not.

```
// Program to test a number, palindrome or not

#include <iostream>
using namespace std;

int main()
{
    int n, num, digit, rev = 0;

    cout << "Enter a positive number: ";
    cin >> num;

    n = num;

    do
    {
       digit = num % 10;
       rev = (rev * 10) + digit;
       num = num / 10;
    } while (num != 0);

    cout << " The reverse of the number is: " << rev << endl;

    if (n == rev)
       cout << " The number is a palindrome.";
    else
       cout << " The number is not a palindrome.";

   return 0;
}
```

OUTPUT:

Lab Question 2

c) Write a C++ program to find the sum of all the natural numbers from 1 to n.

```
// Program to find sum from 1 to N

#include<iostream>
using namespace std;

int main()
{
   int n;
   cout << "Enter a number : ";
   cin >> n;

   int sum=0;

   for(int i=1;i<=n;i++)
      sum+=i;

   cout << sum;

   return 0;
}
```

OUTPUT:

Lab Question 2

d) Write a C++ program to find sum of all the elements, maximum and minimum element in an array

```
// Program on array
#include <iostream>
using namespace std;
int main()
{
   int array[10];
   int i, max, min, size, sum;

   cout << "Enter size of the array: ";
   cin >> size;

   // Input array elements
   cout << "\n Enter the marks of " << size << " students: ";
   for (i = 0; i < size; i++)
      cin >> array[i];

   max = array[0];
   min = array[0];

   for (i = 1; i < size; i++)
   {
      // Checking for max
      if (array[i] > max)
         max = array[i];

      // Checking for min
      if (array[i] < min)
         min = array[i];

    sum=sum+array[i];
   }

   // Print maximum and minimum element
   cout << "\n Maximum marks =" << max;
   cout << "\n Minimum marks =" << min;
   cout<< "\n Sum of all the numbers="<< sum;

   return 0;
}
```

OUTPUT:

Lab Question 3

3 a)Write a C++ program to find whether an entered number is prime or not using a function(with value, with return type)

```
#include <iostream>

using namespace std;
bool CheckPrime(unsigned long n);
int main()
{
   cout << "Happy Programming!" << endl<<endl;
   unsigned long n;
   cout << "Enter the Number to check Prime:"<<endl;
   cin >> n;
   bool P = CheckPrime(n);
   if(P == 1)
      cout << " Number entered is a prime number\n";

   return 0;
}
bool CheckPrime(unsigned long n)
{
   unsigned long i;int flag = 1;
   for(i = 2; i <= n/2; i++)
   {
      if(n % i == 0)
       {
          flag = 0;
          cout<<"Number is not a Prime & divisible by "<<i<<endl;
          return 0;
       }

   }
   if(flag == 1)
      return 1;
}
```

OUTPUT:

Lab Question 3

b) Write a C++ program to search an element in an array using linear search

```
#include <iostream>
#define MAX 5
using namespace std;
void LinearSearch(int[],int);
int main()
{
   cout << "Happy Programming!" << endl << endl;
   int arr[MAX],Key;
   cout << "Enter the elements of the array:"<<endl;
   for(int i=0;i<MAX;i++)
   {
      cin>>arr[i];
   }
   cout << "Enter the element to find:"<<endl;
   cin>>Key;
   LinearSearch(arr,Key);
   return 0;
}
void LinearSearch(int parr[],int k)
{
   int flag = 0;
   for(int i=0;i<MAX;i++)
   {
      if(parr[i]==k)
      {
         flag = 1;
         cout<<"Number is found at location "<<i+1<<endl;
         break;
      }

   }
   if(flag == 0)
    cout<<"Number not found in array "<<endl;
}
```

OUTPUT:

Lab Question 3

c) Write a C++ program to swap 2 values by writing a function that uses call by reference technique.

```
#include <iostream>

using namespace std;
void Swap(int&, int&);
int main()
{
   cout << "Happy Programming! Demonstrating Reference Variables" << endl;
   int Var1,Var2;
   cout<<"Enter two numbers to be swapped:"<<endl;
   cin>>Var1>>Var2;
   cout<<"Before swapping:"<<endl;
   cout<<" Var1 is: \t"<<Var1<<endl;
   cout<<" Var2 is: \t"<<Var2<<endl;
   Swap(Var1,Var2);
   cout<<"After swapping:"<<endl;
   cout<<" Var1 is: \t"<<Var1<<endl;
   cout<<" Var2 is: \t"<<Var2<<endl;

   return 0;
}
void Swap(int &x, int &y)
{
   int temp=x;
   x=y;
   y=temp;
}
```

