# Experiment 5.a

Date 26-September-2018

**Aim:** To write a program to increment ++ and decrement – variables by using operator overloading in c++.

## Algorithm:

1: Start.

2: Define a class as test with a variable a.

3: Define member function as get_data() to get value of variable a.

4: Define member functions as operator++() and operator--() and display().

5: Definition of main().

6: Create an object of class.

7: Call member functions of class using object of class.

8: Stop.

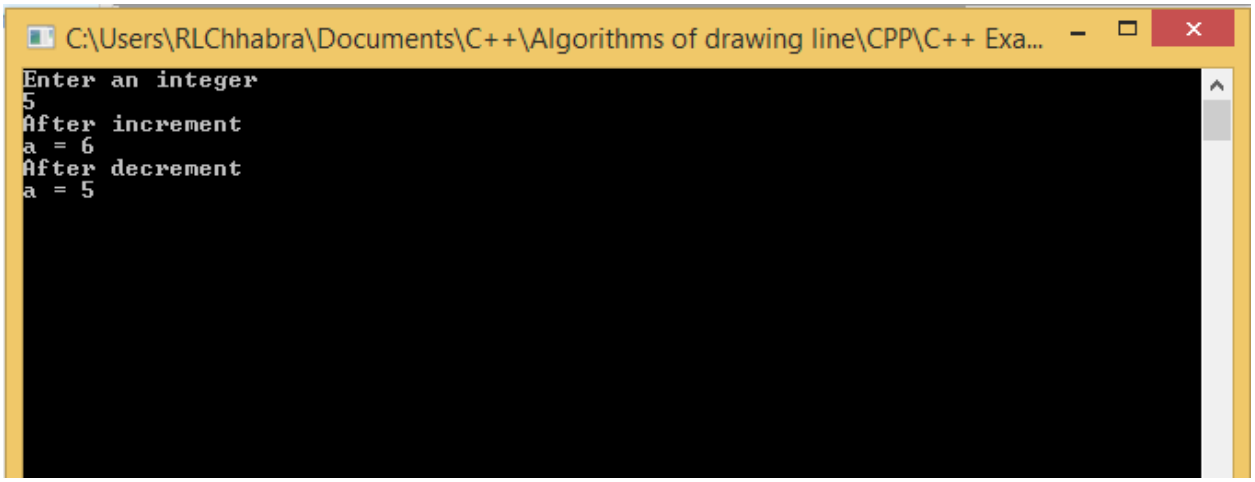
## Program:

```
#include<iostream>

#include<conio.h>

using namespace std ;

class test

{       private:    int a ;

        public:     get_data()

                {       cout<<"Enter an integer"<<endl;   cin>>a;   }

        void operator++()

                {               ++a ;           }
```

```
        void operator--()
        {       --a ;   }
        void display()
        {       cout<<"a = "<<a<<endl;      }   };
int main()
{       test obj ;
        obj.get_data();   ++obj ;
        cout<<"After increment"<<endl;
        obj.display();
        cout<<"After decrement"<<endl;   --obj ;
        obj.display();   getch();
        return 0 ; }
```

**Input Given:** a = 5 ;

**Output:**



# Experiment 5.b

Date 26-Sep-2018

**Aim:** To write a program to concatenate two strings using operator in c++.

## Algorithm:

1: Start.

2: Define a class as String with variable str1.

3: Define member variables as get_str(), operator+() and display().

4: Definition of main().

5: Create an class object.

6: Call class member functions using object.

7: Stop.

