Course: C++ Programming

LAB MANUAL

(Course Code: 22PLC25D)

Semester: II

Name of the Student :

Semester/ Section :

USN and Roll No :

Batch :

DAYANANDA SAGAR COLLEGE OF ENGINEERING
Accredited by National Assessment & Accreditation Council (NAAC) with 'A' Grade
(An Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi
ISO 9001:2008, ISO 14001:2004 and ISO 22000:2005 Certified)
SHAVIGE MALLESWARA HILLS, KUMARASWAMY LAYOUT
BENGALURU-560078

DAYANANDA SAGAR COLLEGE OF ENGINEERING

Accredited by National Assessment & Accreditation Council (NAAC) with 'A' Grade

(An Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi
ISO 9001:2008, ISO 14001:2004 and ISO 22000:2005 Certified)

SHAVIGE MALLESWARA HILLS, KUMARASWAMY LAYOUT

BENGALURU-560078

Vision of the Institution

To impart quality technical education with a focus on Research and Innovation emphasizing on Development of Sustainable and Inclusive Technology for the benefit of society.

Mission of the Institution

- To provide an environment that enhances creativity and Innovation in pursuit of Excellence.
- To nurture teamwork in order to transform individuals as responsible leaders and entrepreneurs.
- To train the students to the changing technical scenario and make them to understand the importance of Sustainable and Inclusive technologies.

DAYANANDA SAGAR COLLEGE OF ENGINEERING

Accredited by National Assessment & Accreditation Council (NAAC) with 'A' Grade

(An Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi
ISO 9001:2008, ISO 14001:2004 and ISO 22000:2005 Certified)

SHAVIGE MALLESWARA HILLS, KUMARASWAMY LAYOUT

BENGALURU-560078

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Vision of the Department

To provide a vibrant learning environment in computer science and engineering with focus on industry needs and research, for the students to be successful global professionals contributing to the society.

Mission of the Department

- To adopt a contemporaryteaching learning process with emphasis on hands on and collaborative learning
- To facilitate skill development through additional training and encourage student forums for enhanced learning.
- To collaborate with industry partners and professional societies and make the students industry ready.
- To encourage innovation through multidisciplinary research and development activities
- To inculcate human values and ethics to groom the students to be responsible citizens.

DAYANANDA SAGAR COLLEGE OF ENGINEERING

Accredited by National Assessment & Accreditation Council (NAAC) with 'A' Grade

(An Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi
ISO 9001:2008, ISO 14001:2004 and ISO 22000:2005 Certified)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Code of Conduct in the Lab

Do's

Students shall

- Come prepared for the program to be developed in the laboratory.
- Report any broken plugs or exposed electrical wires to your faculty/laboratory technician immediately.
- Turn off the machine once you have finished using it.
- Maintain silence while working in the lab.
- Keep the Computer lab premises clean and tidy.
- Place backpacks under the table or computer counters.

Don'ts

Students shall not

- Talk on cell phones in the lab.
- Eat or drink in the laboratory.
- Install or download any software or modify or delete any system files on any lab computers.
- Leave their personal belongings unattended. We are not responsible for any theft.

Course Objectives:

1. Understanding about the fundamental programming concepts and methodologies whichare essential for building good C/C++ programs.
2. Understand how to design C++ classes for code reuse.
3. Understand how to implement constructors, encapsulation, overloading function, inheritance file handling and exception handling.
4. Understand how to use exception handling in C++ programs.

Course Outcomes: At the end of the course, student will be able to:

CO1	Program solutions for real life problems using OO principles in C++.
CO2	Create UML class diagrams for given application scenarios.
CO3	Debug C++ programs by tracing through specific outputs.
CO4	Incorporate exception handling in C++ programs.
CO5	Implement Input/output operations using streams.

