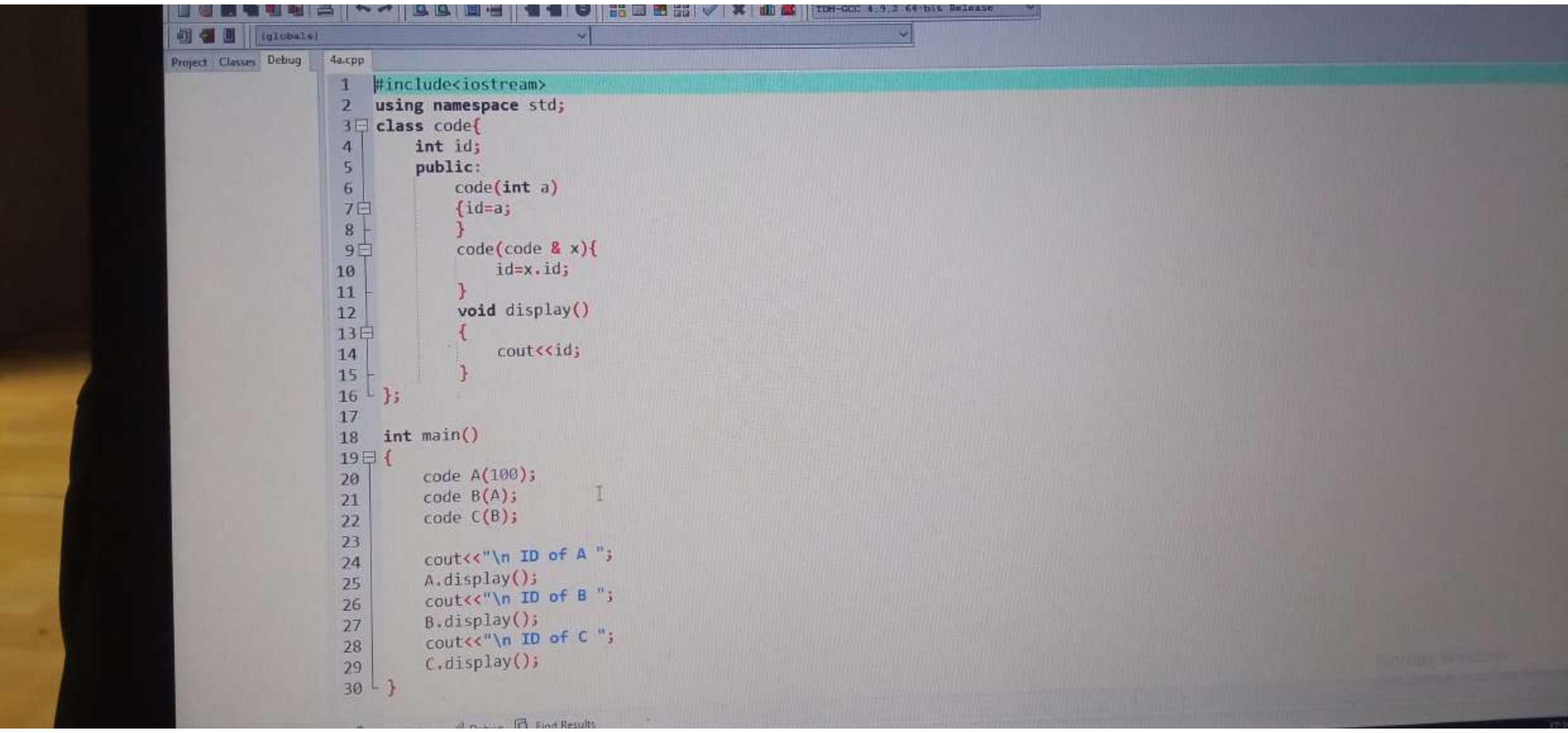
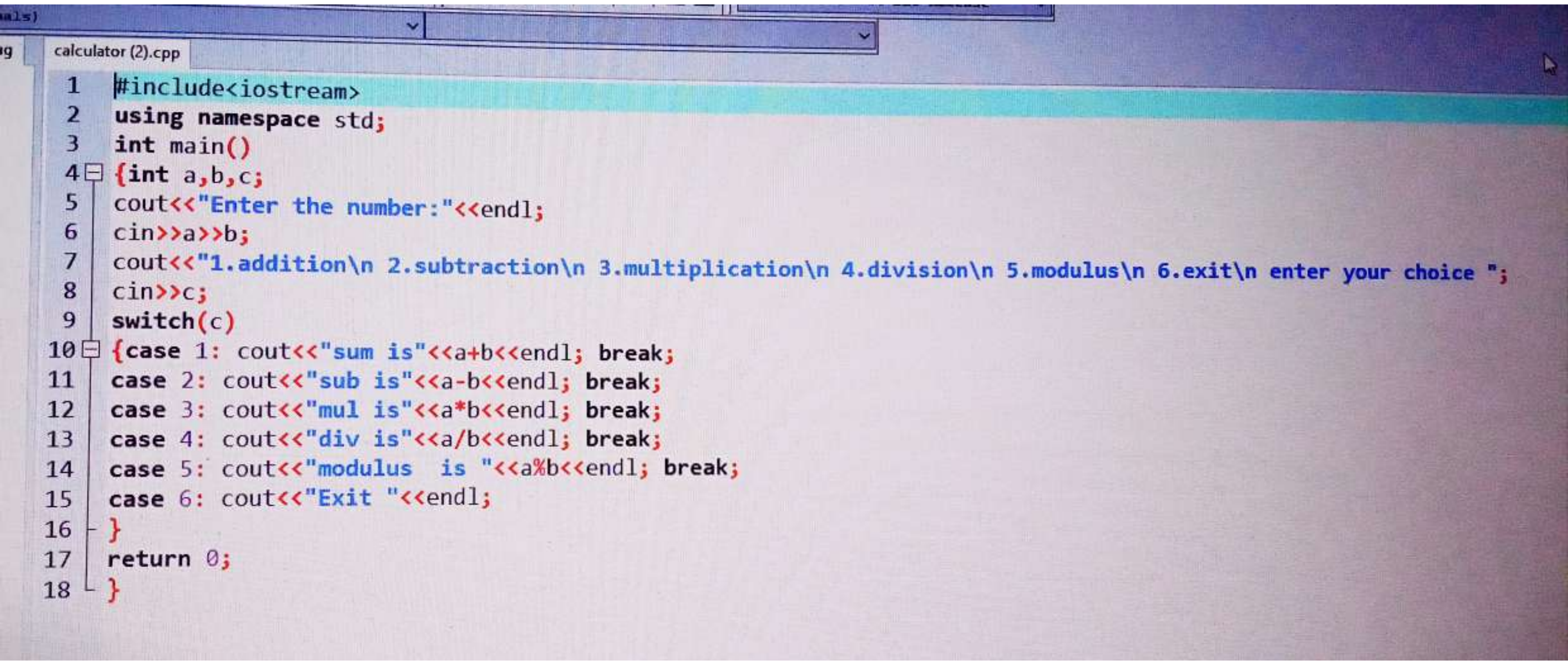
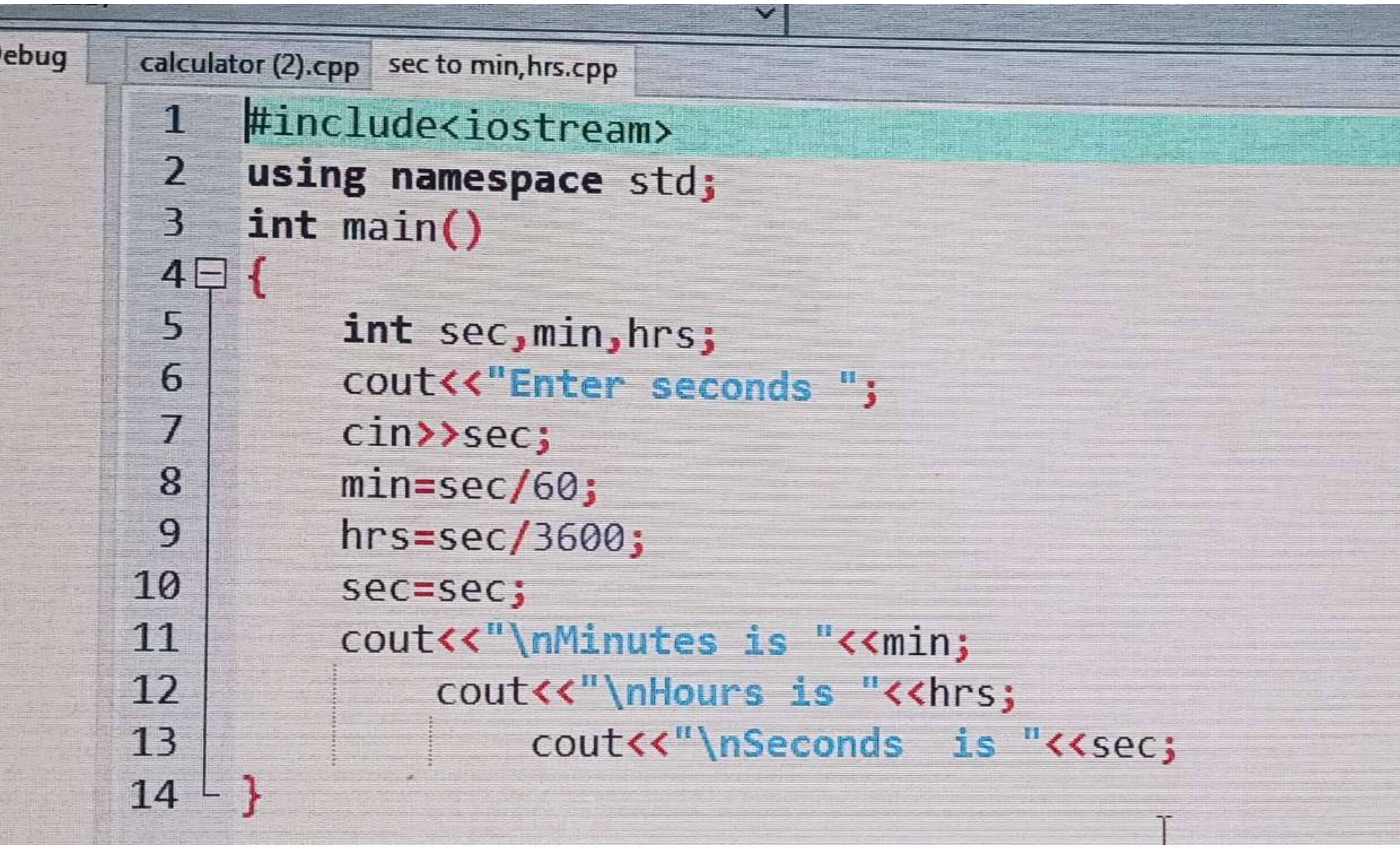