OUTPUT:

Lab Question 4

4 a) Write a C++ program to perform square of a number using inline function

Method1:

```
using namespace std;
inline void square(int numb);
int main()
{
   cout << "Hello world!" << endl;
   int numb ;
   cout<<" Enter a numb to find square"<<endl;
   cin>>numb;
   cout<<"\nSquare of "<<numb<<"  is \t ";
   square(numb);
   return 0;
}
void square(int numb)
{
cout<<numb*numb<<endl;
}
```

Method 2:

```
#include <iostream>

using namespace std;
class Square
{
   int numb;
   int res;

public:
   inline void get_numb()
   {
      cout<<" Enter a numb to find square"<<endl;
      cin>>numb;
   }
   inline void disp_square()
   {
       res = numb *numb;
       cout<<"\nSquare of "<<numb<<"  is \t " <<res<<endl;
   }
};
```

```
int main()
{
   cout << "Hello world!" << endl;
   Square S;
   S.get_numb();
   S.disp_square();
   return 0;
}
```

Method 3:

```
#include <iostream>
using namespace std;
class Square
{
   int numb;
   int res;

public:
   void get_numb();
   void disp_square();
};

int main()
{
   cout << "Hello world!" << endl;
   Square S;
   S.get_numb();
   S.disp_square();
   return 0;
}
inline void Square::get_numb()
{
   cout<<" Enter a numb to find square"<<endl;
   cin>>numb;
}
inline void Square::disp_square()
{
   res = numb *numb;
   cout<<"\nSquare of "<<numb<<"  is \t " <<res<<endl;
}
```

OUTPUT:

Lab Question 4

b) Write a C++ program to create a class called bank_acct with following data member (cust_name, cust_accno, balance) and member functions (read_details, deposit, withdraw, display balance). Read and display details using array of objects and implement deposit and withdraw using inline.

```
#include <iostream>
#include <iomanip>
#include <string.h>

using namespace std;
class bank_acct
{
   char cust_name[30];
   char cust_accno[30];
   float balance;

public:
   bank_acct()
   {
      cust_name[0] ='\0';
      cust_accno[0] ='\0';
      balance = 500;
   }
   void read_details();
   void deposit();
   void withdraw();
   void display_balance();
   void display_details();
   void searchAcc(char[],char);
};

inline void bank_acct :: read_details()
{
   cout<<"Enter the details of the customer"<<endl;
   cout<<"\n Enter the Customer Name: \t";cin>>cust_name;
   cout<<"\n Enter the Customer Account Number: \t";cin>>cust_accno;
}
inline void bank_acct :: deposit()
{
   int depamt;
```

```
   cout<<"\n Enter Deposit Amount = ";
   cin>>depamt;
   balance+=depamt;
}
inline void bank_acct :: withdraw()
{
   int wdamt;
   cout<<"\n Enter Withdraw Amount = ";
   cin>>wdamt;

   if(wdamt>balance)
      cout<<"\n Cannot Withdraw Amount";

   balance-=wdamt;
}
inline void bank_acct :: searchAcc(char AccNum[],char op)
{
   if(!(strcmp(cust_accno,AccNum)))
   {
      if(op == 'd')
         deposit();
      else
         withdraw();

       display_balance();
   }
}
inline void bank_acct :: display_balance()
{
   cout<<" The balance amount of "<< cust_name<<endl;
   display_details();
}
inline void bank_acct :: display_details()
{
   cout<<setw(30)<<"Customer Name: "<<setw(30)<<cust_name<<endl;
   cout<<setw(30)<<"Account Number: "<<setw(30)<<cust_accno<<endl;
   cout<<setw(30)<<"Balance: "<<setw(30)<<balance<<endl;
}
int main()
{
   cout << "Happy Programming!" << endl;
   char ch = 'y',c='s';
   int i=0;
```

```
   char AccNo[30];

   bank_acct B[10];

// Code to Create Account for Customers
   do
   {
      B[i].read_details();
      i++;
      cout<<"Do you want to create account for another customer type 'y' or 'n'"<<endl;
      cin>>ch;

   }while(ch!='n');

// Code For Transaction

    cout<<"Enter the account number for deposit/ withdraw "<<endl;
    cout<<"Account Number: \t";cin>>AccNo;
    cout<< "Enter Transaction deposit/ withdraw ? type (d/w) \t ";cin>>c;
    for(int j=0;j<i;j++)
    {
       B[j].searchAcc(AccNo,c);
    }

   return 0;
}
```