Mapping of Course outcomes to Program outcomes:

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	1	1	1	-	-	1	-	-	1	1	-	1
CO2	1	1	-	-	-	-	-	-	1	2	-	1
CO3	1	1	1	-	-	-	-	-	1	1	-	1
CO4	1	1	1	-	-	-	-	-	1	1	-	1
CO5	1	1	1	-	-	-	-	-	-	1	-	1

LABORATORY WORK			
Program No	**Course Content**	**Hours**	**COs**
1.	Write a C++ program to read two numbers from the keyboard and display the larger value on the screen.	**02**	**CO1 CO2**
2.	Write a C++ program to find the sum of all the natural numbers from 1 to n.	**02**	**CO1 CO2**
3.	Write a C++ program to swap 2 values by writing a function that uses call by reference technique.	**02**	**CO1 CO2**
4.	Write a C++ program to demonstrate function overloading for the following prototypes. add(int a, int b) add(double a, double b).	**02**	**CO1 CO2**
5.	Create a class named Shape with a function that prints "This is a shape". Create another class named Polygon inheriting the Shape class with the same function that prints "Polygon is a shape". Create two other classes named Rectangle and Triangle having the same function which prints "Rectangle is a polygon" and "Triangle is a polygon" respectively. Again, make another class named Square having the same function which prints "Square is a rectangle". Now, try calling the function by the object of each of these classes.	**02**	**CO1 CO2**
6.	Suppose we have three classes Vehicle, Four-Wheeler, and Car. The class Vehicle is the base class, the class Four-Wheeler is derived from it and the class Car is derived from the class Four-Wheeler. Class Vehicle has a method 'vehicle' that prints 'I am a vehicle', class Four- Wheeler has a method 'four-Wheeler' 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 themethods in this order, car(), four Wheeler(), and vehicle(), then the output will be I am a car I have four wheels I am a vehicle Write a C++ program to demonstrate multilevel inheritance using this.	**02**	**CO1 CO2**
7.	Write a C++ program to create a text file, check file created or not, if created it will write some text into the file and then read the text from the file.	**02**	**CO1 CO2 CO3**

8.	Write a C++ program to write and read time in/from binary file using fstream.	**02**	**CO1 CO2 CO3**
9.	Write a function which throws a division by zero exception and catch it in catch block. Write a C++ program to demonstrate usage of try, catch and throw to handle exception.	**02**	**CO1 CO2 CO3 CO4**
10.	Write a C++ program function which handles array of bounds exception using C++.	**02**	**CO2 CO3 CO5**

Program 1: Write a C++ program to read two numbers from the keyboard and display the larger value on the screen.

PROGRAM

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

int main()
{
  int num1, num2; cout<<"Enter
  first number:"; cin>>num1;
  cout<<"Enter second number:";
  cin>>num2;
  if(num1>num2)
  {
        cout<<"First number "<<num1<<" is the largest";
  }
  else
  {
        cout<<"Second number "<<num2<<" is the largest";
  }
  return 0;
}
```

Output:

Enter first number: 10 Enter

second number: 20

Second number 20 the largest

Sample Viva Questions:

1) What is an object? Explain it.
2) Differentiate between Object Oriented and Procedure Oriented languages.
3) Object is an instance of class. Justify your answer.
4) Differentiate between Compiler and Interpreter.
5) What does main function do?
6) For what iostream stands? Differentiate between iostream.h and stdio.h?

Program 2: Write a C++ program to find the sum of all the natural numbers from 1 to n.

Program:

```
#include<bits/stdc++.h>
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:

```
Enter a number : 10

Sum = 55
```

Sample Viva Questions:

1) Explain C++ Operator.
2) What are the types of binary operator?
3) Explain the difference between logical operators and bitwise operators.
4) Explain arithmetic operator.
5) What is the difference P++ and p+1.

Program 3: Write a C++ program to swap 2 values by writing a function that uses call by reference technique.

Program:

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

void swap (int *x, int *y)
{
        int swap;
        swap=*x;
        *x=*y;
        *y=swap;
}

int main( )
{
        int x=500, y=100;
        swap (&x, &y); // passing value to function

        cout<<"valus of x is: "<<x<<endl;
        cout<<"value of y is: "<<y<<endl;
        return 0;
}
```

Output

```
Value of x is: 100

Value of y is: 500
```

Sample Viva Questions:

1) What is a data type? Which are the data types supported in C++?
2) What is the application of scope resolution operator :: in C++?
3) Why is an array called a derived data type?
4) What are Objects? How are they created?
5) What is function overloading?

Program 4: Write a C++ program to demonstrate function overloading for the following prototypes. add(int a, int b)add(double a, double b).

Program:

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


void add(int a, int b)
{
  cout << "sum = " << (a + b);
}

void add(double a, double b)
{
    cout << "sum = " << (a + b);
}

int main()
{
    add(10, 2);
    add(5.3, 6.2);

    return 0;
}
```

OUTPUT:

SUM is : 12 SUM

is : 11.5

Sample Viva Questions:

1) What are function declaration and function definition?
2) Define the term function?
3) What are the advantages of function?
4) Explain the types of function?
5) What is the difference between user define function and library function?
6) What is function prototype?