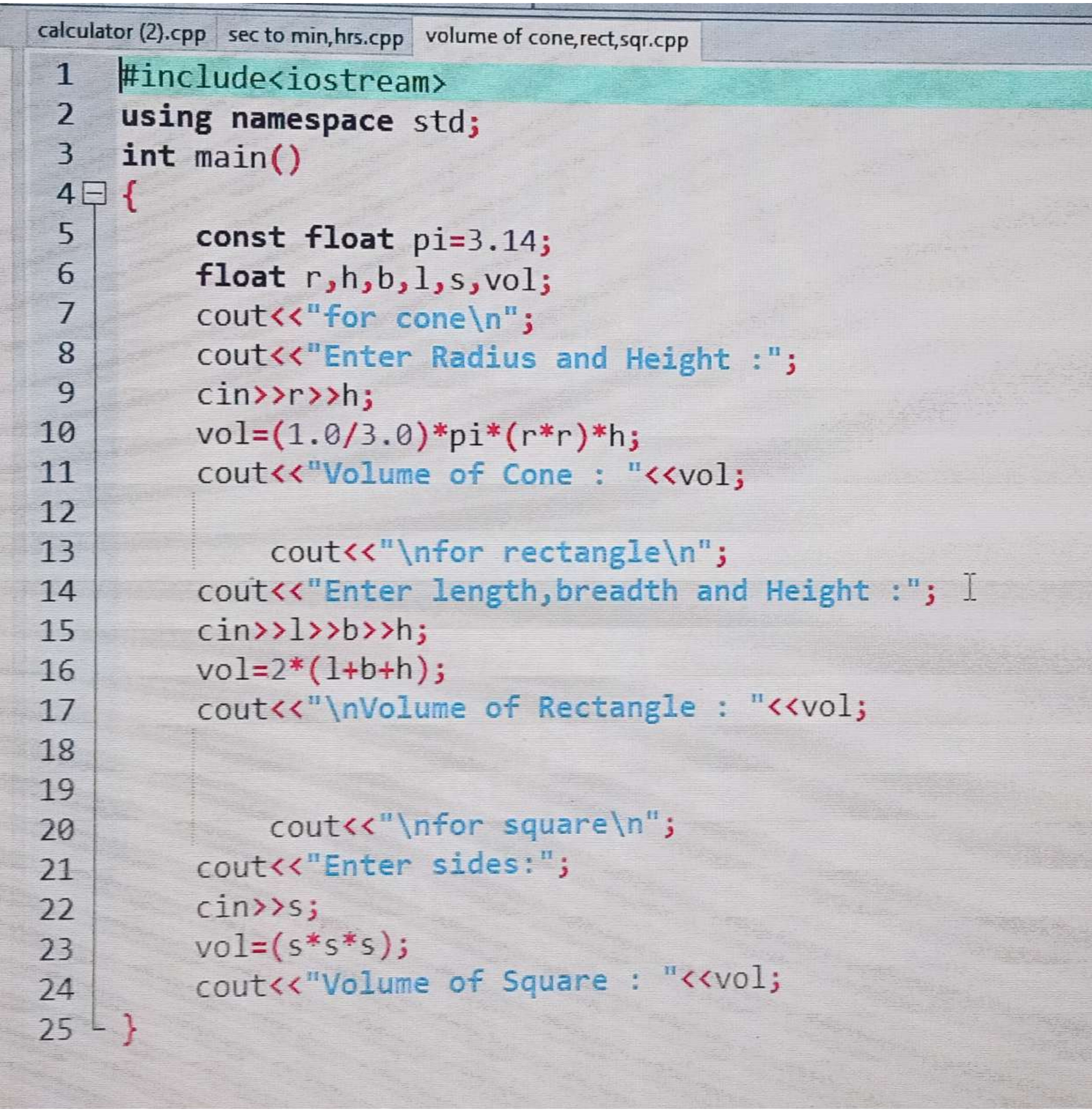
```cpp
#include<iostream>
using namespace std;
class code{
    int id;
    public:
        code(int a)
        {id=a;
        }
        code(code & x){
            id=x.id;
        }
        void display()
        {
            cout<<id;
        }
};

int main()
{
    code A(100);
    code B(A);
    code C(B);

    cout<<"\n ID of A ";
    A.display();
    cout<<"\n ID of B ";
    B.display();
    cout<<"\n ID of C ";
    C.display();
}
```



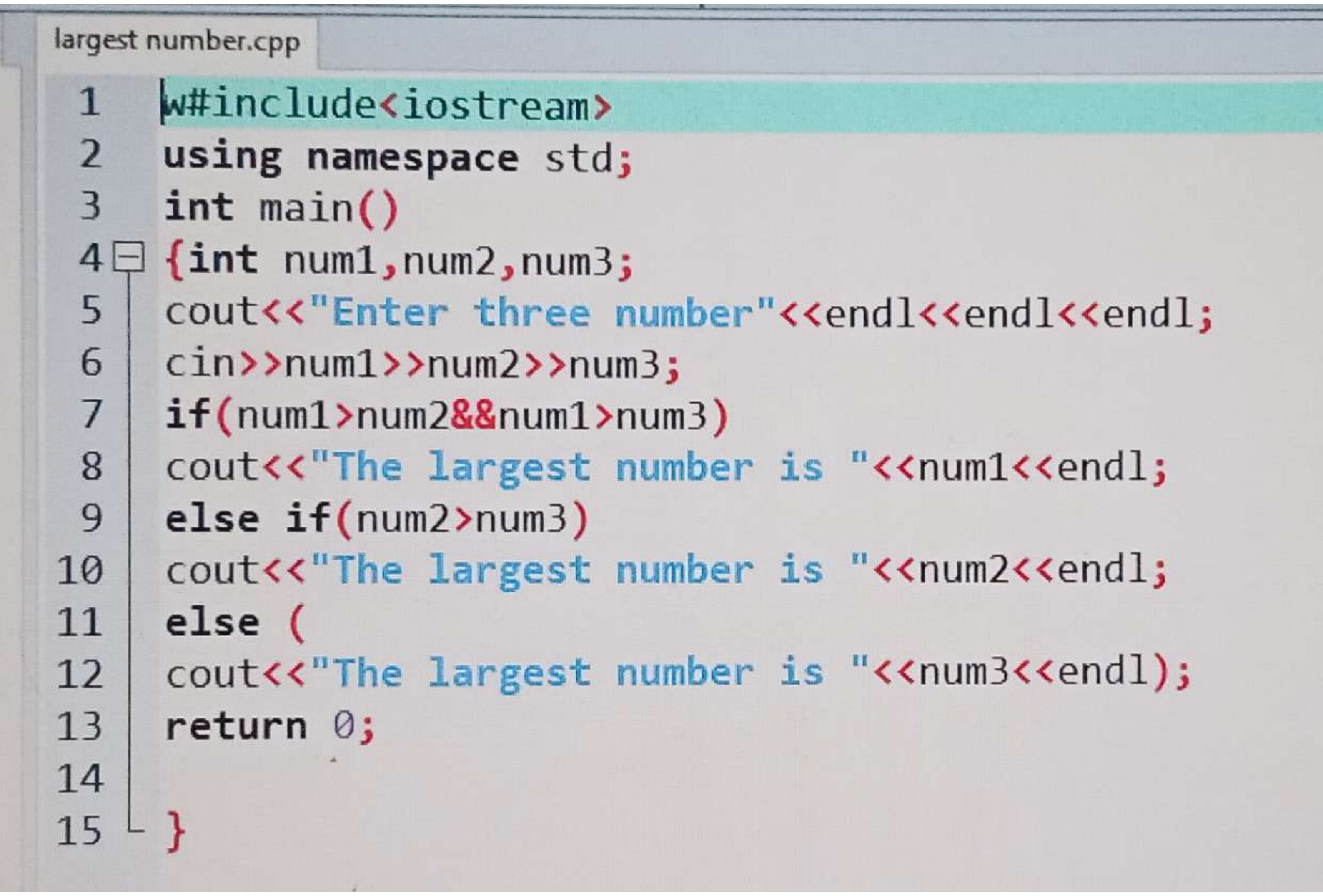
```cpp
#include<iostream>
using namespace std;
int main()
{int a,b,c;
cout<<"Enter the number:"<<endl;
cin>>a>>b;
cout<<"1.addition\n 2.subtraction\n 3.multiplication\n 4.division\n 5.modulus\n 6.exit\n enter your choice ";
cin>>c;
switch(c)
{case 1: cout<<"sum is"<<a+b<<endl; break;
case 2: cout<<"sub is"<<a-b<<endl; break;
case 3: cout<<"mul is"<<a*b<<endl; break;
case 4: cout<<"div is"<<a/b<<endl; break;
case 5: cout<<"modulus  is "<<a%b<<endl; break;
case 6: cout<<"Exit "<<endl;
}
return 0;
}
```



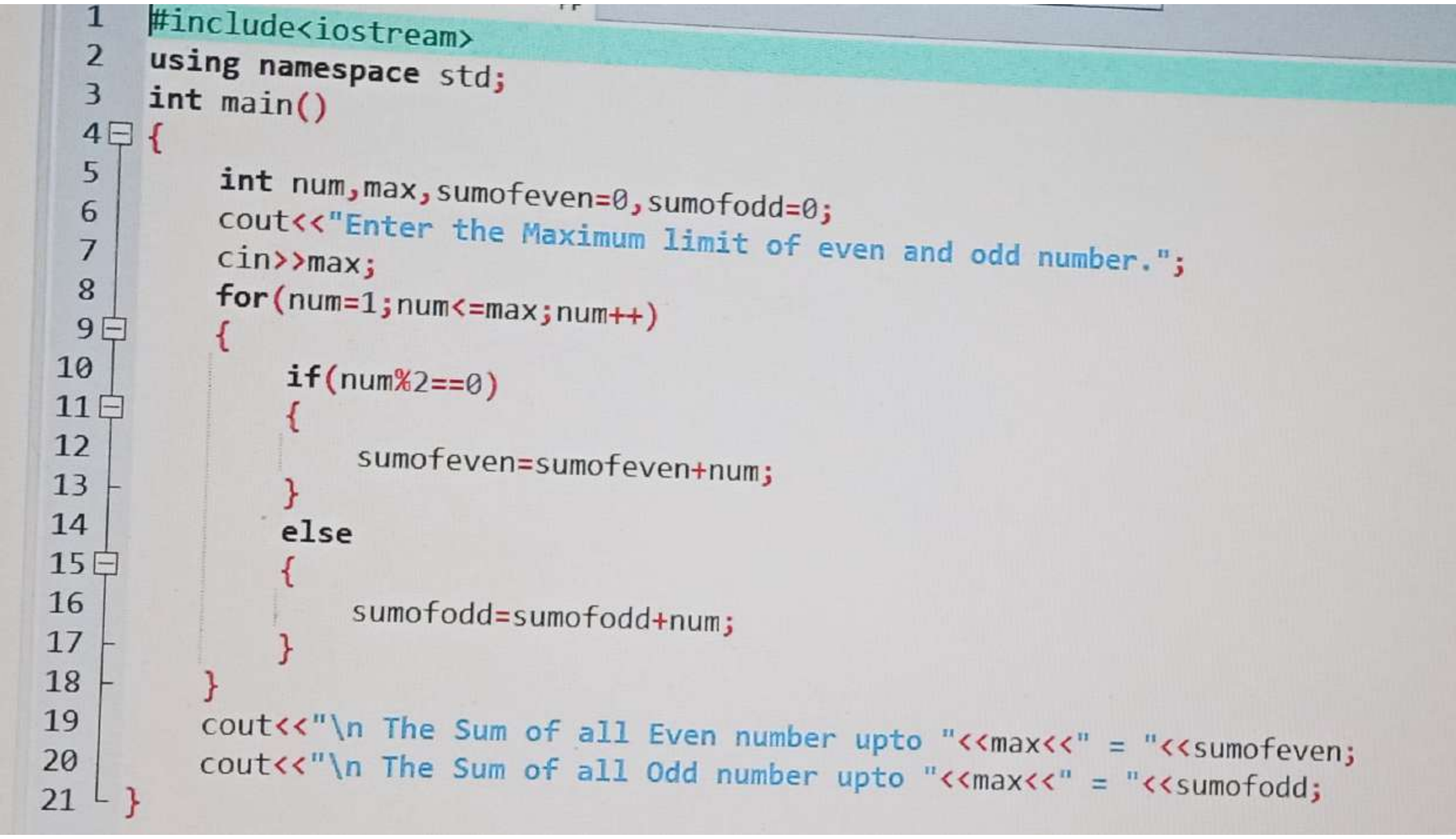
```cpp
#include<iostream>
using namespace std;
int main()
{
	int sec,min,hrs;
	cout<<"Enter seconds ";
	cin>>sec;
	min=sec/60;
	hrs=sec/3600;
	sec=sec;
	cout<<"\nMinutes is "<<min;
		cout<<"\nHours is "<<hrs;
			cout<<"\nSeconds  is "<<sec;
}
```



