

ASSIGNMENT 5

```
// assignment 5 program 1
```

```
/*Illustrate the program to demonstrate public, private and protected members of a class with  
respect to publicly inherited derived class. */
```

```
#include<iostream>
```

```
using namespace std;
```

```
class Parent{
```

```
    private:
```

```
        int a,b;
```

```
    protected:
```

```
        int c;
```

```
    public:
```

```
        int get_a();
```

```
        int get_b();
```

```
        // void getdata();
```

```
        void putdata();
```

```
        void get_ab();
```

```
        void get_c(int x){
```

```
            c=x;
```

```
            cout << endl << "c= " << c;
```

```
        }
```

```
};
```

```
class D:public Parent{
```

```
    public:
```

```
        void mul();
```

```
void display();  
};  
void Parent::get_ab(){  
    a=10;  
    b=23;  
}  
void Parent::putdata(){  
    cout << "\n " << "a = " << a << "b =" << b;  
}  
int Parent::get_a(){  
    return a;  
}  
int Parent::get_b(){  
    return b;  
}  
void D::mul(){  
    c=get_a()*get_b();  
}  
void D::display(){  
    cout << "\n Access to private member of a parent class in child class";  
    cout << "\n a =" << get_a();  
    cout << "\n b =" << get_b();  
    cout << "\n Private member access is done ";  
    cout << "\n Multiplication is =" << c;
```

```

        cout << "\n Multiplication done";
    }
int main(){
    D d;

    d.get_ab();

    cout << "\n Multiplication using private member access of a parent class \n ";

    d.mul();

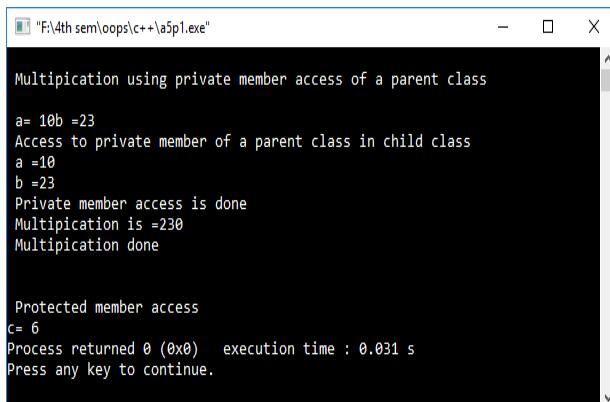
    d.putdata();

    d.display();

    cout << "\n\n\n Protected member access " ;

    d.get_c(6); return 0 ; }

```



```

"F:\4th sem\oops\c++\a5p1.exe"
Multiplication using private member access of a parent class
a= 10b =23
Access to private member of a parent class in child class
a =10
b =23
Private member access is done
Multiplication is =230
Multiplication done

Protected member access
c= 6
Process returned 0 (0x0)   execution time : 0.031 s
Press any key to continue.

```

// assignment 5 program 2

/*"Declare a class called item having data members item_code, item_name, cost and discount.

Derive two classes from class item, namely employee and customer. The class employee has data members like employee_code, employee_name and amount.

The class customer has data members like customer_name and amount. Define following functions for

- initializing data members.

- displaying the values of data members.

- computing amount to be paid for a purchased item. Also define function main to create objects of both derived classes and to show usage of above functions."*/

```
#include<iostream>
```

```
using namespace std;
```

```
class item{
```

```
protected:
```

```
    int item_code;
```

```
    float cost,discount;
```

```
    char item_name[20];
```

```
        float amount;
```

```
public:
```

```
    void initicode();
```

```
    void display();
```

```
};
```

```
void item::initicode(){
```

```
    cout << "\n Enter item code :";
```

```

        cin >> item_code;
        cout << "\n Enter cost ";
        cin >> cost;
        cout << "\n Enter discount ";
        cin >> discount;
        // cout << "\n Enter amount to be paid :";
        //      cin >> amount;
    }
    void item::display(){
        cout << "\n Item code is " << item_code;
        cout << "\n Cost is " << cost;
        cout << "\n Discount is " << discount;
        cout << "\n Amount is " << amount;
    }
    class employee:public item{
        int employee_code;
        char employee_name[20];
    public:
        void initiemp(){
            cout << "\n Enter employee code";
            cin >> employee_code;
            cout << "\n Enter employee name";
            cin >> employee_name;
        }
    }

