**Assignment - 5**

**/\*Illustrate the program to demonstrate public, private and protected members of a class with**

**respect to publicly inherited derived class.\*/**

#include<iostream>

#include<conio.h>

using namespace std;

class B

{

int a; // private; not inheritable

public:

int b; // public; ready for inheritance

void get\_ab();

int get\_a();

void show\_a();

};

class D : private B // private derivation

{

int c;

public:

void mul(void);

void display(void);

};

void B :: get\_ab()

{

cout << "Enter values of a and b: ";

cin >> a >> b;

}

int B :: get\_a()

{

return a;

}

void B :: show\_a()

{

cout << "a = " << a << "\n";

}

void D :: mul()

{

get\_ab();

c = b \* get\_a(); // 'a' cannot be used directly

}

void D :: display()

{

show\_a(); // outputs value of 'a'

cout << "b=" << b << "\n" << "c=" << c << "\n\n";

}

int main()

{

D d;

//d.get\_ab(); // won't work

d.mul();

//d.show\_a(); // won't work

d.display();

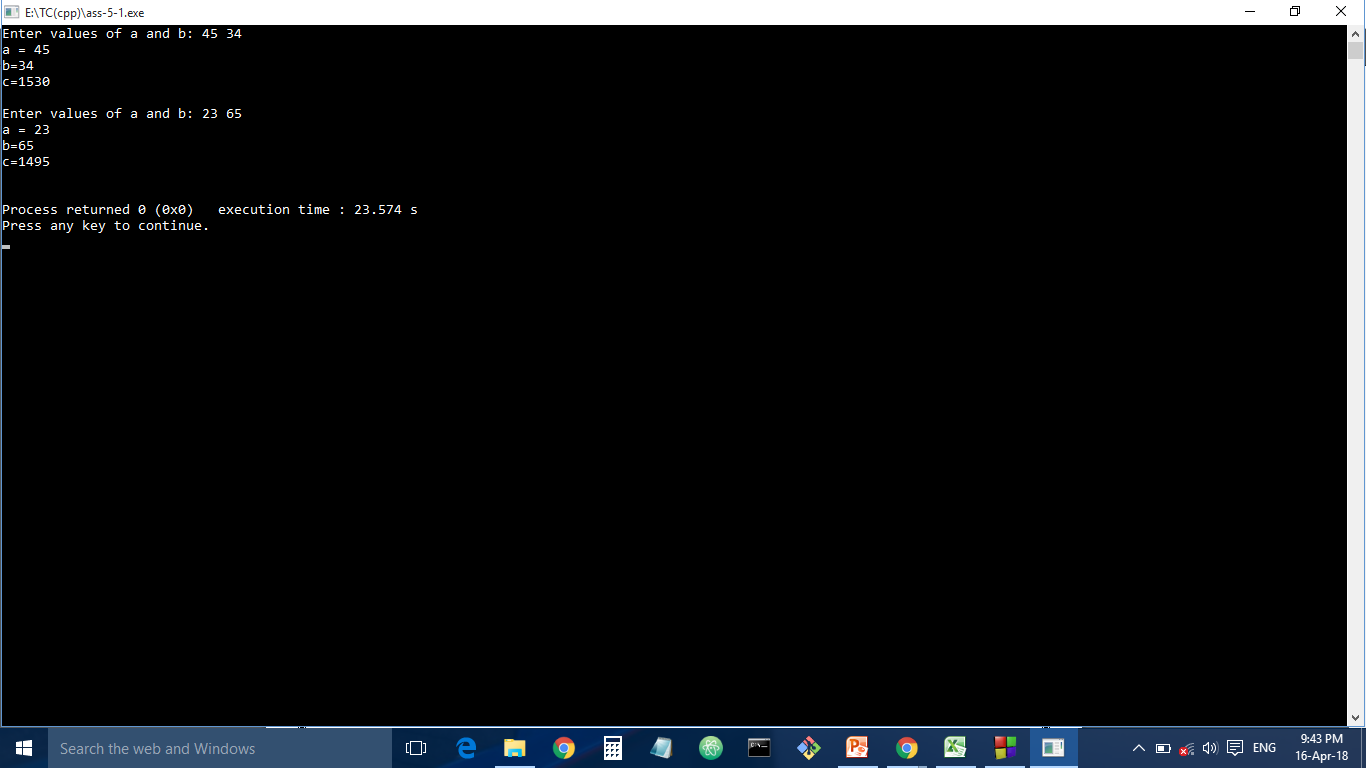
//d.b = 20; // won't work; 'b' has become private