OUTPUT:

Lab Question 5

5 a) Write and execute a C++ Program to display names, roll no's, and grades of 3 students who have appeared in the examination. Create a class with data members as Name, Roll no and Marks for 3 subjects. Write a method to calculate the grade of a student.

```
#include <iostream>
#include <iomanip>

using namespace std;

class STUDENT
{
   char Name[30];
   char RollNo[30];
   int Sub1Mark;
   int Sub2Mark;
   int Sub3Mark;
   char grade;

public:
   void getdetail();
   void calculategrade();
   void displaydetail();

};

void STUDENT :: getdetail()
{
   cout<<"*****Enter the details of the student****"<<endl;
   cout<<"\n Enter the Student Name: \t";cin>>Name;
   cout<<"\n Enter the Student Roll Number: \t";cin>>RollNo;
   cout<<"\n Enter the Student Subject_1 Mark: \t";cin>>Sub1Mark;
   cout<<"\n Enter the Student Subject_2 Mark: \t";cin>>Sub2Mark;
   cout<<"\n Enter the Student Subject_3 Mark: \t";cin>>Sub3Mark;

}
void STUDENT :: calculategrade()
{
   float avg = (Sub1Mark+Sub2Mark+Sub3Mark)/3;
   if(avg>=91 && avg <= 100)
      grade = 'S';
```

```
   else if(avg>=81 && avg <= 90)
      grade = 'A';
   else if(avg>=71 && avg <= 80)
      grade = 'B';
   else if(avg>=61 && avg <= 70)
      grade = 'C';
   else if(avg>=51 && avg <= 60)
      grade = 'D';
   else if(avg>=40 && avg <= 50)
      grade = 'E';
   else
      grade = 'F';

}
void STUDENT ::displaydetail()
{
   calculategrade();
   for(int i=0;i<10;i++)
      cout<<"*\t";
   cout<<endl;

   cout<<setw(30)<<"Student Name: "<<setw(30)<<Name<<endl;
   cout<<setw(30)<<"Student Roll Number: "<<setw(30)<<RollNo<<endl;
   cout<<setw(30)<<"Student Subject_1 Mark: "<<setw(30)<<Sub1Mark<<endl;
   cout<<setw(30)<<"Student Subject_2 Mark: "<<setw(30)<<Sub2Mark<<endl;
   cout<<setw(30)<<"Student Subject_3 Mark: "<<setw(30)<<Sub3Mark<<endl;
   cout<<setw(30)<<"Student Grade: "<<setw(30)<<grade<<endl;

   for(int i=0;i<10;i++)
      cout<<"*\t";
   cout<<endl;
}
int main()
{
   cout << "Happy Programming!" << endl;

   STUDENT S[3];
   char ch = 'y'; int i=0;

   do
   {
      S[i].getdetail();
      i++;
```

```
      cout<<"Do you want to details of another Student type 'y' or 'n'"<<endl;
      cin>>ch;
      cout<<endl;
   }while(ch!='n' && i<3);

   for(int j=0;j<i;j++)
   {
      S[j].displaydetail();
   }
   return 0;
}
```

OUTPUT:

Lab Question 5

b) Create a C++ class that includes constructors to do the following.

- Create an uninitialized string.
- Initialize an object with a string constant at the time of creation.
  Create an object and initialize with another object. Also write a function to concatenate two strings.

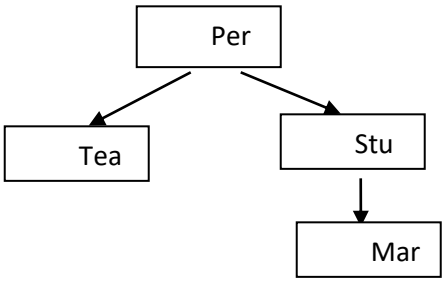
```
#include <iostream>
#include <string.h>
#include <iomanip>
#define MAX 80
using namespace std;
class STRING
{
   char str[MAX];
public:
   STRING()
   {
      str[0]='\0';
   }
   STRING(const char pstr[])
   {
      strcpy(str,pstr);
   }
   STRING strconcat(STRING X)
   {
      STRING temp;
      if(strlen(str)+strlen(X.str)<MAX)
      {
         strcpy(temp.str,str);
         strcat(temp.str,X.str);
      }
      return temp;
   }
   void display()
   {
      cout<<setw(30)<<"The string is:\t"<<str<<endl;
   }
};

int main()
{
```

```
    cout << "Happy Programming!" << endl;
    STRING S1,S2("Hello"),S3("MSRIT");
    cout<<"The uninitialized string for object S1 using constructor with no arg is:"<<endl;
    S1.display();
    cout<<"The initialized string for object S2 using parametrized constructor is:"<<endl;
    S2.display();
    cout<<"The initialized string for object S3 using parametrized constructor is:"<<endl;
    S3.display();
    cout<<endl<<"Concatenating object S2 with object S3"<<endl;
    S1 = S2.strconcat(S3);
    S1.display();
    return 0;
}
```