```

```

        void eamount();
};

void employee::eamount(){
    initiemp();
    initicode();
    display();
    amount =cost-(cost*discount)/100;
    cout << "\n Amount to be paid for employees " << amount;
}

class customer:public item{
    char customer_name[20];
public:
    void initicust(){
        cout << "\n Enter customer name ";
        cin >> customer_name;
    }
    void camount();

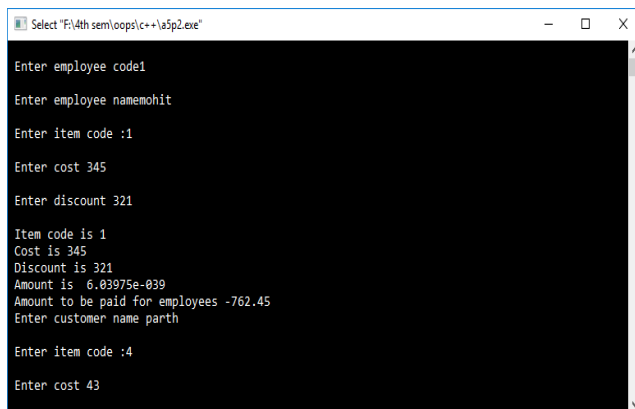
};

void customer::camount(){
    initicust();
    initicode();
    display();
    amount=cost-(cost*amount)/100;

```

```
    cout << "\n Amount is " << amount;  
}
```

```
int main(){  
    employee e;  
    customer c;  
    e.eamount();  
    c.camount();  
    return 0; }
```



The screenshot shows a Windows command prompt window titled "Select 'F:\4th sem\oops\c++\a5p2.exe'". The program prompts for employee and customer data. The user enters '1' for employee code, 'mohit' for employee name, '1' for item code, '345' for cost, and '321' for discount. The program then displays the calculated amount for the employee as -762.45. The user then enters '4' for item code and '43' for cost, but the output for the customer's amount is not visible in the screenshot.

```
Select "F:\4th sem\oops\c++\a5p2.exe"  
Enter employee code:1  
Enter employee name:mohit  
Enter item code :1  
Enter cost 345  
Enter discount 321  
Item code is 1  
Cost is 345  
Discount is 321  
Amount is -6.03975e+039  
Amount to be paid for employees -762.45  
Enter customer name parth  
Enter item code :4  
Enter cost 43
```

// Assignment 5 programme 3

/*Create a class student that stores roll_no, name.

Create a class test that stores marks obtained in five subjects and percentage.

Class result derived from test and test contains the total marks and percentage obtained in test.

The class result can inherit the details of marks obtained in test and roll number of the students through multilevel inheritance.*/

```
#include<iostream>
```

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
using namespace std;
```

```
class student {
```

```
    protected:
```

```
        int rollno;
```

```
        char name[20];
```

```
    public:
```

```
        void getdata();
```

```
};
```

```
void student::getdata(){
```

```
    cout << "\n Enter roll no and name:";
```

```
    cin >> rollno >> name;
```

```
}
```

```
class test:public student{
```

```
    protected:
```

```
        float m[5];
```

```
        float percentage;
```

```
    public:
```

```

        void get_test();
};
void test::get_test(){
float sum;
for(int i=0;i<5;i++){
        cout << "\n Enter mark of subject " << i+1 << ":\t";
        cin >> m[i];
        sum=sum+m[i];
    }
    percentage=(sum/500)*100;
}
class result:public test{
float total;
    public:
        void showdata();
};
void result::showdata(){
    for(int i=0;i<5;i++){
        total=total+m[i];
    }

    cout << "\n Roll no :" << rollno ;
    cout << "\n Marks :" << total;
    cout << "\n Name :" << name;
    cout << "\n percentage :" << percentage;
}

```

```
}
```

```
int main(){
```

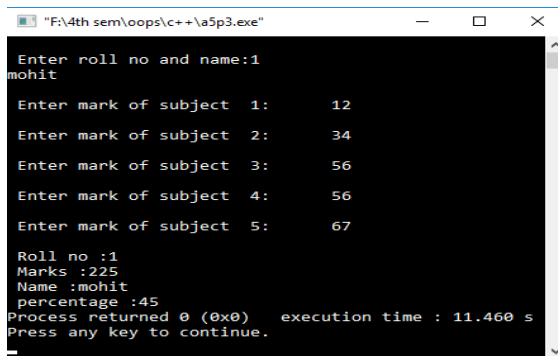
```
    result r;
```