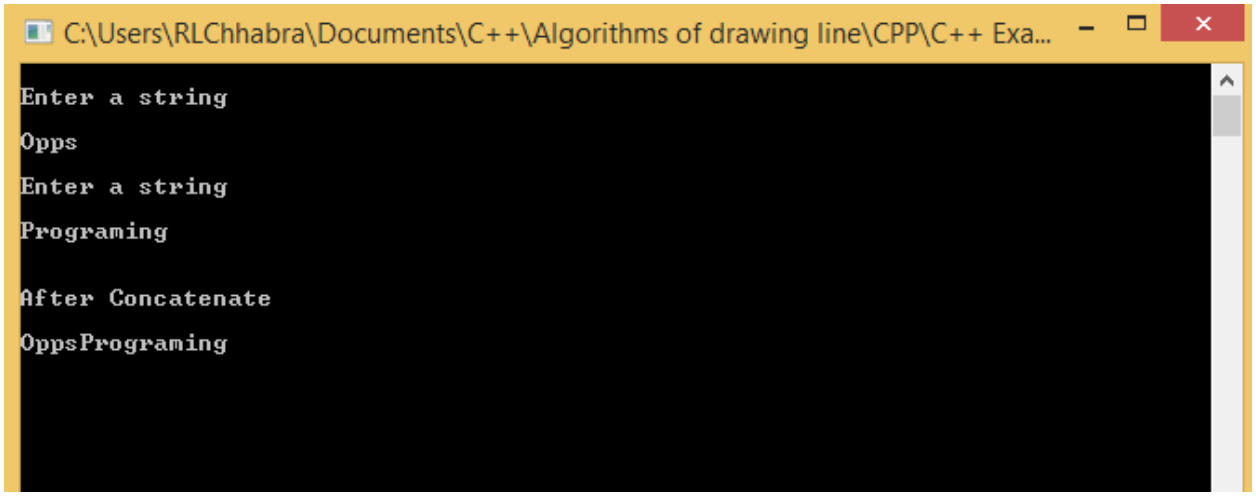
## Program:

```
#include<iostream>

#include<conio.h>

#include<string.h>

using namespace std ;

class String

{   private:  string str1 ;

    public:   void get_str()   {      cout<<"\nEnter a string\n"<<endl;

                                      getline(cin, str1);  }

              String operator+(String obj)

                                {         String s ;              s.str1 = str1 + obj.str1 ;     return s ;     }
```

```
        void display()
        {       cout<<"\n\nAfter Concatenate\n"<<endl;
                cout<<str1<<endl;    }   };
int main()
{       String s1 , s2 , s3  ;
        s1.get_str();
        s2.get_str();
        s3 = s1 + s2 ;
        s3.display();
        getch();
        return 0 ;    }
```

**Input Given:** str1 = “Opps” , str1 = “Programing” ;

**Output:**



# Experiment 5.c

Date-26-Sep-2018

**Aim:** To write a program to subtract two complex numbers using binary operator in c++.

## Algorithm:

1: Start.

2: Define a class as complex with variables a and b.

3: Define member functions as get_data() and operator-() and display().

4: Definition of main().

5: Create a object of class.

6: Call class member functions using object.

7: stop.

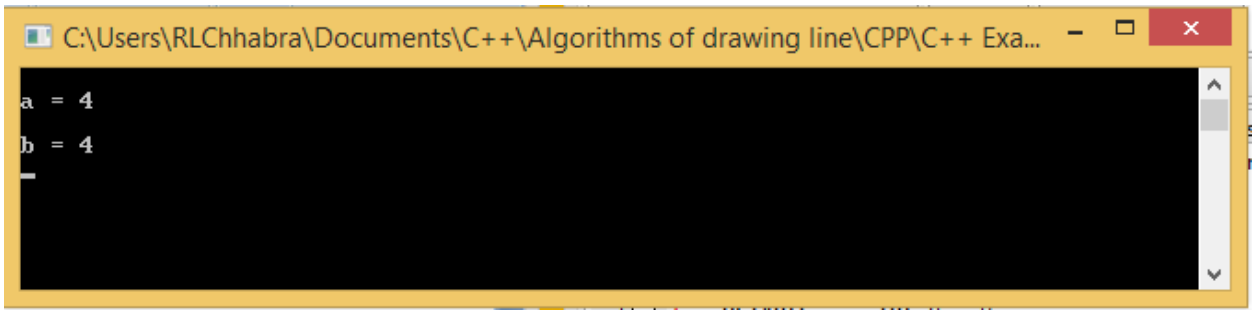
## Program:

```
#include<iostream>

#include<conio.h>

using namespace std ;

class complex

{                                   private: int a , b ;

   public:

                                    void get_data( int a , int b )

                                    {       this->a = a;    this->b = b ;   }

                                     complex operator-(complex o)

                                     {      complex temp ;
```

```
                temp.a = a - o.a ;
                temp.b = b - o.b ;
                return temp ;  }
        void display()
        {       cout<<"a = "<<a<<"\nb = "<<b<<endl;       }  } ;
int main()
{                       complex c1 , c2 , c3 ;
                        c1.get_data(8,9);
                        c2.get_data(4,5);
                        c3 = c1 - c2 ;
                        c3.display();
                        getch();
                        return 0 ;   }
```

**Input Given:** for first object a = 8 , b = 9 and for second object a = 4 , b = 5 .

**Output:**



# Experiment 6.a

Date -26-Sep-2018

**Aim:** To write a program to read and print employees information using multiple inheritance in c++.

## Algorithm:

1: Start.