OUTPUT:

Lab Question 6

6 a) Write a C++ program to implement the following inheritance.



- Assume suitable data members and member functions for all the classes.
- Display the number of publications for a teacher and percentage marks for a student.

```
#include <iostream>
#include <iomanip>
#define MAX 80
using namespace std;

class Person
{
private:
   char Name[MAX];
   char AadharNum[MAX];
   char DOB[MAX];
public:
   void GetPersonInfo()
   {
      cout<<"\n Enter the Name: \t";cin>>Name;
      cout<<"\n Enter the Aadhar Number: \t";cin>>AadharNum;
      cout<<"\n Enter the DOB in DD-MM-YYYY: \t";cin>>DOB;
   }
   void DisplayPersonInfo()
   {
      cout<<setw(30)<<" Name: "<<setw(30)<<Name<<endl;
      cout<<setw(30)<<" Aadhar Number: "<<setw(30)<<AadharNum<<endl;
      cout<<setw(30)<<" DOB: "<<setw(30)<<DOB<<endl;

   }
```

```
};
class Teacher : public Person
{
   char Department[MAX];
   char Designation[MAX];
   int Publication;
public:
   void GetTeacherinfo();
   void DisplayTeacherInfo();
};
class Student : public Person
{
   char Department[MAX];
   char RollNo[MAX];
public:
   void GetStudentInfo();
   void DisplayStudentInfo();

};
class Marks : public Student
{
   int Sub1Mark;
   int Sub2Mark;
   int Sub3Mark;
public:
   void GetStudentMarks();
   void DisplayStudentMarks();
};
/* Member Function Definitions for class Marks*/
void Marks :: GetStudentMarks()
{
   cout<<"\n Enter the Student Subject_1 Mark: \t";cin>>Sub1Mark;
   cout<<"\n Enter the Student Subject_2 Mark: \t";cin>>Sub2Mark;
   cout<<"\n Enter the Student Subject_3 Mark: \t";cin>>Sub3Mark;
}
void Marks :: DisplayStudentMarks()
{
   cout<<setw(30)<<"Student Subject_1 Mark: "<<setw(30)<<Sub1Mark<<endl;
   cout<<setw(30)<<"Student Subject_2 Mark: "<<setw(30)<<Sub2Mark<<endl;
   cout<<setw(30)<<"Student Subject_3 Mark: "<<setw(30)<<Sub3Mark<<endl;
   cout<<setw(30)<<"Percentage:
"<<setw(30)<<((Sub1Mark+Sub2Mark+Sub3Mark)/3)<<"%"<<endl;
}
```

```
/* Member Function Definitions for class Student*/
void Student :: GetStudentInfo()
{
   GetPersonInfo();
   cout<<"Enter the Department: \t";cin>>Department;
   cout<<"\n Enter the Roll Number: \t";cin>>RollNo;
}
void Student ::DisplayStudentInfo()
{
   DisplayPersonInfo();
   cout<<setw(30)<<" Department: "<<setw(30)<<Department<<endl;
   cout<<setw(30)<<" Roll Number: "<<setw(30)<<RollNo<<endl;

}
/* Member Function Definitions for class Teacher*/
void Teacher :: GetTeacherinfo()
{
   Person::GetPersonInfo();
   cout<<"Enter the Department: \t";cin>>Department;
   cout<<"\n Enter the Designation: \t";cin>>Designation;
   cout<<"\n Enter the Number of Publication: \t";cin>>Publication;
}
void Teacher :: DisplayTeacherInfo()
{
   DisplayPersonInfo();
   cout<<setw(30)<<" Department: "<<setw(30)<<Department<<endl;
   cout<<setw(30)<<" Designation: "<<setw(30)<<Designation<<endl;
   cout<<setw(30)<<" Number of Publication: "<<setw(30)<<Publication<<endl;
}
/* Main Function */
int main()
{
   cout << "Happy Programming!" << endl<<endl;

   Teacher T[3];
   char ch = 'y'; int i=0;

   cout<<"*********Enter Teacher Details **********"<<endl<<endl;
   do
   {
      T[i].GetTeacherinfo();
      i++;
      cout<<"Do you want to details of another Teacher type 'y' or 'n'"<<endl;
```

```
            cin>>ch;
            cout<<endl;
          }while(ch!='n' && i<3);

        cout<<"*********Display Teacher Details **********"<<endl<<endl;
          for(int j=0;j<i;j++)
          {
            T[j].DisplayTeacherInfo();
          }

         cout<<endl<<"*********Enter Students Details **********"<<endl<<endl;

          Marks S[3];
          ch = 'y'; i=0;

          do
          {
            S[i].GetStudentInfo();
            S[i].GetStudentMarks();
            i++;
            cout<<"Do you want to details of another Student type 'y' or 'n'"<<endl;
            cin>>ch;
            cout<<endl;
          }while(ch!='n' && i<3);

         cout<<"*********Display Student Details **********"<<endl<<endl;
          for(int j=0;j<i;j++)
          {
            S[j].DisplayStudentInfo();
            S[j].DisplayStudentMarks();
          }

          return 0;
}
```