```
    r.getdata();
```

```
    r.get_test();
```

```
    r.showdata();
```

```
    return 0; }
```



```
"F:\4th sem\oops\c++\a5p3.exe"
Enter roll no and name:1
mohit
Enter mark of subject 1: 12
Enter mark of subject 2: 34
Enter mark of subject 3: 56
Enter mark of subject 4: 56
Enter mark of subject 5: 67
Roll no :1
Marks :225
Name :mohit
percentage :45
Process returned 0 (0x0)   execution time : 11.460 s
Press any key to continue.
```

// Assignment 5 programme 4

/*"Assume that Circle is defined using radius and Cylinder is defined using radius and height.

Write a Circle class as base class and inherit the Cylinder class from it.

Develop classes such that user can compute the area of Circle objects and volume of Cylinder objects.

Area of Circle is $\pi * \text{radius} * \text{radius}$,
while volume of Cylinder is $\pi * (\text{radius} * \text{radius}) * \text{height}$."*/

```
#include<iostream>
```

```
using namespace std;
```

```
#define pi 3.14
```

```
class circle{
```

```
protected:
```

```
    float radius;
```

```
public:
```

```
    void getradius();
```

```
    void area_circle();
```

```
};
```

```
void circle::getradius(){
```

```
    cout << "\n Enter radius of a circle :";
```

```
    cin >> radius;
```

```
}
```

```
void circle :: area_circle(){
```

```
    float area= pi*radius*radius;
```

```

        cout << "\t" << area;}

class cylinder{
    protected:
        float height;
        float radius;
    public:
        void getheight();
        void getradius1();
        void volume_cylinder();
};

void cylinder::getheight(){
    cout << "\n Enter a height for a cylinder : ";
    cin >> height;}

void cylinder::getradius1(){
    cout << "\n Enter radius for a cylinder : ";
    cin >> radius;
}

void cylinder::volume_cylinder(){
    float volume;

    volume=pi*radius*radius*height;

    cout << "\t" << volume ;}

class result:public circle,public cylinder{
    public:
        void display_result();

```

```

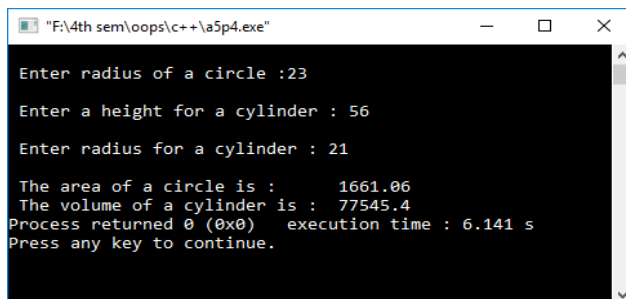
};

void result::display_result(){
    cout << "\n The area of a circle is : " ;
        area_circle();
    cout << "\n The volume of a cylinder is :";
        volume_cylinder();
}

int main(){
    result r1;
    r1.getradius();
    r1.getheight();
    r1.getradius1();
    r1.display_result();

    return 0;}

```



```

F:\4th sem\oops\c++\a5p4.exe
Enter radius of a circle :23
Enter a height for a cylinder : 56
Enter radius for a cylinder : 21
The area of a circle is :      1661.06
The volume of a cylinder is :  77545.4
Process returned 0 (0x0)   execution time : 6.141 s
Press any key to continue.

```

```
// Assignment 5 programme 5

/*"Consider a class network as shown in figure given below. The class Employee
derives information from both Account and Admin classes which in turn
derive information from the class Person. Define all the four classes and write
a program to create, update and display the information contained in
Employee objects."*/

#include<iostream>

using namespace std;

class Person{
protected:
    char name[20];
    int code;
};

class Account: virtual public Person{
protected:
    float pay;

};

class Admin: virtual public Person{
protected:
    int experience;

};
```

```

class Employee:public Account,public Admin{
    public:
        void intiemp();
        void update();
        void display();

};

void Employee::intiemp(){
    cout << "\n Enter name of an employee : ";
    cin >> name;
    cout << "\n Enter code for an employee : ";
    cin >> code;
    cout << "\n Enter experience for an employee :";
    cin >> experience;
    cout << "\n Enter amount to be paid to the employee :";
    cin >> pay;

}

void Employee::update(){
    cout << "\nWhat you want to update :";
    cout << "\n1.update a name :";
    cout << "\n2.update a code :";
    cout << "\n3.update a experience :";

```