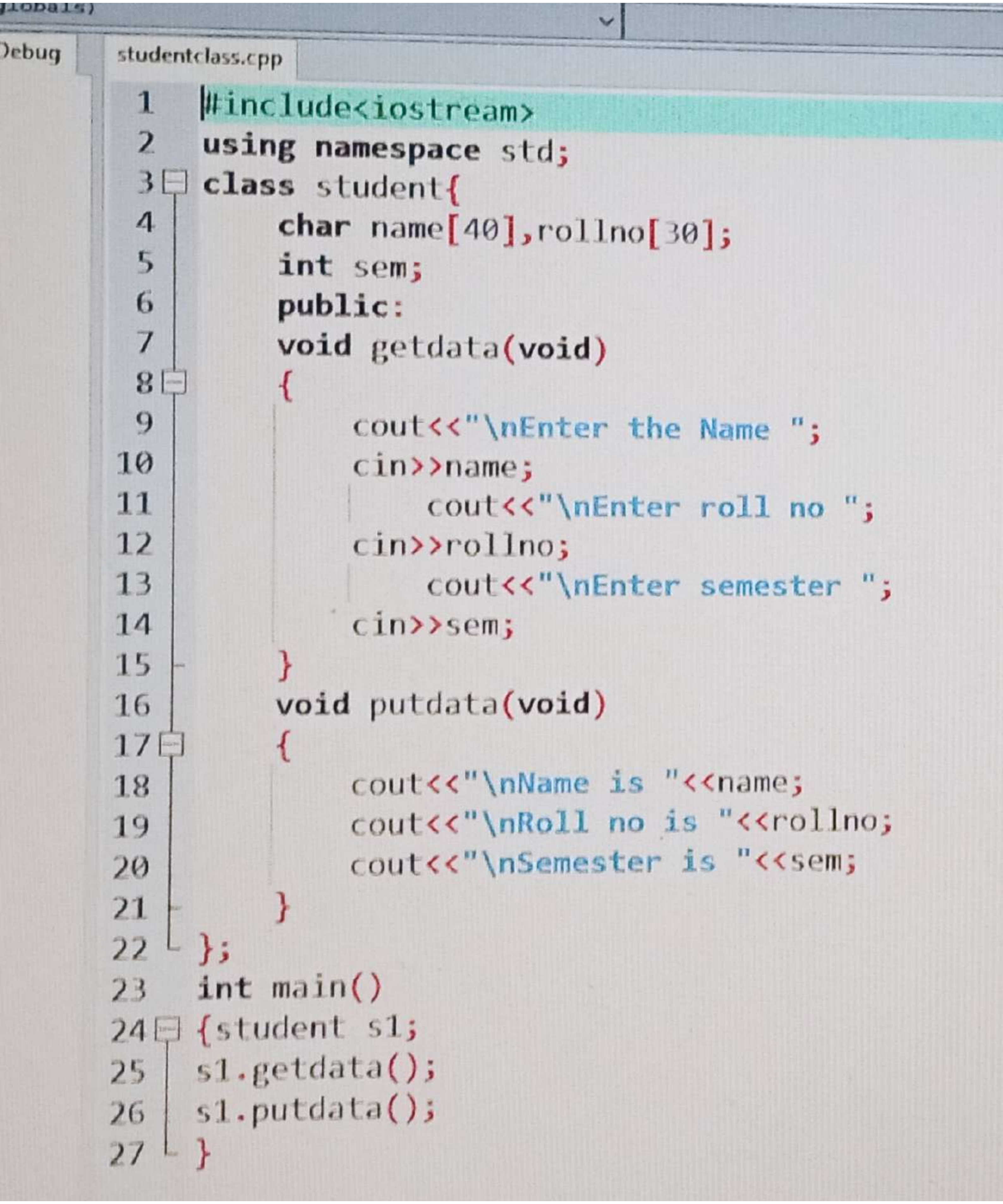
```cpp
#include<iostream>
using namespace std;
int main()
{
    const float pi=3.14;
    float r,h,b,l,s,vol;
    cout<<"for cone\n";
    cout<<"Enter Radius and Height :";
    cin>>r>>h;
    vol=(1.0/3.0)*pi*(r*r)*h;
    cout<<"Volume of Cone : "<<vol;

        cout<<"\nfor rectangle\n";
    cout<<"Enter length,breadth and Height :";
    cin>>l>>b>>h;
    vol=2*(l+b+h);
    cout<<"\nVolume of Rectangle : "<<vol;


        cout<<"\nfor square\n";
    cout<<"Enter sides:";
    cin>>s;
    vol=(s*s*s);
    cout<<"Volume of Square : "<<vol;
}
```



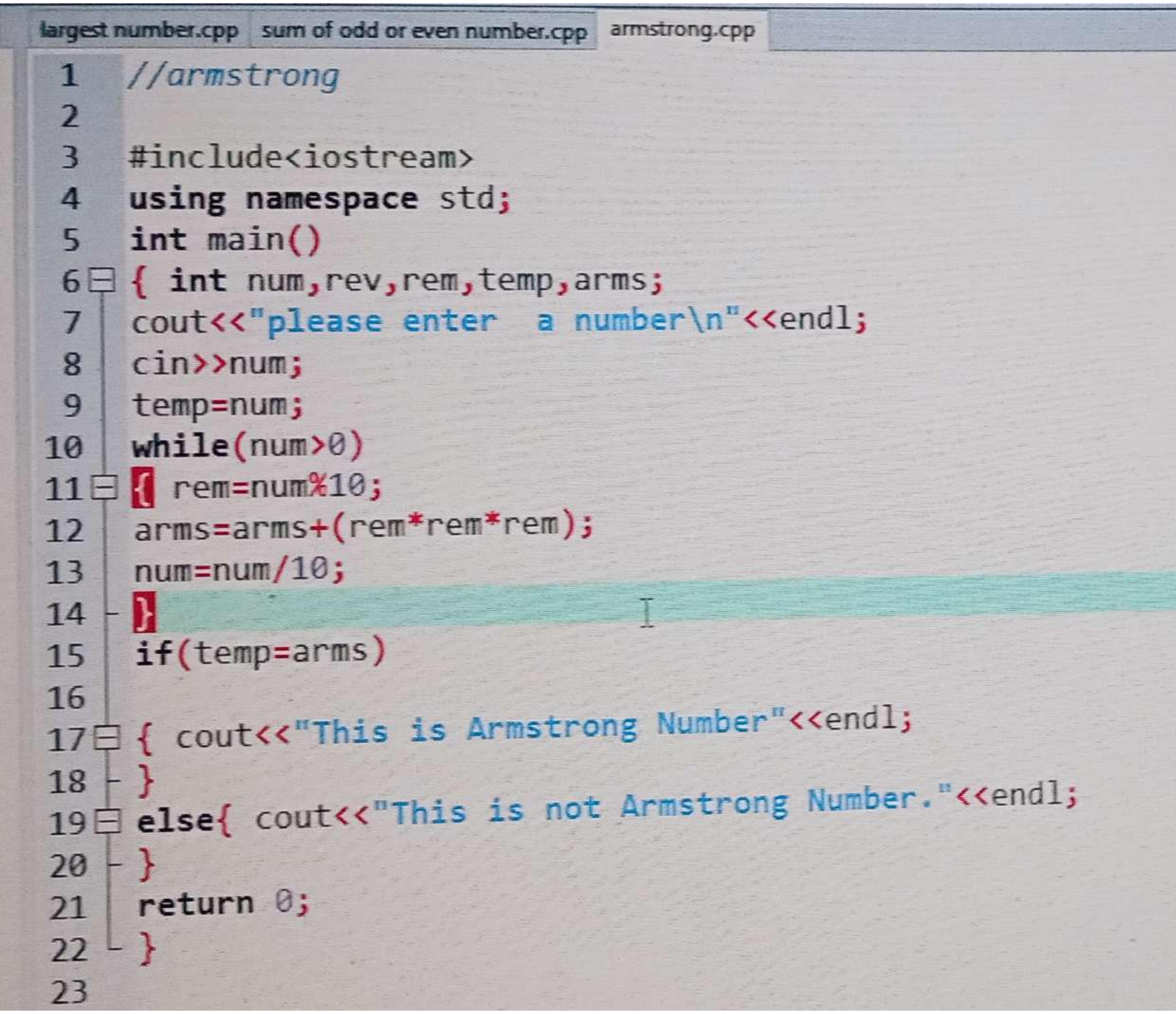
```
w#include<iostream>
using namespace std;
int main()
{int num1,num2,num3;
cout<<"Enter three number"<<endl<<endl<<endl;
cin>>num1>>num2>>num3;
if(num1>num2&&num1>num3)
cout<<"The largest number is "<<num1<<endl;
else if(num2>num3)
cout<<"The largest number is "<<num2<<endl;
else (
cout<<"The largest number is "<<num3<<endl);
return 0;

}
```



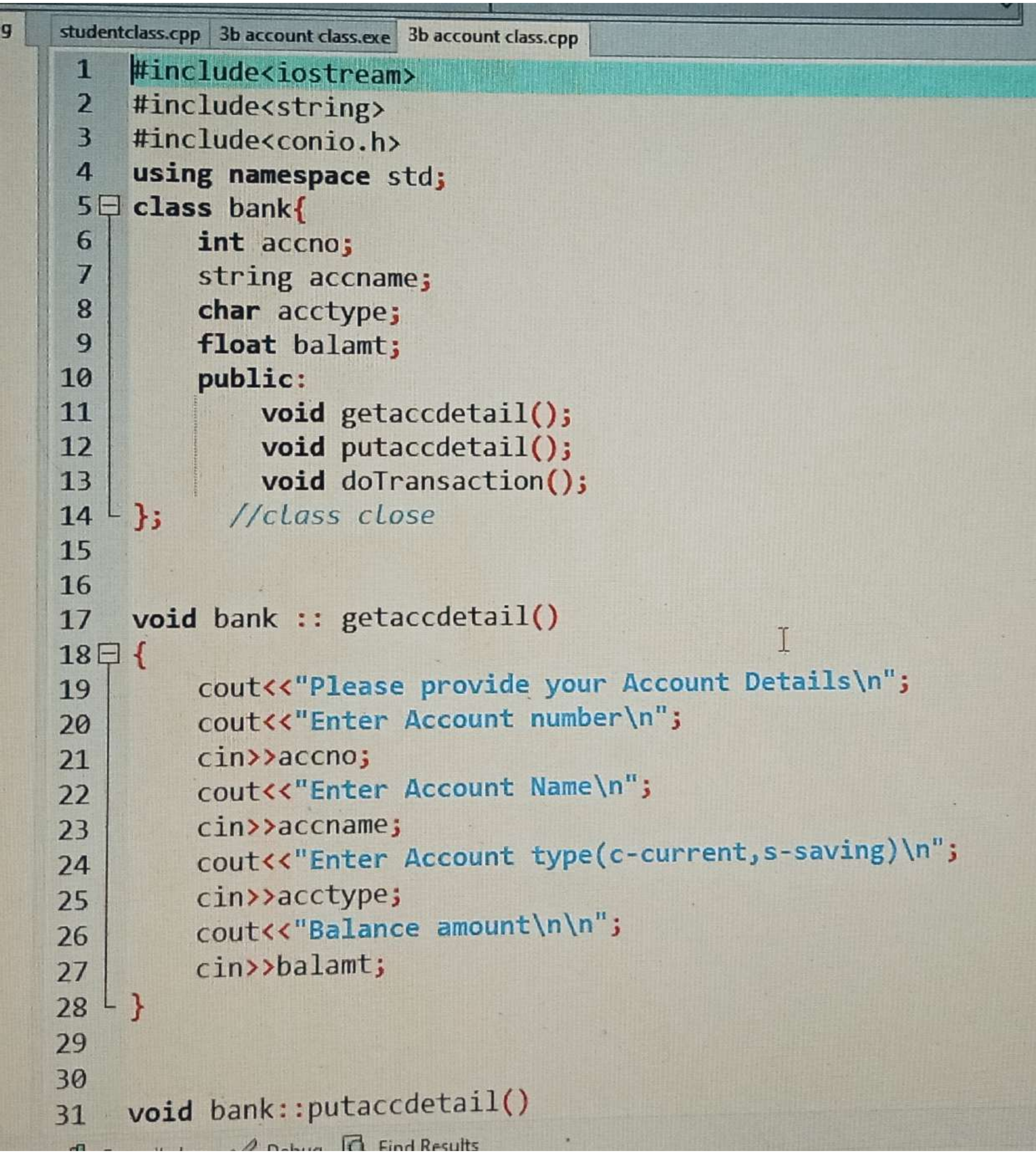
```cpp
#include<iostream>
using namespace std;
int main()
{
    int num,max,sumofeven=0,sumofodd=0;
    cout<<"Enter the Maximum limit of even and odd number.";
    cin>>max;
    for(num=1;num<=max;num++)
    {
        if(num%2==0)
        {
            sumofeven=sumofeven+num;
        }
        else
        {
            sumofodd=sumofodd+num;
        }
    }
   cout<<"\n The Sum of all Even number upto "<<max<<" = "<<sumofeven;
   cout<<"\n The Sum of all Odd number upto "<<max<<" = "<<sumofodd;
}
```