d.mul();

d.display();

return 0;

}



**/\*"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>

#include<conio.h>

using namespace std;

class item

{

protected:

int item\_code;

string item\_name;

int cost;

int discount;

public:

void getdata()

{

cout<<"Enter item\_code,item\_name,cost,discount:";

cin>>item\_code>>item\_name>>cost>>discount;

}

};

class employee:public item

{

protected:

int employee\_code;

string employee\_name;

int amt;

public:

void setdata()

{

cout<<"Enter employee\_code,employee\_name:";

cin>>employee\_code>>employee\_name;

}

void amount()

{

getdata();

setdata();

amt=cost

- (cost\*discount)/100;

cout<<"Amount is "<<amt;

}

};

class customer:public item {

string customer\_name;

int amt;

public:

void setdata() {

cout<<"Enter customer\_name:";

cin>>customer\_name;

}

void amount() {

getdata();

setdata();

amt=cost

- (cost\*discount)/100;

cout<<"Amount is "<<amt;

}

};

int main() {

employee e1;

e1.amount();

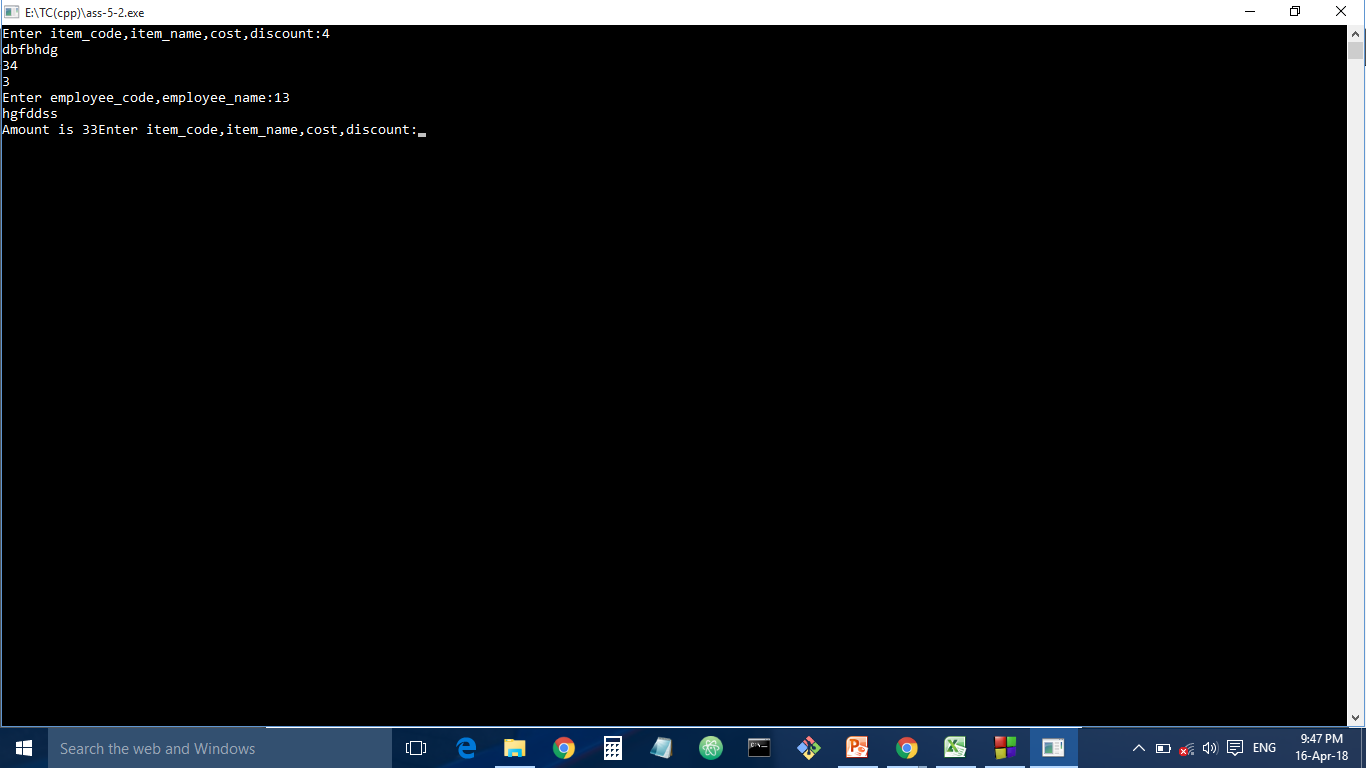
customer c1;