Program 5. Create a class named Shape with a function that prints "This is a shape". Create another class named Polygon inheriting the Shape class with the same function that prints "Polygon is a shape". Create two other classes named Rectangle and Triangle having the same function which prints "Rectangle is a polygon" and "Triangle is a polygon" respectively. Again, make another class named Square having the same function which prints "Square is a rectangle". Now, try calling the function by the object of each of these classes.

Program:

```
#include <iostream>

using namespace std;

class Shape
{
    public:
    Shape(){}
    virtual void print()
    {
            cout<<"\nThis is a shape.";
    }
};

class Polygon: public Shape
{
    public:
    Polygon(){}
    void print()
    {
        cout<<"\nPolygon is a shape.";
    }
};

class Rectangle: public Polygon
{
    public:
    Rectangle(){}
    void print()
    {
        cout<<"\nRectangle is a Polygon.";
    }
};

class Triangle: public Polygon
{
    public:
    Triangle(){}
    void print()
```

```
    {
        cout<<"\nTriangle is a Polygon.";
    }
};

class Square: public Rectangle
{
    public:
    Square(){}
    void print()
    {
        cout<<"\nSquare is a Rectangle.";
    }
};

int main()
{
    Shape S;
    Polygon P;
    Rectangle R;
    Triangle T;
    Square Sq;
    S.print();
    P.print();
    R.print();
    T.print();
    Sq.print();
    return 0;
}
```

Output:

This is a shape. Polygon is a shape. Rectangle is a Polygon. Triangle is a Polygon. Square is a Rectangle.

Sample Viva Questions:

1) What is Inheritance?
2) What are the types of inheritance?
3) Explain Multiple and Multilevel Inheritance?
4) What are the visibility modes or access specifiers? Explain.
5) Explain Polymorphism.

Program 6: Suppose we have three classes Vehicle, Four Wheeler, and Car. The class Vehicle is the base class, the class Four Wheeler is derived from it and the class Car is derived from the class Four Wheeler. Class Vehicle has a method 'vehicle' that prints 'I am a vehicle', class Four Wheeler has a method 'four Wheeler' 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(), four Wheeler(), and vehicle(), then the output will be

I am a car

I have four wheels I

am a vehicle

Write a C++ program to demonstrate multilevel inheritance using this.

Program:

```
#include <iostream> using

namespace std;

class Vehicle
{
        public:
        void vehicle()
        {
                cout<<"I am a vehicle\n";
        }
};

class FourWheeler : public Vehicle
{
        public:
        void fourWheeler()
        {
                cout<<"I have four wheels\n";
        }
};

class Car : public FourWheeler
{
        public:
        void car()
```

```
        {
                cout<<"I am a car\n";
        }
};

Int main()
{
        Car obj; obj.car();
        obj.fourWheeler();
        obj.vehicle(); return
        0;

}
```

Output:

I am a car

I have four wheels I

am a vehicle

Sample Viva Questions:

1) Explain Virtual function.
2) What is Pure Virtual function?
3) What is the main purpose of using Inheritance? Explain its types.
4) What is abstract class?
5) What is the main reason of using pure virtual function?
6) What are the types of Polymorphism?

Program 7 : Write a C++ program to create a text file, check file created or not, if created it will write some text into the file and then read the text from the file.

Program:

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

int main()
{
   fstream file; //object of fstream class

   //opening file "sample.txt" in out(write) mode
   file.open("sample.txt",ios::out);

   if(!file)
   {
      cout<<"Error in creating file!!!"<<endl;
      return 0;
   }

   cout<<"File created successfully."<<endl;
   //write text into file file<<"ABCD.";
   //closing the file
   file.close();

   //again open file in read mode
   file.open("sample.txt",ios::in);

   if(!file)
   {
      cout<<"Error in opening file!!!"<<endl;
      return 0;
   }

   //read untill end of file is not found. char
   ch; //to read single character cout<<"File
   content: ";

   while(!file.eof())
   {
      file>>ch; //read single character from file
      cout<<ch;
   }
```