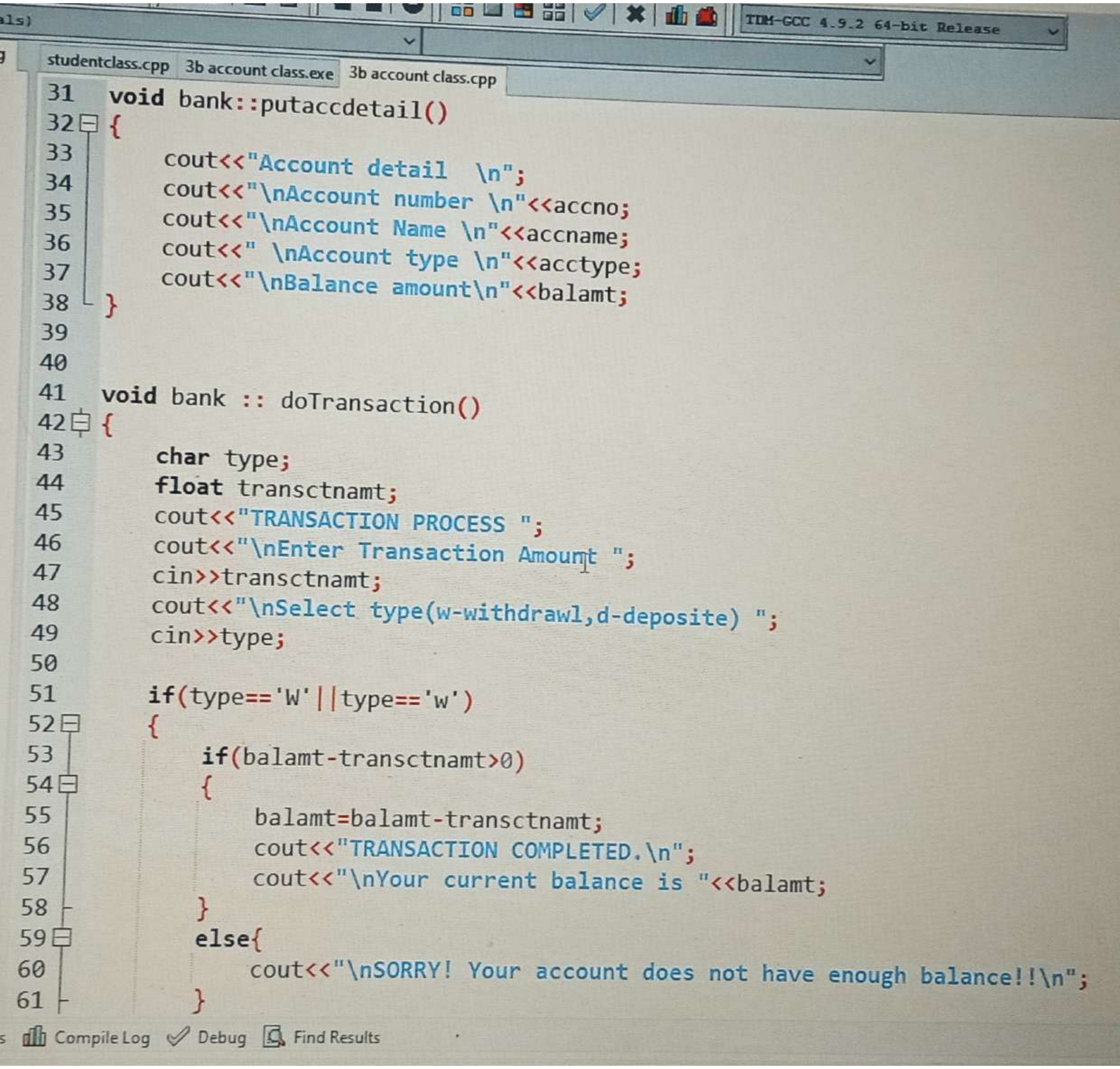
```
#include<iostream>
using namespace std;
class student{
	char name[40],rollno[30];
	int sem;
	public:
	void getdata(void)
	{
		cout<<"\nEnter the Name ";
		cin>>name;
			cout<<"\nEnter roll no ";
		cin>>rollno;
			cout<<"\nEnter semester ";
		cin>>sem;
	}
	void putdata(void)
	{
		cout<<"\nName is "<<name;
		cout<<"\nRoll no is "<<rollno;
		cout<<"\nSemester is "<<sem;
	}
};
int main()
{student s1;
s1.getdata();
s1.putdata();
}
```



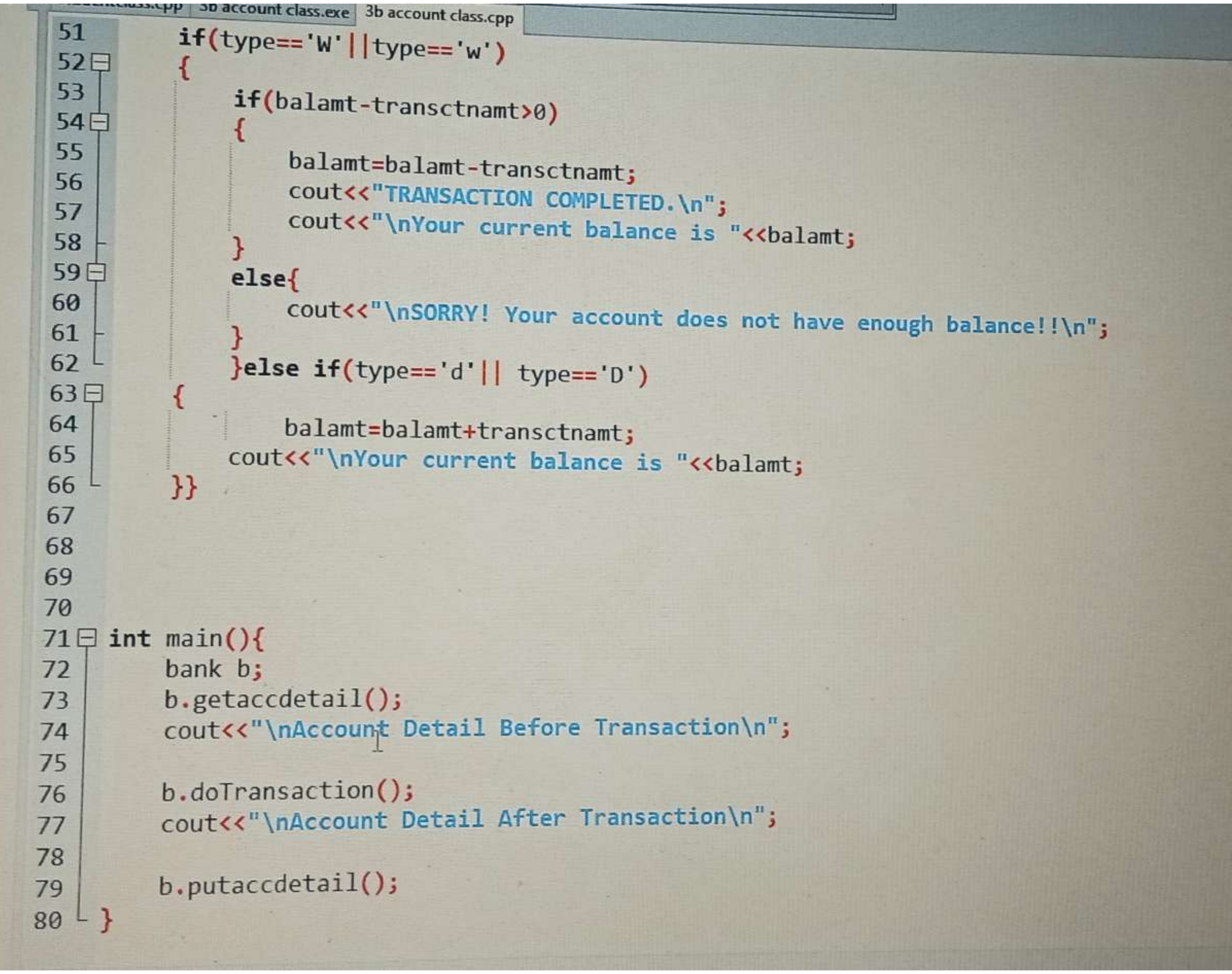
```cpp
//armstrong

#include<iostream>
using namespace std;
int main()
{ int num,rev,rem,temp,arms;
cout<<"please enter  a number\n"<<endl;
cin>>num;
temp=num;
while(num>0)
{ rem=num%10;
arms=arms+(rem*rem*rem);
num=num/10;
}
if(temp=arms)

{ cout<<"This is Armstrong Number"<<endl;
}
else{ cout<<"This is not Armstrong Number."<<endl;
}
return 0;
}