c1.amount();

getch();

return 0;

}



**/\*"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 Cylinde**

**class from it. Develop classes such that user can compute the a of Circle**

**objects and volume of Cylinder objects. a of Circle is pie \*radius\*radius,**

**while volume of Cylinder is pie\*(radius \* radius)\*height."\*/**

#include<iostream>

#include<conio.h>

using namespace std;

#define pie 3.14

class circle

{

protected:

int radius;

public:

circle()

{

cout<<"Enter radius:";

cin>>radius;

}

};

class cylinder:public circle

{

private:

float a;

float v;

int height;

public:

cylinder()

{

cout<<"Enter height:";

cin>>height;

}

void area()

{

a=pie\*radius\*radius;

}

void volume()

{

v=pie\*radius\*radius\*height;

}

void showdata()

{

area();

volume();

cout<<"a="<<a<<"\nvlume="<<v;

}

};

int main()

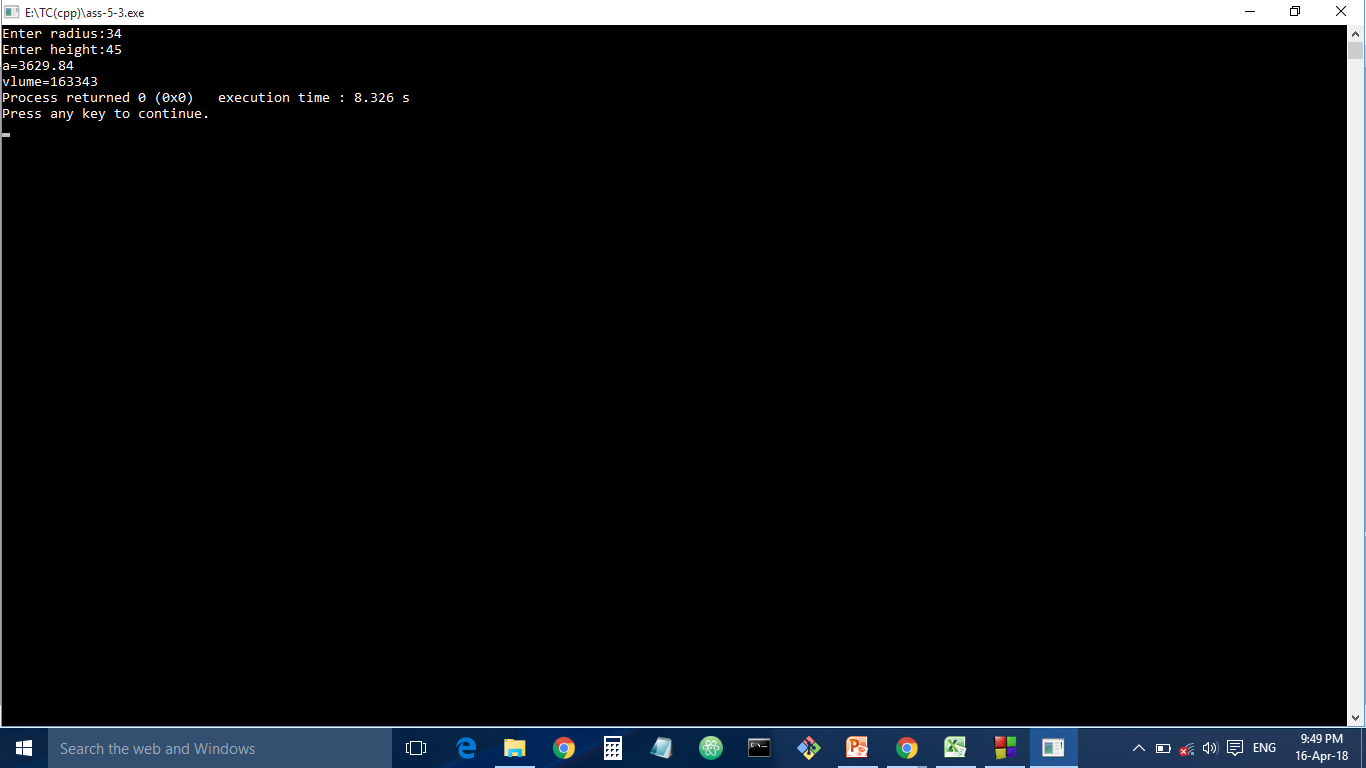
{

cylinder c1;

c1.showdata();

return 0;

}



**/\*"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 Cylinde**

**class from it. Develop classes such that user can compute the a of Circle**

**objects and volume of Cylinder objects. a of Circle is pie \*radius\*radius,**

**while volume of Cylinder is pie\*(radius \* radius)\*height."\*/**

#include<iostream>

#include<conio.h>

using namespace std;

#define pie 3.14

class circle

{

protected:

int radius;

public:

circle()

{

cout<<"Enter radius:";

cin>>radius;

}

};

class cylinder:public circle

{

private:

float a;

float v;

int height;

public:

cylinder()

{

cout<<"Enter height:";

cin>>height;

}

void area()

{

a=pie\*radius\*radius;

}

void volume()

{

v=pie\*radius\*radius\*height;

}

void showdata()

{

area();

volume();

cout<<"a="<<a<<"\nvlume="<<v;

}

};

int main()