```

cout << "\n4.update a pay :";
int option;
cout << "\n Enter your option :";
cin >> option;
switch(option){
    case 1:
        cout << "\n Enter a name that you want to update :";
        cin >> name;
        break;
    case 2:
        cout << "\n Enter a code that you want to update :";
        cin >> code;
        break;
    case 3:
        cout << "\n Enter a no. of years of experience that you want to
update :";
        cin >> experience;
        break;
    case 4:
        cout << "\n Enter no. of payment that you want to change :";
        cin >> pay;
        break;
    default:
        cout << "\n Wrong choice ";
        break;
}

```

```

}
}

void Employee::display(){
cout << "\nName :" << name << endl << "Code :" << code << "\nExperience :" <<
experience << "\nPay :" << pay << endl;
}

int main(){
    Employee e[3];
    for(int i=0;i<2;i++){
        cout << "\n Enter information for the employee :" << i+1 ;
        e[i].inttemp();
    }
    for(int i=0;i<2;i++){
        e[i].display();
    }

    for(int i=0;i<2;i++){
        cout << "\n Enter an updation for employee :" << i+1;
        e[i].update();
    }
    for(int i=0;i<2;i++){
        e[i].display();
    }

    return 0;}

```

```
"F:\4th sem\oops\c++\a5p5.exe"

Enter information for the employee :2
Enter name of an employee : parth
Enter code for an employee : 2
Enter experience for an employee :45
Enter amount to be paid to the employee :345

Name :mohit
Code :1
Experience :34
Pay :456

Name :parth
Code :2
Experience :45
Pay :345

Enter an updation for employee :1
What you want to update :
1.update a name :
2.update a code :
3.update a experience :
4.update a pay :
Enter your option :1
```

// Assignment 5 // Programme 6

/*Consider the following class structure as shown in the figure.

The class Result derives information from the classes Internal, University and External respectively.

The Internal and External classes access information from the Student class.

Define all five classes and write a suitable program to create and display the information contained in Result object.*/

```
#include<iostream>
```

```
using namespace std;
```

```
class student{
```

```
protected:
```

```
    char name[20];
```

```
    int en_ro;
```

```
public:
```

```
    void get_student()
```

```
{
```

```
    cout << "\n Enter name and enrollment no of the student :";
```

```
    cin >> name >> en_ro; }
```

```
};
```

```
class internal: virtual public student{
```

```
protected:
```

```
    float internal_mark;
```

```
public:
```

```
    void get_internalmark();
```

```

};

void internal::get_internalmark(){
    cout << "\n Enter internal marks of the student :";
    cin >> internal_mark;

}

class exterenal:virtual public student{
    protected:
        float external_mark;
    public:
        void get_externalmark();

};

void exterenal::get_externalmark(){
    cout << "\n Enter external marks of the students :";
    cin >> external_mark;
}

class university{
    protected:
        float university_mark;
    public:
        void get_universitymark();
};

void university::get_universitymark(){

```

```

        cout << "\n Enter university marks for the students :";

        cin >> university_mark;
    }

class result:public internal,public external,public university{
public:

    void show_result();

};

void result::show_result(){

    cout << "\n Name :" << name << "\n enrollment no :" << en_ro << "\n Internal
marks :" << internal_mark << "\n External marks :" << external_mark << "\n
University marks :" << university_mark << endl;

}

int main(){

    result student;

    student.get_student();

    student.get_internalmark();

    student.get_externalmark();

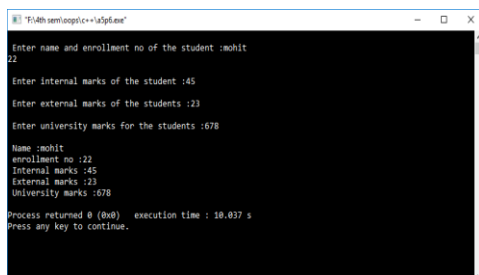
    student.get_universitymark();

    student.show_result();

    return 0;

}

```



```

C:\Users\user> g++ 143.cpp
Enter name and enrollment no of the student :mohit
22
Enter internal marks of the student :45
Enter external marks of the students :23
Enter university marks for the students :678
Name :mohit
enrollment no :22
Internal marks :45
External marks :23
University marks :678
Process returned 0 (0x0)   execution time : 10.037 s
Press any key to continue.

```