```



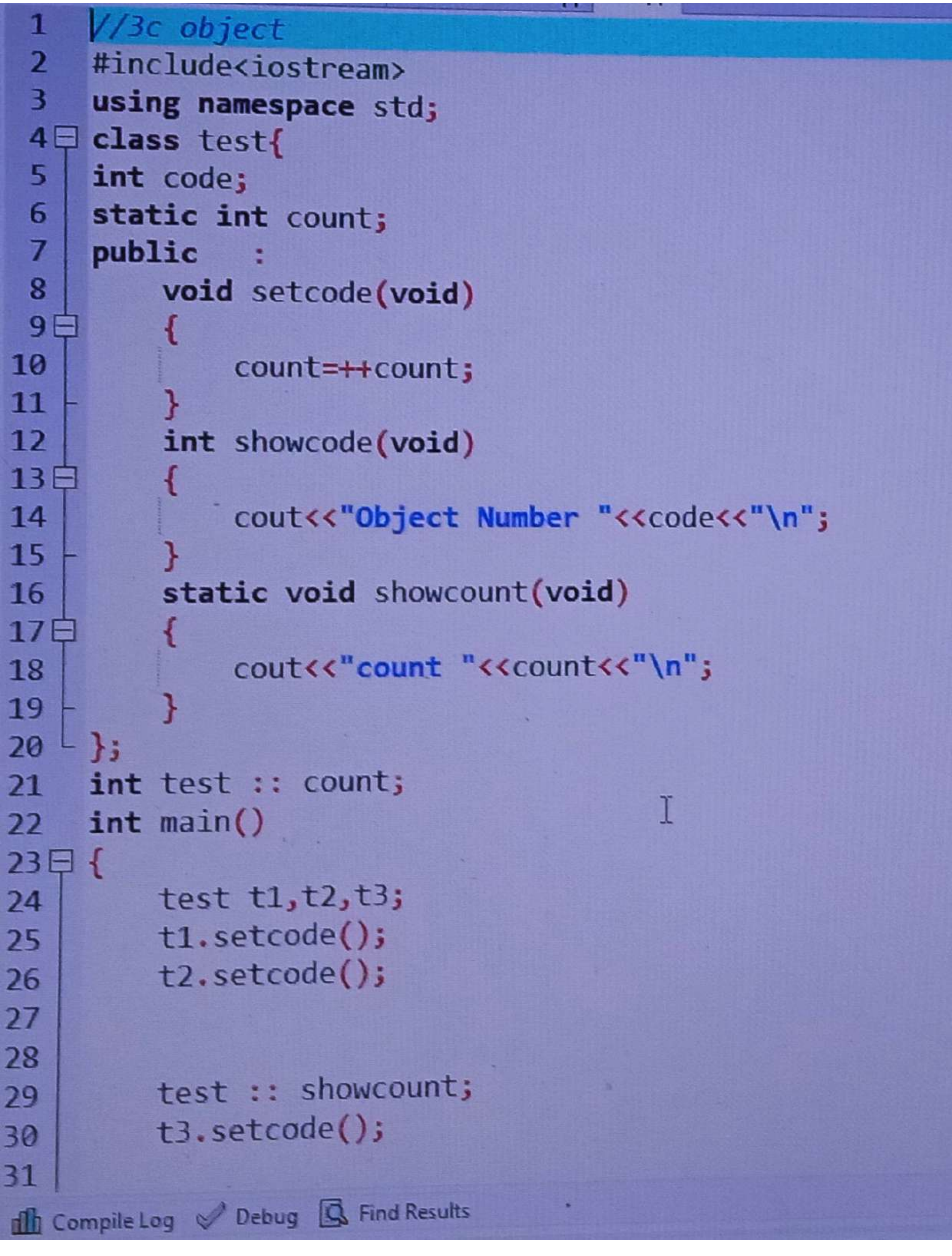
```
#include<iostream>
#include<string>
#include<conio.h>
using namespace std;
class bank{
    int accno;
    string accname;
    char acctype;
    float balamt;
    public:
        void getaccdetail();
        void putaccdetail();
        void doTransaction();
};    //class close


void bank :: getaccdetail()
{
    cout<<"Please provide your Account Details\n";
    cout<<"Enter Account number\n";
    cin>>accno;
    cout<<"Enter Account Name\n";
    cin>>accname;
    cout<<"Enter Account type(c-current,s-saving)\n";
    cin>>acctype;
    cout<<"Balance amount\n\n";
    cin>>balamt;
}


void bank::putaccdetail()
```



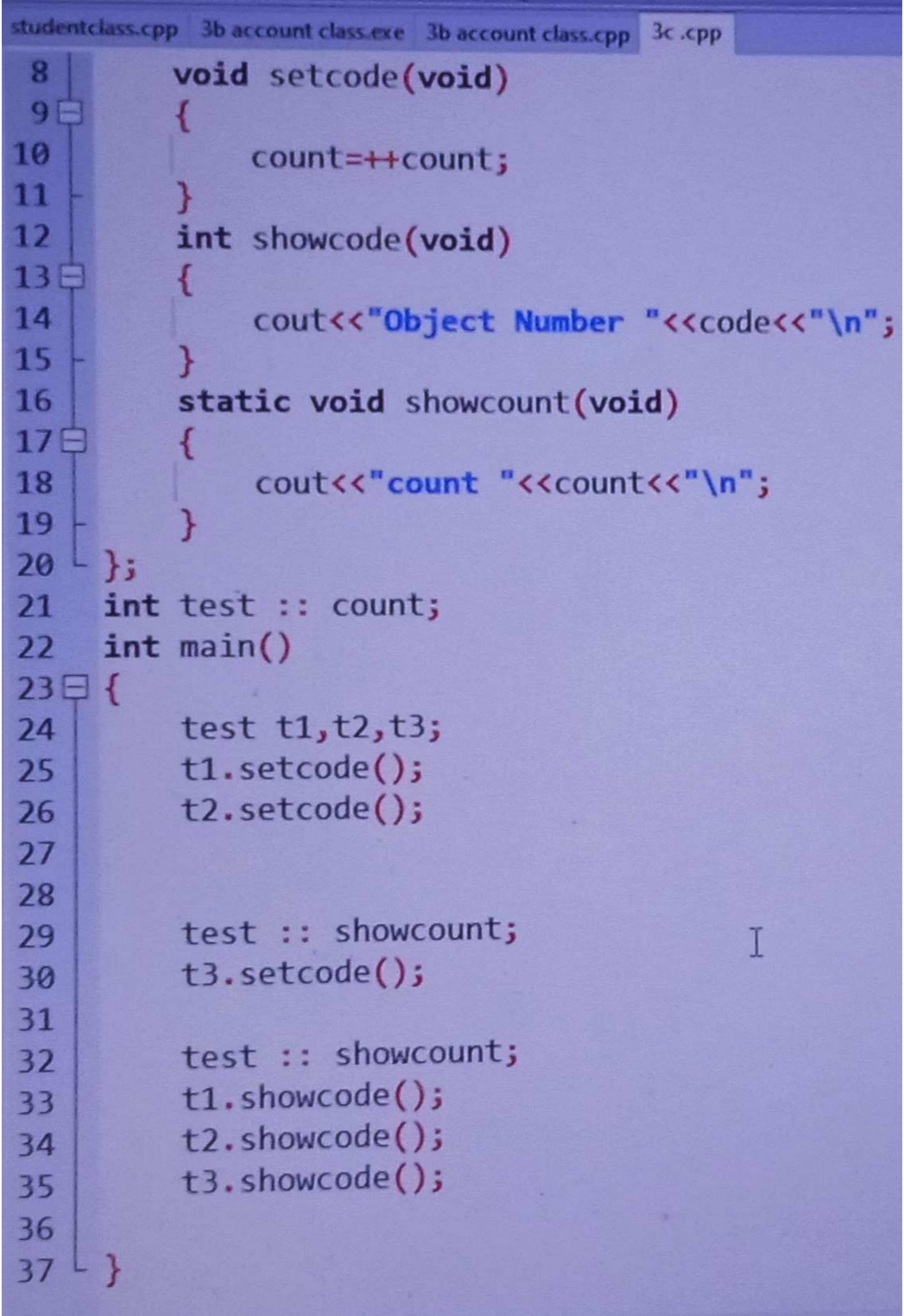
```cpp
void bank::putaccdetail()
{
	cout<<"Account detail  \n";
	cout<<"\nAccount number \n"<<accno;
	cout<<"\nAccount Name \n"<<accname;
	cout<<" \nAccount type \n"<<acctype;
	cout<<"\nBalance amount\n"<<balamt;
}


void bank :: doTransaction()
{
	char type;
	float transctnamt;
	cout<<"TRANSACTION PROCESS ";
	cout<<"\nEnter Transaction Amount ";
	cin>>transctnamt;
	cout<<"\nSelect type(w-withdrawl,d-deposite) ";
	cin>>type;

	if(type=='W'||type=='w')
	{
		if(balamt-transctnamt>0)
		{
			balamt=balamt-transctnamt;
			cout<<"TRANSACTION COMPLETED.\n";
			cout<<"\nYour current balance is "<<balamt;
		}
		else{
			cout<<"\nSORRY! Your account does not have enough balance!!\n";
		}
```



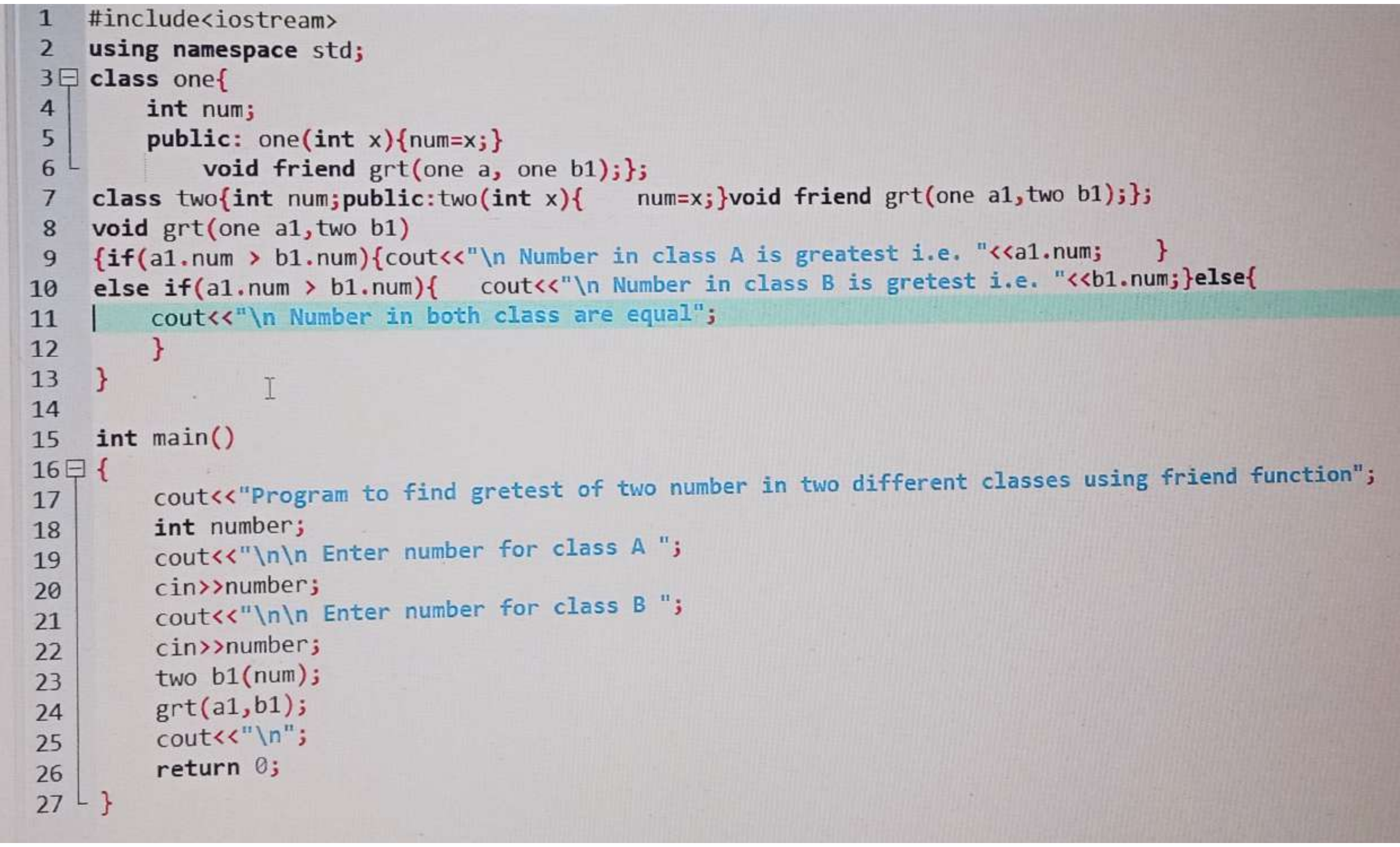
```
        if(type=='W'||type=='w')
        {
            if(balamt-transctnamt>0)
            {
                balamt=balamt-transctnamt;
                cout<<"TRANSACTION COMPLETED.\n";
                cout<<"\nYour current balance is "<<balamt;
            }
            else{
                cout<<"\nSORRY! Your account does not have enough balance!!\n";
            }
            }else if(type=='d'|| type=='D')
        {
                balamt=balamt+transctnamt;
            cout<<"\nYour current balance is "<<balamt;
        }}



int main(){
    bank b;
    b.getaccdetail();
    cout<<"\nAccount Detail Before Transaction\n";

    b.doTransaction();
    cout<<"\nAccount Detail After Transaction\n";

    b.putaccdetail();
}
```