```
    file.close(); //close file

    return 0;
}
```

Output:

File created successfully.
File content: ABCD.

Sample Viva Questions:

1) Define is a file?
2) Explain the types of file.
3) What is the difference between istream and ostream?
4) What are the basic file operations?
5) Explain fstream with respect of input and output.

Program 8 : Write a C++ program to write and read time in/from binary file using fstream.

Program:

```
#include <iostream>
#include <fstream>
#include <iomanip> //for setfill() and setw()

using namespace std;

#define FILE_NAME "time.dat"

//function to write time into the file
void writeTime(int h, int m, int s){

	char str[10];

	fstream file;
	file.open(FILE_NAME, ios::out|ios::binary);

	if(!file){
		cout<<"Error in creating file!!!"<<endl;
		return;
	}

	//make string to write
	sprintf(str,"%02d:%02d:%02d",h,m,s);

	//write into file
	file.write(str,sizeof(str));
	cout<<"Time "<<str<<" has been written into file."<<endl;

	//close the file
	file.close();

}

//function to read time from the file
void readTime(int *h,int *m, int *s){

	char str[10];
	int inH,inM,inS;

	fstream finC;
	finC.open(FILE_NAME,ios::in|ios::binary);
	if(!finC){
```

```
                cout<<"Error in file opening..."<<endl;
                return;
        }
        if(finC.read((char*)str,sizeof(str))){
                //extract time values from the file sscanf(str,"%02d:%02d:%02d",&inH,&inM,&inS);
                //assign time into variables, which are passing in function
                *h=inH;
                *m=inM;
                *s=inS;
        }
        finC.close();
}

int main()
{
        int m,h,s;

        cout<<"Enter time:\n"; cout<<"Enter
        hour: ";                        cin>>h;
        cout<<"Enter minute: "; cin>>m;
        cout<<"Enter second: "; cin>>s;

        //write time into file
        writeTime(h,m,s);

        //now, reset the variables
        h=m=s=0;

        //read time from the file
        readTime(&h,&m,&s);

        //print the time cout<<"The
        time is
"<<setw(2)<<setfill('0')<<h<<":"<<setw(2)<<setfill('0')<<m<<":"<<setw(2)<<setfill('0')<<s<<endl;

        return 0;
}
```

Output:

```
Enter time:
Enter hour: 10
Enter minute: 15
Enter second: 5
Time 10:15:05 has been written into file. The
time is 10:15:05
```

Sample Viva Questions:

1) What is file handling?
2) What is a binary file?
3) What are the difference between ifstream and ofstream classes?
4) Expain the types of input and output function.
5) What are the advantages of saving data in binary form?
6) What are input and output stream?

Program 9: Write a function which throws a division by zero exception and catch it in catch block. Write a C++ program to demonstrate usage of try, catch and throw to handle exception.

Program:

```
#include <iostream>
#include <stdexcept> // To use runtime_error using
namespace std;

// Defining function Division
float Division(float num, float den)
{
    // If denominator is Zero
    // throw runtime_error if
    (den == 0) {
        throw runtime_error("Math error: Attempted to divide by Zero\n");
    }

    // Otherwise return the result of division return
    (num / den);

} // end Division int

main()
{
    float numerator, denominator, result;
    numerator = 12.5;
    denominator = 0;

    // try block calls the Division function try {
        result = Division(numerator, denominator);

        // this will not print in this example cout
        << "The quotient is "
            << result << endl;
    }

    // catch block catches exception thrown
    // by the Division function catch
    (runtime_error& e) {

        // prints that exception has occurred
        // calls the what function
        // using runtime_error object
        cout << "Exception occurred" << endl
```

```
            << e.what();
    }

} // end main
```

Output:

Exception occurred

Math error: Attempted to divide by Zero

Sample Viva Questions:

1) What is an Exception?
2) What is the difference between an error and an exception?
3) When does an exception occurs and why?
4) Define try block, catch block and throw statement.
5) How is an exception handled in C++?

Program 10: Write a C++ program function which handles array of bounds exception using C++.

Program:

```
#include <iostream> #include
<stdexcept>

using namespace std; int main()
{
        int a[l1l0],1i,slize; try{
                cout<<"Enter array size:"<<endl; cin>>size;
                if (size>10) throw 5;
                if (size<=0) throw
                3.2%;
                cout<<"Enter array elements"<<endl;

                for (i=0;i<size;i++) cin>>alil:;
                cout<<"Array elements are:"<< endl;

                for (i=0;i<size;i++)
                cout<<al[il<<" ";
          }

        catch (int x) {
        cout<<"Array size out of bounds exception occured"<<endl;