OUTPUT:

Lab Question 7

7 a) Write a C++ program to demonstrate multilevel inheritance for the following:

Suppose we have three classes Vehicle, FourWheeler, and Car. The class Vehicle is the base class, the class FourWheeler is derived from it and the class Car is derived from the class FourWheeler. Class Vehicle has a method 'vehicle' that prints 'I am a vehicle', class FourWheeler has a method 'fourWheeler' that prints 'I have four wheels', and class Car has a method 'car' that prints 'I am a car'. So, as this is a multi-level inheritance; we can have access to all the other classes methods from the object of the class Car. We invoke all the methods from a Car object and print the corresponding outputs of the methods.

So, if we invoke the methods in this order, car(), fourWheeler(), and vehicle(), then the output will be

```
I am a car
I have four wheels
I am a vehicle
```

```
#include <iostream>

using namespace std;
class Vehicle
{
public:
   void vehicle()
   {
      cout<<"I am a Vehicle"<<endl;
   }
};
class FourWheeler : public Vehicle
{
public:
   void fourwheeler()
   {
      cout<<"I have four Wheels"<<endl;
   }
};
class Car : public FourWheeler
{
public:
   void car()
   {
      cout<<"I am a Car"<<endl;
   }
```

```
};
int main()
{
   cout << "Happy Programming!" << endl;
   Car Cobj;

   Cobj.car();
   Cobj.fourwheeler();
   Cobj.vehicle();
   return 0;
}
```

OUTPUT:

Lab Question 8

8 a) Write a C++ program to overload function for computing the area triangle, circle and square

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

 void area(float,float);
 void area(float,double);
 void area(float);
 int main()
 {
    cout << "Happy Programming! Demonstrating Function Overloading" << endl<<endl;
    float height,base;
    cout<<"Calculate Area of Triangle for the following measurements"<<endl;
    cout<<"\n Enter the Height of Triangle in cms: \t";cin>>height;
    cout<<"\n Enter the Base of Triangle in cms: \t";cin>>base;
    area(height,base);
    float radius;
    cout<<"Calculate Area of Circle for the following measurements"<<endl;
    cout<<"\n Enter the radius of Circle in cms: \t";cin>>radius;
    area(radius,3.14);
    float side;
    cout<<"Calculate Area of Square for the following measurements"<<endl;
    cout<<"\n Enter the side of Square in cms: \t";cin>>side;
    area(side);
    return 0;
 }
 void area(float h,float b)
 {
    cout<<setw(30)<<"Area of Triangle:"<<setw(30)<<(0.5*h*b)<<endl;
 }
 void area(float r,double C)
 {
    cout<<setw(30)<<"Area of Circle:"<<setw(30)<<(C*r*r)<<endl;
 }
 void area(float s)
 {
    cout<<setw(30)<<"Area of Square:"<<setw(30)<<(s*s)<<endl;
 }
```

Lab Question 8

b) Write a C++ program to overload a function to add two numbers of different data types (int, float, double)