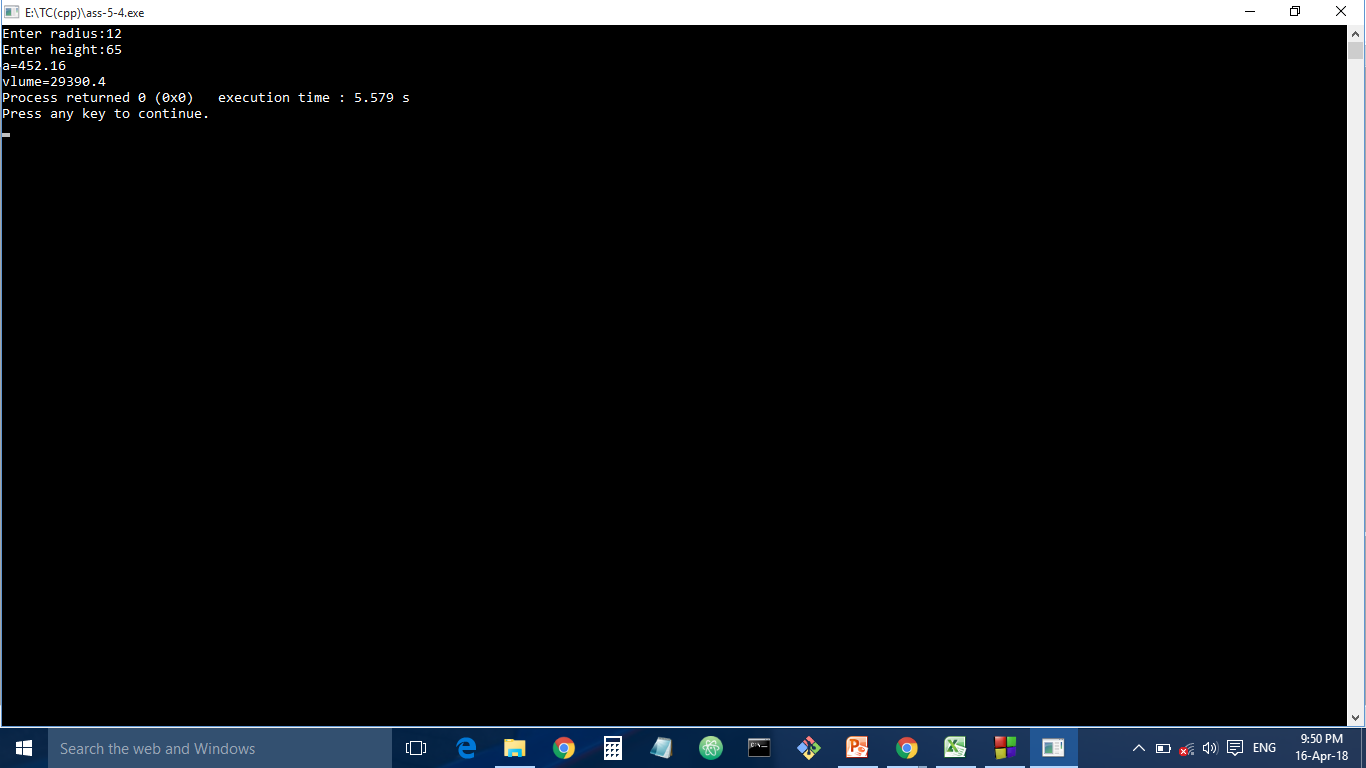
{

cylinder c1;

c1.showdata();

return 0;

}



**/\*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

{

public:

char name[20];

int code;

};

class account : virtual public person

{

public:

float pay;

};

class admin : virtual public person

{

public:

int exp;

};

class emp : public account, public admin

{

public:

void create()

{

cout<<"Name=";

cin>>name;

cout<<"Code=";

cin>>code;

cout<<"Pay=";

cin>>pay;

cout<<"Experience=";

cin>>exp;

}

void update()

{

int n;

cout<<"\nTo update\n";

cout<<"1.Name\n";

cout<<"2.Code\n";

cout<<"3.Pay\n";

cout<<"4.Experience\n";

cout<<"Enter your choice:";

cin>>n;

switch(n)

{

case 1: cout<<"Enter name:";

cin>>name;

break;

case 2: cout<<"Enter code:";

cin>>code;

break;

case 3: cout<<"Enter pay:";

cin>>pay;

break;

case 4: cout<<"Enter experience:";

cin>>exp;

break;

default: break;

}

}

void display()

{

cout<<"Name="<<name<<"\nCode="<<code<<"\nPay="<<pay<<"\nexperience="<<exp;

}

};

int main()

{

emp e1;

e1.create();