}
catch (float vy)
{
        cout<<"Neagative array size exception occured"<<endl;

}

return 0;

}
```

Output:

1) Enter array size:
10
Enter array elements
2
5

8
9
5
6
3
1
6
7
Array elements are:
2 5 8 9 5 6 3 1 6 7

2) Enter array size: 20
Array size out of bounds exception occurred.

Sample Viva Questions:

1) What is an array?
2) When do we use multiple catch?
3) What are the advantages of using exception handling mechanism in a program?
4) When should a program throw an exception?
5) What happens when an exception is thrown outside a try block?
6) What happens when an exception rethrown within a catch block?

DAYANANDA SAGAR COLLEGE OF ENGINEERING

An Autonomous Institute affiliated to Visvesvaraya Technological University (VTU),Approved by AICTE and UGC,
Accredited by NAAC with 'A' grade,
All UG courses Accredited by NBA, and ISO 9001-2015 Certified Institution

Under Graduation Courses (BE)

- Artificial Intelligence & Machine Learning
- Aeronautical Engineering
- Automobile Engineering
- Biotechnology
- Chemical Engineering
- Civil Engineering
- Computer Science and Engineering
- Computer Science and Business Systems
- Computer Science (Cyber Security)
- Computer Science (Data Science)
- Computer Science and Design
- Electrical & Electronics Engineering
- Electronics & Communication Engineering
- Information Science & Engineering
- Electronics & Instrumentation Engineering
- Mechanical Engineering
- Medical Electronics Engineering
- Electronics & Telecommunication Engineering

Post Graduation Courses (MBA, MCA, M.Tech)

- Structural Engineering
- Highway Technology
- Design Engineering
- Computer Integrated Manufacturing
- Digital Electronics and Communication
- VLSI Design and Embedded Systems
- Power Electronics
- Microelectronics and Control Systems
- Computer Network Engineering
- Digital Communication & Networking
- Computer Science & Engineering
- Master of Business Administration
- Master of Computer Applications

DSCE HAS **GRANTS** CLOSE TO **04 CRORES** FUNDED BY:

DSCE HAS MoU'S WITH:

SPECIAL FEATURES

- Students are trained in unique Industry labs of Volvo, Toyota, Bosch, Texas Instruments, Autodesk, Altair
- IEDC Center catering to entrepreneurial activities of the students.
- Student clubs, Professional Society Student chapters, Innovation Labs provide students opportunity to hone their skills.
- DSCE enjoys the distinction of showing its achievements in all major sporting events at VTU sports competition.

HIGHLIGHTS

? DSCE is an autonomous institution under VTU, Recognized by UGC, Approved by AICTE
Accredited by NAAC with 'A' grade, 14 UG Programmes accredited by NBA, ISO Certified & NIRF Ranked instituteDSCE has 4 DECADES of proven track record

?DSCE now enters the prestigious PLATINUM standard league in AICTE-CII National Survey 2021 of best Industry linked Institutes

? DSCE has many MoUs with leading Industry & Research centers

? Has got Maximum four star Ranking in ARIIA, Atal Ranking of Institutions on Innovation Achievements done by MHRD, Govt. of India.

? 100% Placements in reputed companies for all eligible students

?Centre for Innovation & Leadership in DSCE provides pre-placement training in soft skills, CV Writing, Aptitude Assessment, Mock interviews, Group Discussion etc.,

? Industrial conclaves, hands on training, workshops, industrial visits, internships, supported by Industry updated curriculum to meet industrydemands.

? WIFI campus with IT connectivity & e-class rooms

?Well qualified experienced faculty with rich Industry and R & D experience having global exposure

? Separate best in class hostel facility for boys and girls

FOR MORE INFORMATION PLEASE VISIT OUR CAMPUS
DAYANANDA SAGAR COLLEGE OF ENGINEERING, SHAVIGE MALLESHWARA HILLS,
KUMARASWAMY LAYOUT, BANASHANKARI, BENGALURU - 78
WEBSITE: https://www.dsce.edu.in/ **E-MAIL:** admissions@dayanandasagar.edu
PHONE: +91 80 42161751/ 26662226 **FAX:** 080-26660789