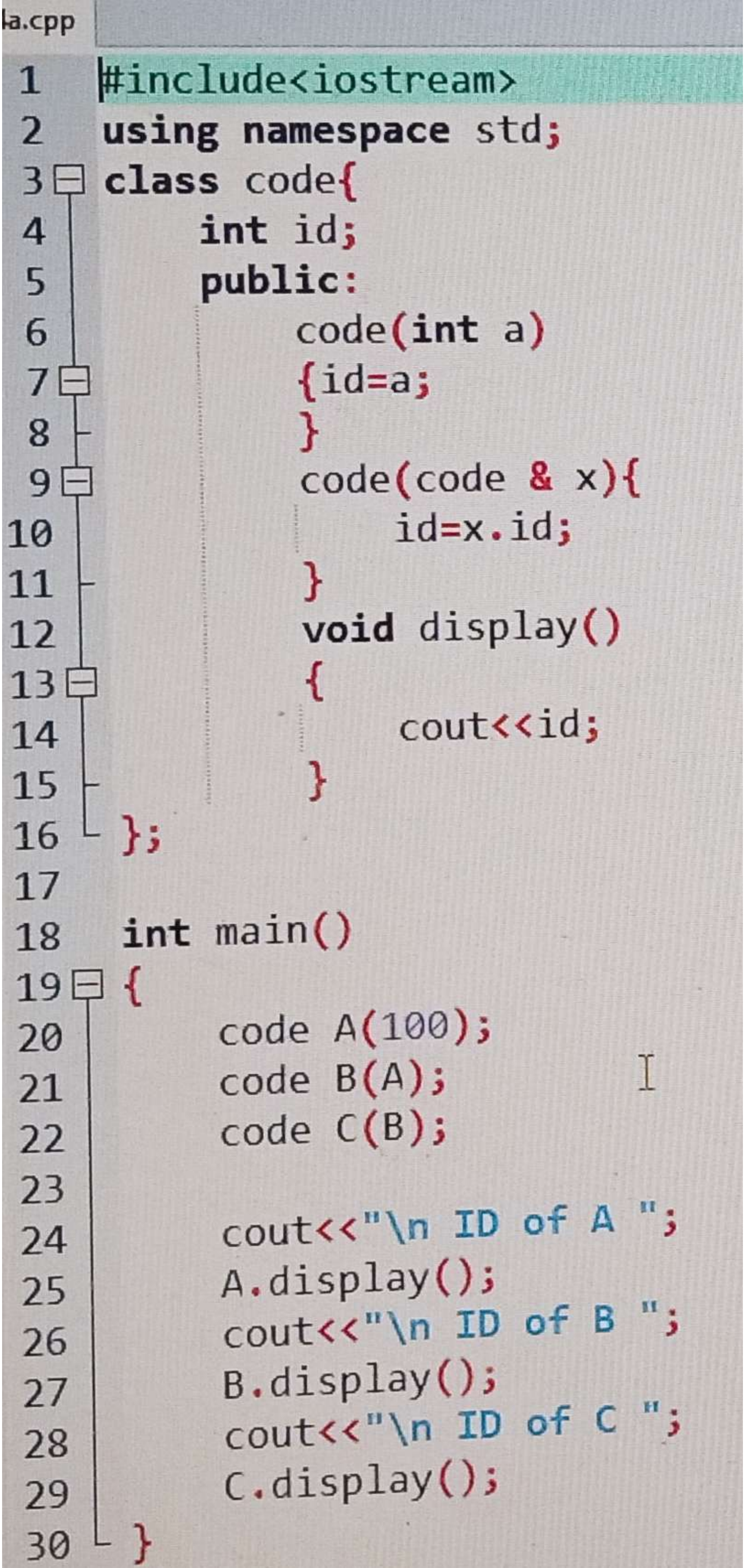
```
//3c object
#include<iostream>
using namespace std;
class test{
int code;
static int count;
public   :
     void setcode(void)
     {
         count=++count;
     }
     int showcode(void)
     {
         cout<<"Object Number "<<code<<"\n";
     }
     static void showcount(void)
     {
         cout<<"count "<<count<<"\n";
     }
};
int test :: count;
int main()
{
     test t1,t2,t3;
     t1.setcode();
     t2.setcode();


     test :: showcount;
     t3.setcode();

```



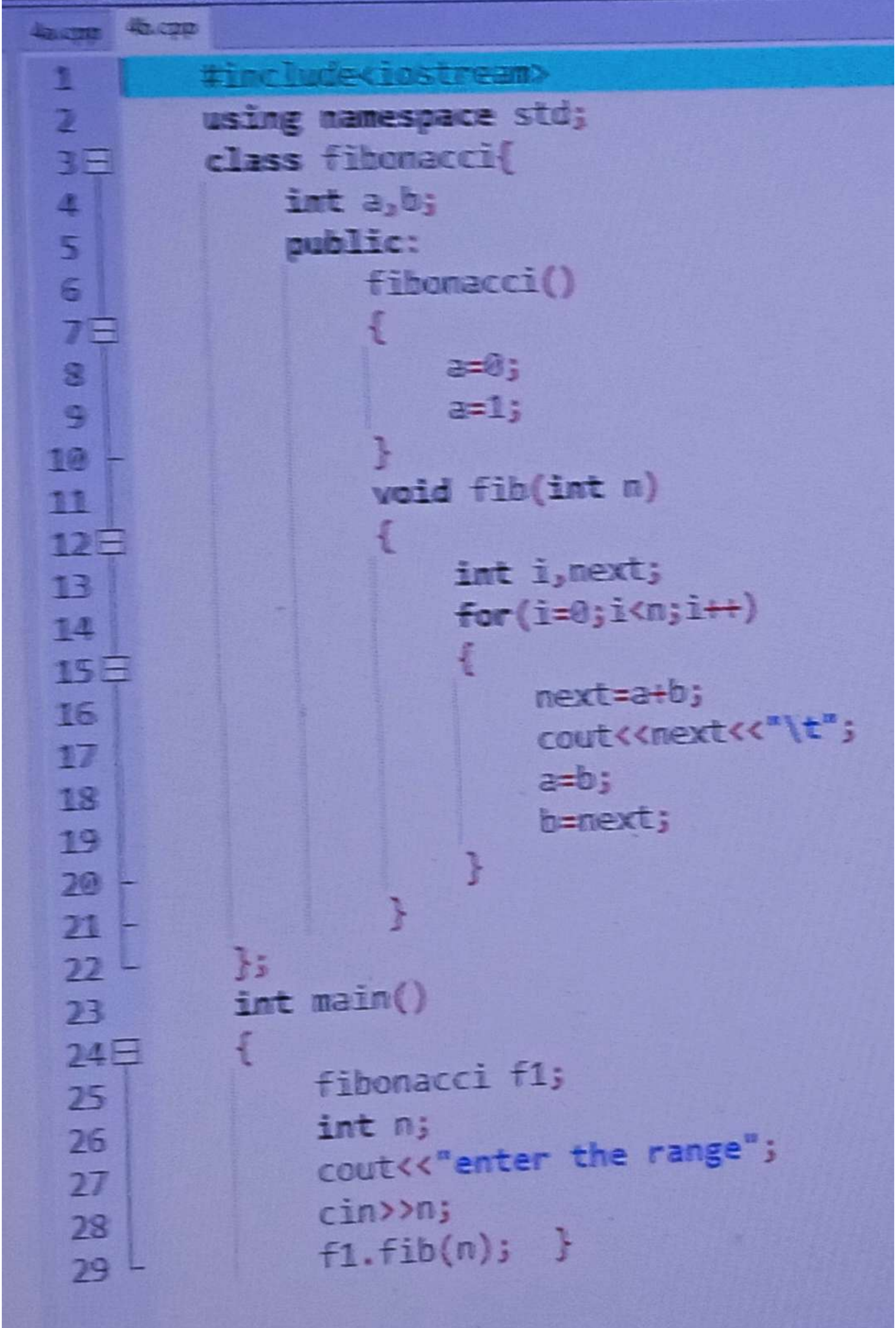
```
    void setcode(void)
    {
        count=++count;
    }
    int showcode(void)
    {
        cout<<"Object Number "<<code<<"\n";
    }
    static void showcount(void)
    {
        cout<<"count "<<count<<"\n";
    }
};
int test :: count;
int main()
{
    test t1,t2,t3;
    t1.setcode();
    t2.setcode();


    test :: showcount;
    t3.setcode();

    test :: showcount;
    t1.showcode();
    t2.showcode();
    t3.showcode();

}
```



```cpp
#include<iostream>
using namespace std;
class one{
	int num;
	public: one(int x){num=x;}
		void friend grt(one a, one b1);};
class two{int num;public:two(int x){	num=x;}void friend grt(one a1,two b1);};
void grt(one a1,two b1)
{if(a1.num > b1.num){cout<<"\n Number in class A is greatest i.e. "<<a1.num;	}
else if(a1.num > b1.num){	cout<<"\n Number in class B is gretest i.e. "<<b1.num;}else{
	cout<<"\n Number in both class are equal";
	}
}

int main()
{
	cout<<"Program to find gretest of two number in two different classes using friend function";
	int number;
	cout<<"\n\n Enter number for class A ";
	cin>>number;
	cout<<"\n\n Enter number for class B ";
	cin>>number;
	two b1(num);
	grt(a1,b1);
	cout<<"\n";
	return 0;
}
```



```
#include<iostream>
using namespace std;
class code{
    int id;
    public:
        code(int a)
        {id=a;
        }
        code(code & x){
            id=x.id;
        }
        void display()
        {
            cout<<id;
        }
};

int main()
{
    code A(100);
    code B(A);
    code C(B);

    cout<<"\n ID of A ";
    A.display();
    cout<<"\n ID of B ";
    B.display();
    cout<<"\n ID of C ";
    C.display();
}
```



```cpp
#include<iostream>
using namespace std;
class fibonacci{
    int a,b;
    public:
        fibonacci()
        {
            a=0;
            a=1;
        }
        void fib(int n)
        {
            int i,next;
            for(i=0;i<n;i++)
            {
                next=a+b;
                cout<<next<<"\t";
                a=b;
                b=next;
            }
        }
};
int main()
{
    fibonacci f1;
    int n;
    cout<<"enter the range";
    cin>>n;
    f1.fib(n);  }
```