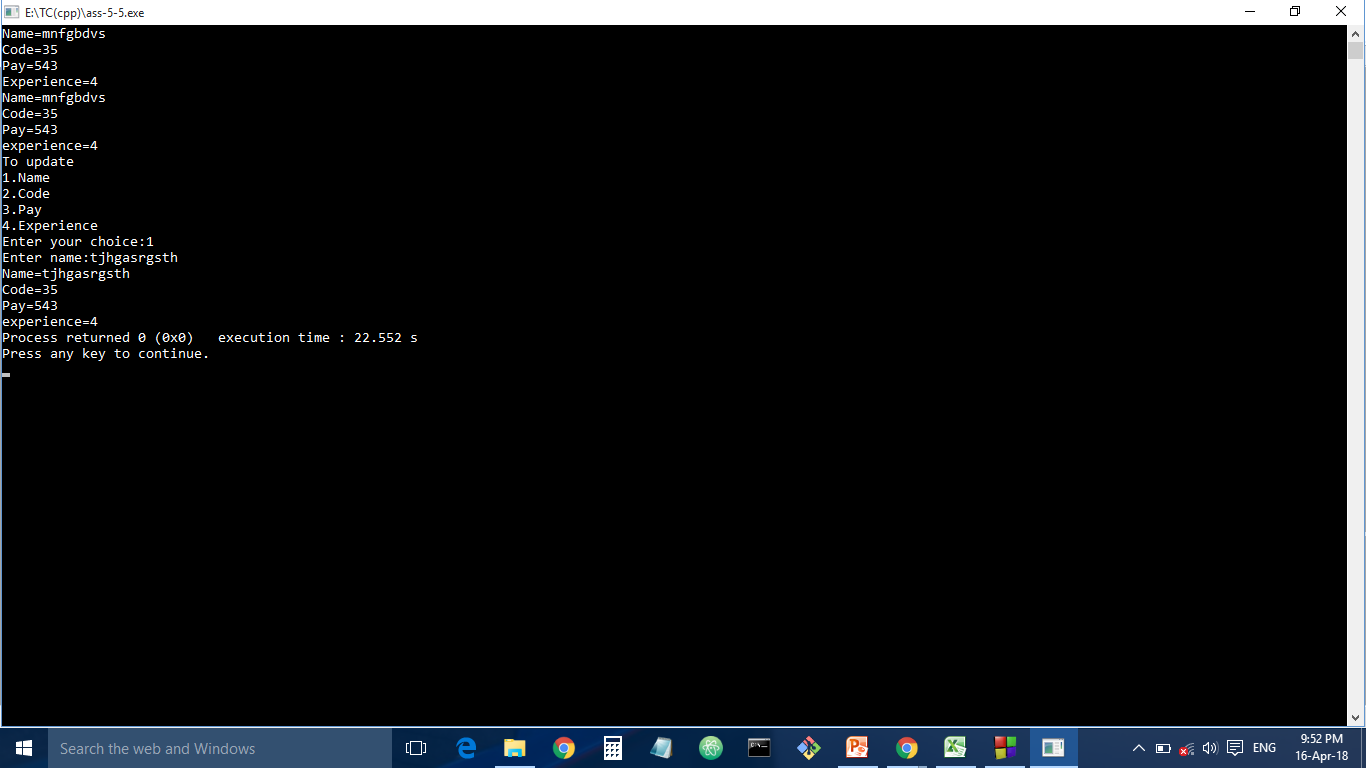
e1.display();

e1.update();

e1.display();

return 0;

}



**/\*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>

#include<conio.h>

using namespace std;

class student

{

protected:

string name;

string en\_no;

public:

student()

{

cout<<"Enter name:";

cin>>name;

cout<<"Enter Enrollment number:";

cin>>en\_no;

}

};

class internal:virtual public student

{

protected:

int internal\_m;

public:

internal()

{

cout<<"Enter internal marks:";

cin>>internal\_m;

}

};

class external:virtual public student

{

protected:

int external\_m;

public:

external()

{

cout<<"Enter external marks:";

cin>>external\_m;

}

};

class university

{

protected:

int university\_m;

public:

university()

{

cout<<"Enter university marks:";

cin>>university\_m;

}

};

class result:public internal,public external,public university

{

public:

void showdata()

{

cout<<" Name:"<<name<<"\n Enrollment number:"<<en\_no<<"\n internal marks"<<internal\_m<<"\n external marks:"<<external\_m<<"\n university marks:"<<university\_m;

}

};

int main()

{

result r1;

r1.showdata();

getch();

return 0;

}

