EXPERIMENT NUMBER –4.1

STUDENT'S NAME –

STUDENT'S UID –

CLASS AND GROUP –

SEMESTER –2nd

AIM OF THE EXPERIMENT:-LEARN HOW TOUSE CONSTRUCTOR AND DESTRUCTOR IN C++

FLOWCHART/ALGORITHM:-

PRACTICAL 4.1:-WAP to find area of rectangle using constructor overloading. Also define destructor to delete the memory allocated to objects.

STEP 1. START.
STEP 2. Declare the class area and the variable l and b.
STEP3. Now declare the constructor with no parameter (default) and constructor with two parameter.
STEP4.Also declare the destructor.
STEP5. Now in the main function declare the object of the class and with the help of object print the area and see the required result.
STEP 6. Stop.

PRACTICAL 4.2:WAP to create database of the following items: Name of the student (String), Roll number of the student (int), Height of the student (cm), Weight of the student (kg/gms)

1) Create a Constructor to initialize values

2) Create display () function to display the details

3) Illustrate the use of copy constructor

4) Also implement the concept of destructor.

STEP1. START.
STEP2. Declare the class student and declare the variables name, roll no, height and student.

STEP3. Now declare the constructor in the class and assign the values to the variable.
STEP4. Declare the show () function in the public block to print the details.
STEP5. Declare the copy constructor and destructor respectively.
STEP6. Now in the main function ask the user to enter the details and declare the object for the class and pass the values to the function parameter to print the desired result.
STEP7. STOP.

PROGRAM CODE:-

PROGRAM- 4.1:-

```
#include<iostream>
using namespace std;
class area
{
 int a,l,b;
 public:
 area()
 {
   l=5;
   b=6;
   cout<<"Simple constructor called\n";
   cout<<"length="<<l<<"\nbreadth="<<b<<endl;
 }
 area(int x,int y)
 {
   l=x;
   b=y;
 }
 void calc();
 void print();
 ~area();
};
void area::calc()
{
  a=l*b;
}
void area::print()
{
  cout<<"Area is : "<<a<<endl;
```

```
}
area::~area()
{
cout << "Object is being deleted" << endl;
}
int main()
{
  int l,b;
  area a1;
  a1.calc();
  a1.print();
  cout<<"Enter length and breadth for parameterised       constructor:\n";
  cin>>l>>b;
  area a2(l,b);
  a2.calc();
  a2.print();
  return 0;
}
```

PROGRAM- 4.2:

```
#include <iostream>
using namespace std;
class student
{
        private:
                string name;
                int  rollNo;
                int  height;
                int weight;
        public:
                student(string n,int r,int h,int w)
           {
              name=n;
              rollNo=r;
              height=h;
              weight=w;
            }
              void display(void);
                 ~student();
};
void student::display(void){
        cout << "Student details:\n";
        cout << "Name:"<< name << ",Roll Number:" << rollNo << ",Height:" << height << ",Weight:"
<< weight;
}
student :: ~student()
```

```
{
        cout<<"\ndestructor is called\n";
}
int main()
{
        student std("Vinay Prakash",9325,5,45);
        student std1=std;
        std1.display();
        return 0;
}
```

ERRORS ENCOUNTERED DURING PROGRAM'S EXECUTION:- NO ERROR

(Kindly jot down the compile time errors encountered)

PROGRAMS' EXPLANATION (in brief):-

PRACTICAL 4.1:-

In this program, first we have first declared the class area and the variable l and b, then we have overloaded the constructor as one default constructor and another parameter constructor and also declared destructor to clear the data and in the main function we have declared the object for the area class and printed the required area.

PRACTICAL 4.2:

In this program, first we have declared the class student and the variables then we have declared the display() function to print the values and declared constructor and copy constructor respectively and declared destructor and in the main function we have inputted the value of the variables from the user and printed them after declaring the object for the class and accessing display() function to print it.

OUTPUT:-

PRACTICAL 4.1:-

```
input
Simple constructor called
length=5
breadth=6
Area is : 30
Enter length and breadth for parameterised        constructor:
17
19
Area is : 323
Object is being deleted
Object is being deleted


...Program finished with exit code 0
Press ENTER to exit console.
```

PROGRAM- 4.2:

```
input
Student details:
Name:Vinay,Roll Number:9329,Height:5,Weight:45
destructor is called

destructor is called


...Program finished with exit code 0
Press ENTER to exit console.
```

LEARNING OUTCOMES

- Understand the concepts of object-oriented programming including programming process and compilation process.
- Apply different techniques to decompose a problem and programmed a solution with its sub modules.
- Analyze and explain the behavior of simple programs involving the programming addressed in the course.
- Implement and evaluate the programs using the syntax and semantics of object-oriented programming.
- Design the solution of real-world problems in order to determine that the program performs as expected.

EVALUATION COLUMN (To be filled by concerned faculty only)

Sr. No.	Parameters	Maximum Marks	Marks Obtained
1.	Worksheet Completion including writing learning objective/ Outcome	10	
2.	Post-Lab Quiz Result	5	
3.	Student engagement in Simulation/ Performance/ Pre-Lab Questions	5	
4.	Total Marks	20	