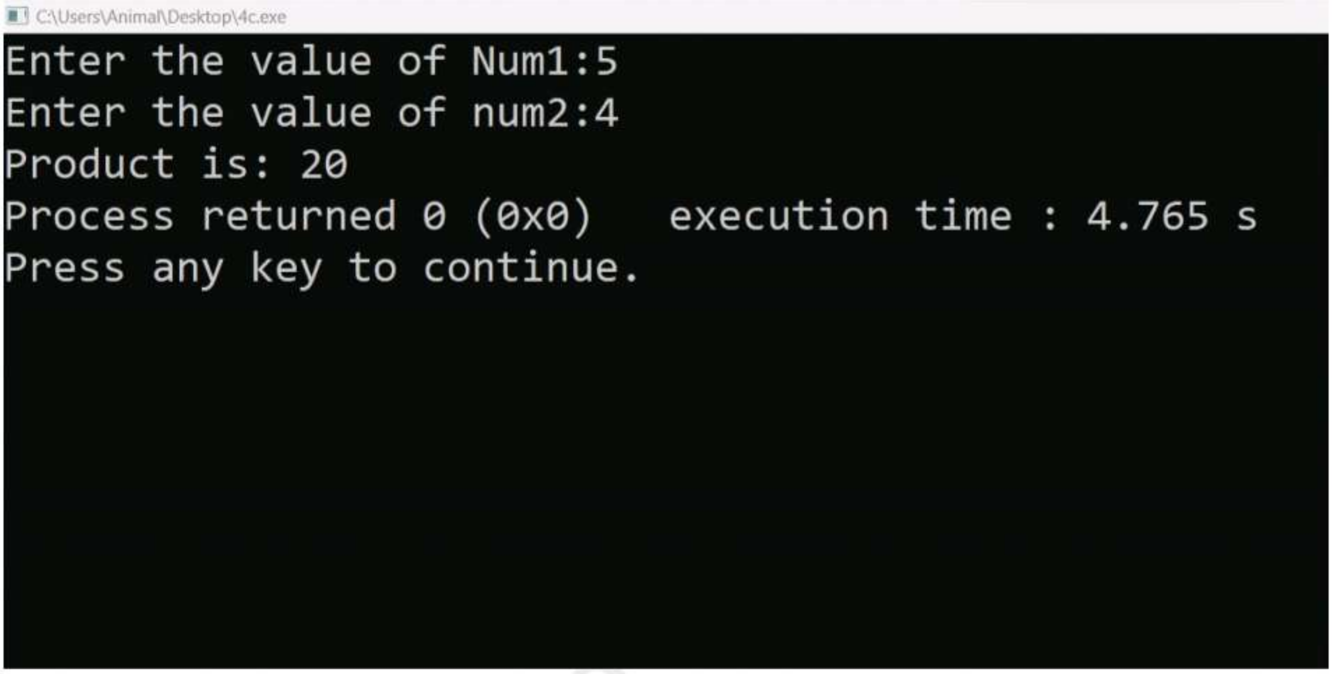
PRACTICAL 5A

AIM:- Write a C++ Program that illustrates single inheritance.

CODE:-

```
#include <iostream>
using namespace std;
class base
{
   public:
   int num1;
   void readnum1()
   {
      cout<<"Enter the value of Num1:";
      cin>> num1;
   }
};
class derive: public base
{
public:
   int num2;
   void readnum2()
   {
      cout<<"Enter the value of num2:";
      cin>> num2;
   }
   void product()
   {
      cout<<"Product is: "<<num1*num2;
   }
};

int main()
{
   derive d1;
   d1.readnum1();
   d1.readnum2();
   d1.product();
}
```

OUTPUT:-



C:\Users\Animal\Desktop\4c.exe

```
Enter the value of Num1:5
Enter the value of num2:4
Product is: 20
Process returned 0 (0x0)   execution time : 4.765 s
Press any key to continue.
```

PRACTICAL 5B

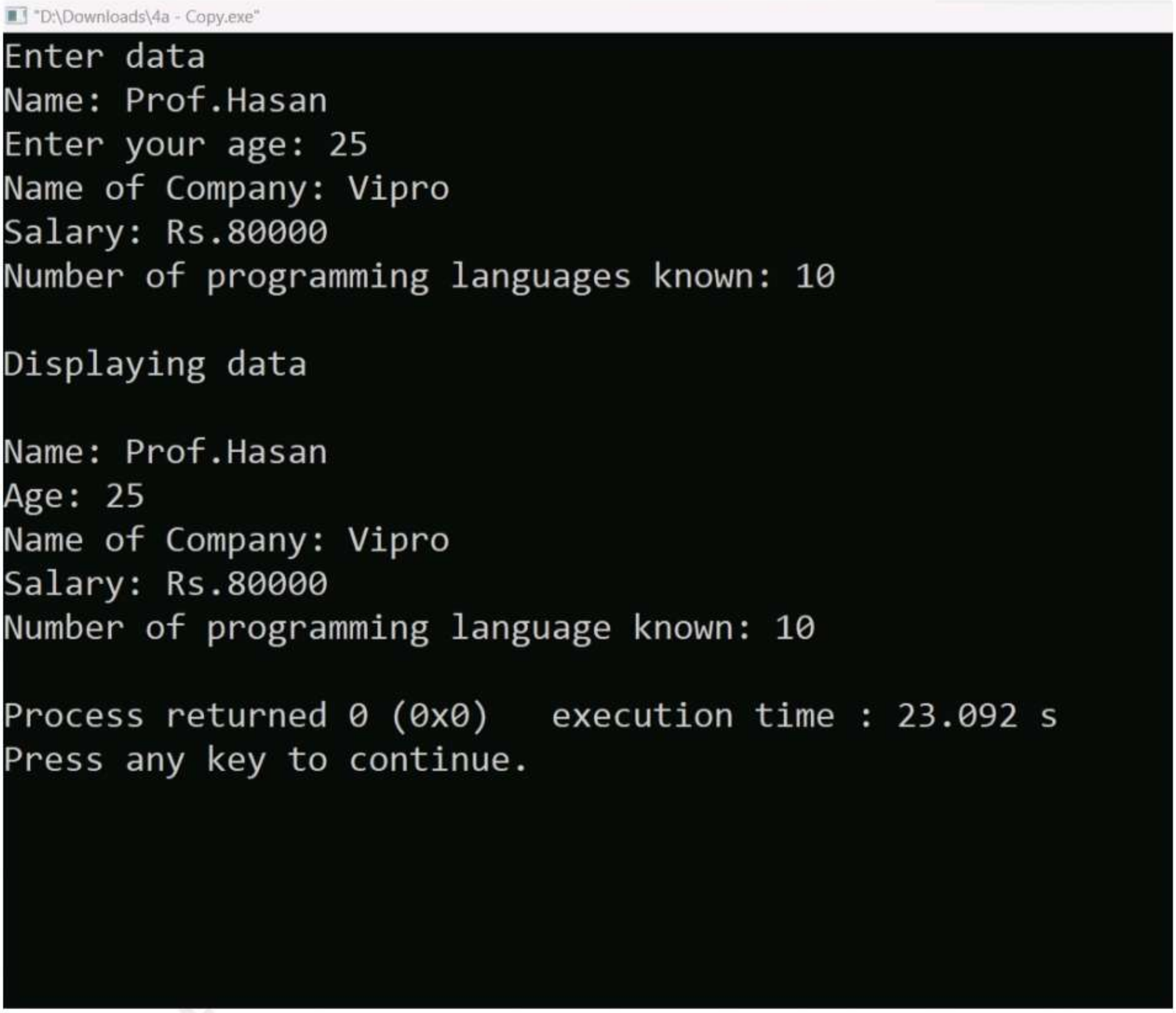
AIM:- Write a C++ Program that illustrates Multilevel inheritance.

CODE:-

```
#include<iostream>
using namespace std;
class person
{
public:
char name[50];
int age;
public:
void persondata()
{
cout<<"Name: ";
cin>>name;
cout<<"Enter your age: ";
cin>>age;
}
};
class employee: public person
{
public:
char company[100];
float salary;
public:
void empinfo()
{
cout<<"Name of Company: ";
cin>>company;
cout<<"Salary: Rs.";
cin>>salary;
}

//Topic: Inheritance
};
class programmer: public employee
{
public:
```

```
int number;

public:
void programdata()
{
cout<<"Number of programming languages known: ";
cin>>number;
}
void display()
{
cout<<"Name: "<<name<<endl;
cout<<"Age: "<<age<<endl;
cout<<"Name of Company: "<<company<<endl;
cout<<"Salary: Rs."<<salary<<endl;
cout<<"Number of programming language known: "<<number<<endl;
}
};
int main()
{
programmer p;
cout<<"Enter data"<<endl;
p.persondata();
p.empinfo();
p.programdata();
cout<<endl<<"Displaying data\n"<<endl;
p.display();
return 0;
}
```

OUTPUT:-



```
Enter data
Name: Prof.Hasan
Enter your age: 25
Name of Company: Vipro
Salary: Rs.80000
Number of programming languages known: 10

Displaying data

Name: Prof.Hasan
Age: 25
Name of Company: Vipro
Salary: Rs.80000
Number of programming language known: 10

Process returned 0 (0x0)   execution time : 23.092 s
Press any key to continue.
```

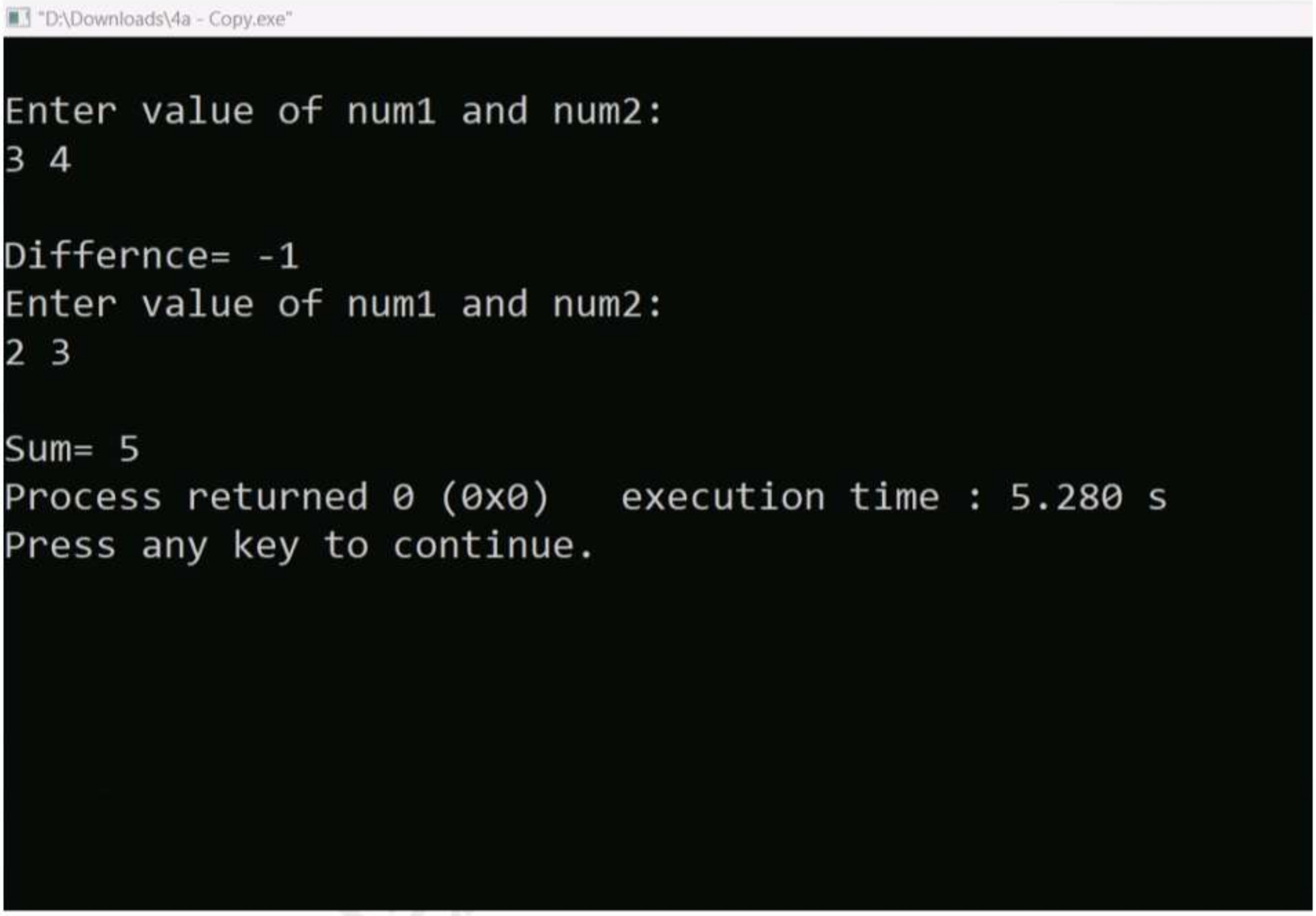
PRACTICAL 5C

AIM:- Write a C++ Program that illustrates Hierarchical inheritance.