```
#include <iostream>
#include<iomanip>

using namespace std;

void add(int,int);
void add(float,float);

int main()
{
cout << "Happy Programming! Demonstrating Function Overloading" << endl<<endl;
int num1,num2;
cout<<"Addition of 2 numbers by calling function add(int,int);"<<endl;
cout<<"\n Enter the First Number: \t";cin>>num1;
cout<<"\n Enter the Second Number: \t";cin>>num2;
add(num1,num2);
float fnum1,fnum2;
cout<<"Addition of 2 numbers by calling function void add(float,float);"<<endl;
cout<<"\n Enter the First Number: \t";cin>>fnum1;
cout<<"\n Enter the Second Number: \t";cin>>fnum2;
add(fnum1,fnum2);
return 0;
}
void add(int n1,int n2)
{
cout<<setw(30)<<" Result of Addition of int numbers"<<setw(30)<<(n1+n2) <<endl;
}
void add(float fn1,float fn2)
{
cout<<setw(30)<<" Result of Addition of float numbers"<<setw(30)<<(fn1+fn2)
<<endl;
}
```

OUTPUT:

Lab Question 9

9 a) Write a C++ program to create a text file, check file created or not, if created write some text into the file and then read the text from the file.

```
// C++ program to demonstrate file open, read, write and close operations
#include <iostream>
#include <fstream>

using namespace std;

int main(){

  char text[200];

  fstream file;
  file.open ("example.txt", ios::out | ios::in );
   if(!file)
   {
      cout<<"Error in creating file!!!"<<endl;
      return 0;
   }

   cout<<"File created successfully."<<endl;

  cout << "Write text to be written on file." << endl;
  cin.getline(text, sizeof(text));

  // Writing on file
  file << text << endl;

  // Reding from file
  file >> text;
  cout << text << endl;

  //closing the file
  file.close();
  return 0;
}
```

OUTPUT:

Lab Question 9

b) Write a C++ program to read the contents from a text file, count and display the number of alphabets present in it.

```
// C++ program to count number of alphabets in a text file
#include<iostream>
#include<fstream>
using namespace std;
int main()
{
	ifstream  fin("read.txt");
	char ch;
	int i, c=0, sp=0;
	while(fin)
	{
		fin.get(ch);
		i=ch;
		if((i > 63 && i < 91) || (i > 96 && i < 123))
			c++;
		else
			if(ch== ' ')
				sp++;
	}
	cout<<"\n No. of Characters in a File : "<<c;
	cout<<"\n Space between the Words     : "<<sp;
	return 0;
}
```

OUTPUT:

Lab Question 10

10 a) Write a C++ program that creates a Calculator class. The class contains two variables of integer type. Design a constructor that accepts two values as parameter and set those values.

- Design four methods named Add (), Subtract (), multiply (), Division ( ) for performing addition, subtraction, multiplication and division of two numbers.
- For addition and subtraction, two numbers should be positive. If any negative number is entered then throw an exception in respective methods. So design an exception handler (ArithmeticException) in Add () and Subtract () methods respectively to check whether any number is negative or not.
- For division and multiplication two numbers should not be zero. If zero is entered for any number then throw an exception in respective methods. So design an exception handler (ArithmeticException) in multiply () and Division () methods respectively to check whether any number is zero or not.

```
// Program to demonstrate exception handling
#include <iostream>
using namespace std;

int add(int a, int b) {
if(( a<0 ) || (b<0))  {
throw "Enter positive number";
}
return (a+b);
}

int sub(int a, int b) {
if(( a<0 ) || (b<0))  {
throw " Enter positive number ";
}

return (a-b);
}


int division(int a, int b) {
if( (b == 0 ) || (a==0) ) {
throw "Enter a number greater than zero”;
}
return (a/b);
}
```

```
int product(int a, int b) {
if(( a == 0 ) && (b==0))  {
throw " Enter a number greater than zero!";
}
return (a*b);
}

int main () {
int x = 50;
int y = 0;
int  z = 0;

try {
z = division(x, y);
cout << z << endl;
} catch (const char* msg) {
cerr << msg << endl;
}
try {
z = product(x, y);
cout << z << endl;
} catch (const char* msg) {
cerr << msg << endl;
}
try {
z = add(x, y);
cout << z << endl;
} catch (const char* msg) {
cerr << msg << endl;
}
try {
z = sub(x, y);
cout << z << endl;
} catch (const char* msg) {
cerr << msg << endl;
}

return 0;
}
```