2: Define a base class as personal with variables name , gender and age.

3: Define member function as get() to take value of variables.

4: Define a base class as professional with variables company_name , employee_id and salary.

5: Define a member function as get_data().

6: Define derived(personal and professional inherite) class as employee.

7: Define member functions as get_info() , display().

8: Definition of main().

9: Declare a variable n.

10: Create strings of objects of employee class as e[n].

11: Call employee class member functions with help of object.

12: Stop.

## Program:

```
#include<iostream>

#include<conio.h>

#include<stdlib.h>

using namespace std ;

class personal
```

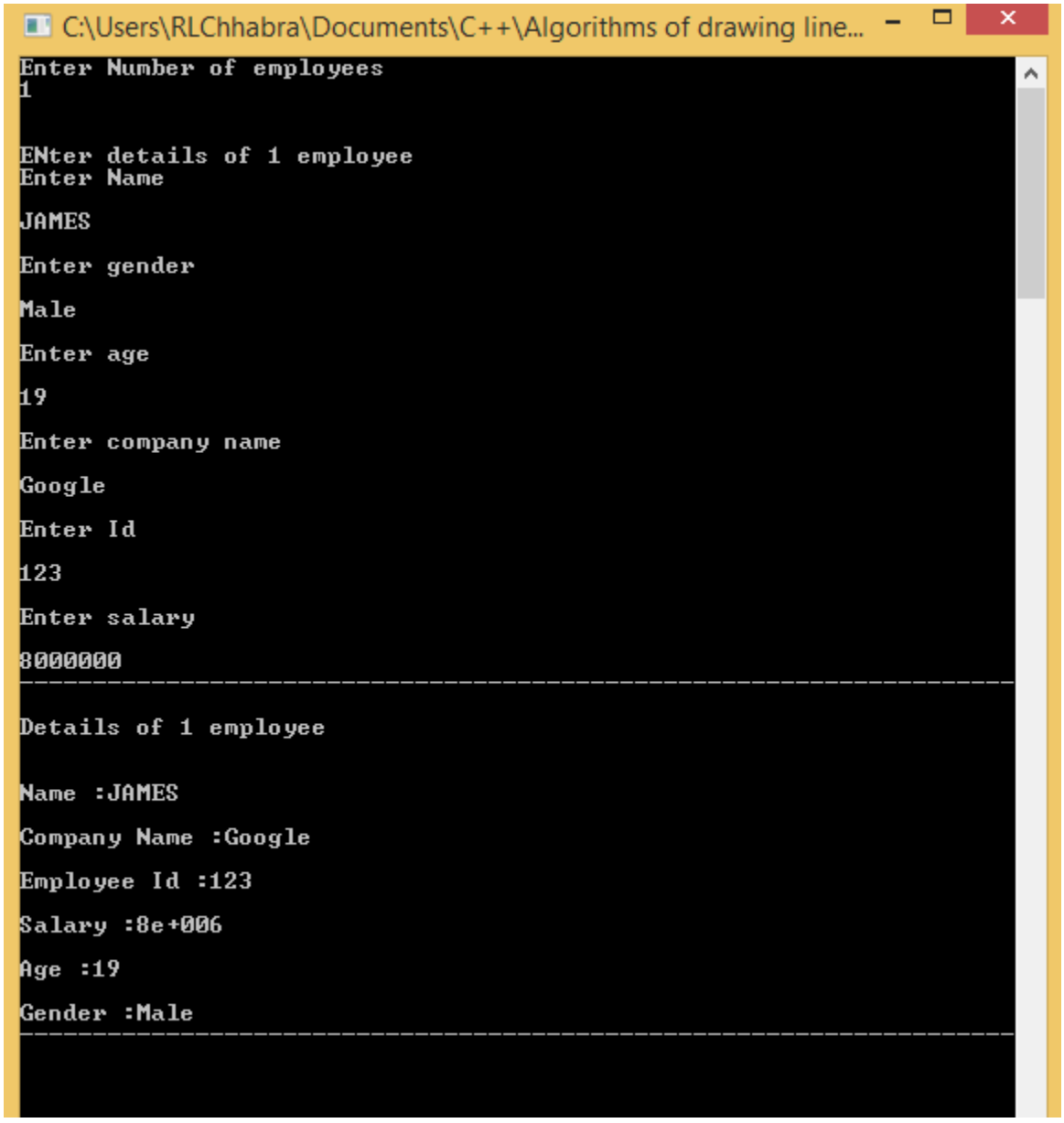
```
{
   protected:
                    string name ;
                    string gender ;
                    int age ;
                    void get()
                    {   cout<<"Enter Name"<<endl;
                       fflush(stdin);
                       getline(cin,name);
                       cout<<"Enter gender"<<endl;
                       getline(cin,gender);
                       cout<<"Enter age"<<endl;
                       cin>>age;     }    };
class professional
{
          protected:
                    string company_name ;
                    int employee_id ;
                    float salary ;
                    void get_data()
                    {   cout<<"Enter company name"<<endl;
                              fflush(stdin);
                              getline(cin,company_name);
                              cout<<"Enter Id"<<endl;
```

```
                cin>>employee_id;
                cout<<"Enter salary"<<endl;
                cin>>salary;    }   };
class employee:public personal , public professional
{       public:
                void get_info()
                {       get();   get_data();     }
                void display()
                {       cout<<"Name :"<<name<<endl;
                        cout<<"Company Name :"<<company_name<<endl;
                        cout<<"Employee Id :"<<employee_id<<endl;
                        cout<<"Salary :"<<salary<<endl;
                        cout<<"Age :"<<age<<endl;
                        cout<<"Gender :"<<gender<<endl;    }    };
int main()
{       int n ;
        cout<<"Enter Number of employees"<<endl;
        cin>>n ;
        employee e[n];
        for(int i=0 ; i<n ; i++)
        {       cout<<"\n\nENter details of "<<i+1<<" employee"<<endl;
                e[i].get_info(); }
        for(int i=0 ; i<n ;i++)
        {       cout<<"\n\nDetails of "<<i+1<<" employee\n"<<endl;
```

```
            e[i].display();   }          getch();

        return 0 ;    }
```

**Input Given:** n = 1 , Name = “JAMES” Company_name = “Google” Gender = “Male” Age = 19 , Salary = 8000000/- Employee_id = 123.

## Output:



# Experiment 7.a

Date-26-Sep-2018

**Aim:** To write a program to read & print a student information using single inheritance in c++.

## Algorithm:

1: Start.

2: Define base class as get_info with variables name , college , branch , roll.

3: Define member function of class as get_data().

4: Define derived class of get_info as student.

5: Define member function as get() and display().

6: Definition of main().

7: Create objects of student class.

8: Call member functions using objects.

9: Stop.

## Program:

```
#include<iostream>

#include<conio.h>

#include<stdlib.h>

using namespace std ;

class get_info

{           protected:  string name ;   string college;   string branch;   int roll ;

                    void get_data()
```

```
                { fflush(stdin);
                  cout<<"Enter Sudent name"<<endl; getline(cin,name);
              cout<<"Enter branch name"<<endl; getline(cin,branch);
                  cout<<"Enter roll number"<<endl; cin>>roll; fflush(stdin);
                  cout<<"Enter college name"<<endl; getline(cin,college);          } };
class student : public get_info
{       public:
                void get()
                {               get_data();     }
                void display()
                {       cout<<"Name :"<<name<<endl;
                        cout<<"Roll No:"<<roll<<endl;
                        cout<<"Branch :"<<branch<<endl;
                        cout<<"College :"<<college<<endl;   } };
int main()
{       int n ;
        cout<<"Enter number of students"<<endl;   cin>>n ;
        student s[n];
        for(int i=0 ; i<n ; i++)
        {       cout<<"ENter detail of "<<i+1<<" student"<<endl;   s[i].get();       }
        for(int i=0 ; i<n ; i++)
        {       cout<<"\nDisplay details of "<<i+1<<" Stuent\n"<<endl;
                s[i].display();   }         getch();
        return 0 ;   }
```

**Input Given:** Name = “Bob” , Branch = “CSE” , Roll = 1 , College = “IIT Bomby”.

**Output:**