CODE:-

```
#include <iostream>
using namespace std;
class number1
{
   public:
        int num1, num2;
        void getdata()
        {
           cout << "\nEnter value of num1 and num2:\n";
        cin >> num1 >> num2;
        }
};
class number2 : public number1
{
   public:
        void differnce()
        {
           cout << "\nDiffernce= " << num1 - num2;
        }
};
class number3 : public number1
{
   public:
        void sum()
        {
      cout << "\nSum= " << num1 + num2;
        }
};
int main()
{
   number2 obj1;
   number3 obj2;
   obj1.getdata();
   obj1.differnce();
   obj2.getdata();
   obj2.sum();
   return 0;
}
```

OUTPUT:-



```
Enter value of num1 and num2:
3 4

Differnce= -1
Enter value of num1 and num2:
2 3

Sum= 5
Process returned 0 (0x0)   execution time : 5.280 s
Press any key to continue.
```

PRACTICAL 5D

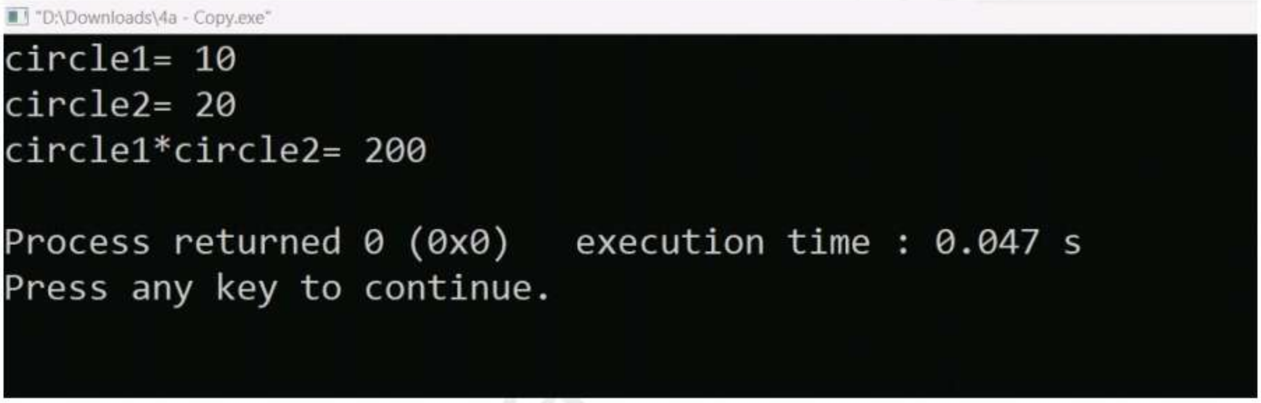
AIM:- Write a C++ Program that illustrates Multiple inheritance.

CODE:-

```
#include<iostream>
using namespace std;
class shape1
{
  protected:
  int circle1;
  public:
    void get_circle1(int);
};
class shape2
{
   protected:
  int circle2;
  public:
    void get_circle2(int);
};
class total : public shape1 , public shape2
{
  public:
  void display(void);
};
void shape1::get_circle1(int x)
{
  circle1=x;
}
void shape2::get_circle2(int y)
{
  circle2=y;
}
void total::display(void)
{
  cout<<"circle1= "<<circle1<<"\n";
  cout<<"circle2= "<<circle2<<"\n";
  cout<<"circle1*circle2= "<<circle1*circle2<<"\n";
}
```

```
int main()
{
   total p;
   p.get_circle1(10);
   p.get_circle2(20);
   p.display();
   return 0;
}
```

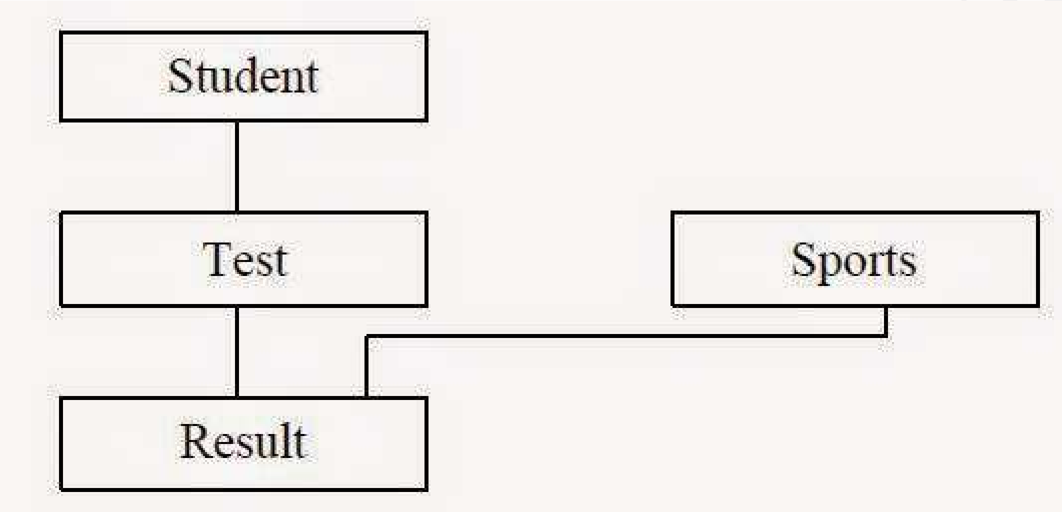
OUTPUT:-



```
circle1= 10
circle2= 20
circle1*circle2= 200

Process returned 0 (0x0)   execution time : 0.047 s
Press any key to continue.
```

PRACTICAL 5E

AIM:- Write a C++ Program to design a student class representing student roll no. and a test class (derived class of student) representing the scores of the student in various subjects and sports class representing the score in sports. The sports and test class should be inherited by a result class having the functionality to add the scores and display the final result for a student.



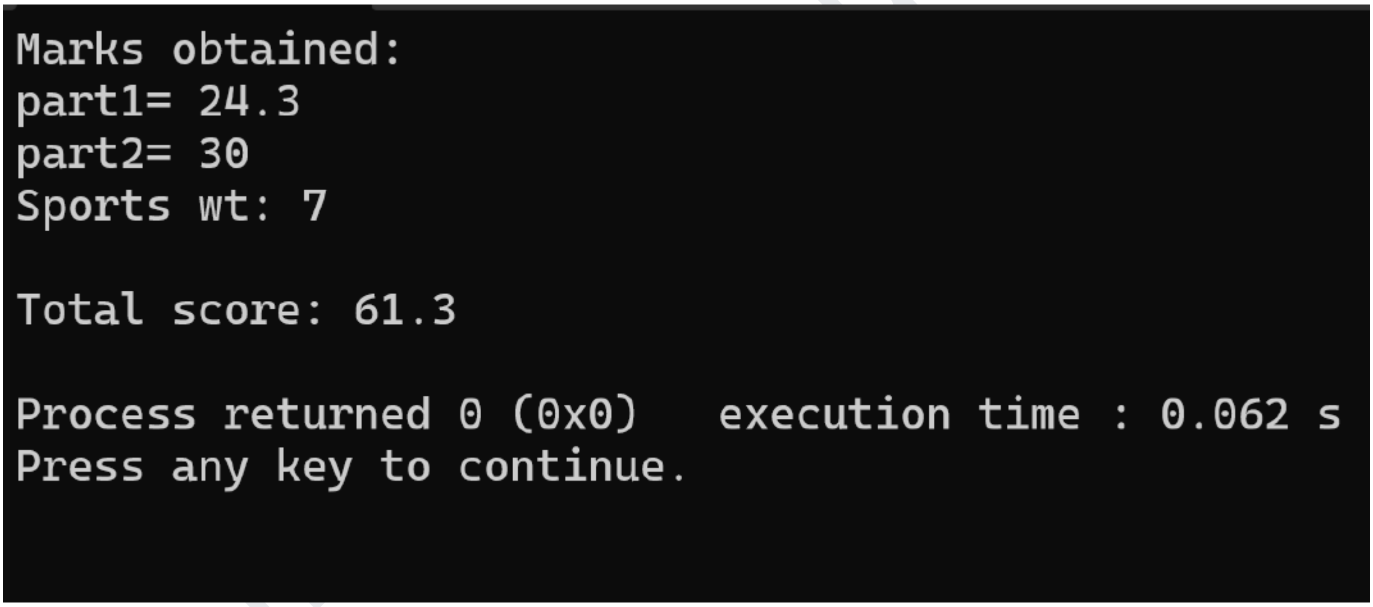
CODE:-

```
#include<iostream>
using namespace std;
class student
{
   protected:
       int roll_number;
   public:
       void get_number(int a)
       {
           roll_number = a;
       }
       void put_number(void)
       {
           cout<<"Roll no= "<<roll_number<<"\n";
       }
};
class test : public student
```

```
{
protected:
    float part1, part2;
public:
    void get_marks(float x, float y)
    {
        part1 = x; part2 = y;
    }
    void put_marks(void)
    {
        cout<<"Marks obtained:"<<"\n"
        <<"part1= "<<part1<<"\n"
        <<"part2= "<<part2<<"\n";
    }
};
class sports
{
protected:
    float score;
public:
    void get_score(float s)
    {
        score=s;
    }
    void put_score(void)
    {
        cout<<"Sports wt: "<<score<<"\n\n";
    }
};
class result:public test, public sports
{
    float total;
public:
    void display(void);
};
void result :: display(void)
{
    total=part1+part2+score;
    put_number();
    put_marks();
    put_score();
```

```
 cout<<"Total score: "<<total<<"\n";
}
int main()
{
   result student_1;
   student_1.get_number(1234);
   student_1.get_marks(24.3,30.0);
   student_1.get_score(7.0);
   student_1.display();
   return 0;
}
```

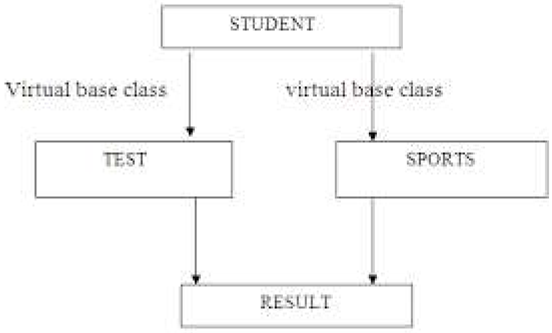
OUTPUT:-



```
Marks obtained:
part1= 24.3
part2= 30
Sports wt: 7

Total score: 61.3

Process returned 0 (0x0)   execution time : 0.062 s
Press any key to continue.
```

PRACTICAL 6

AIM:- Write a C++ Program to implement the concept of virtual base class.

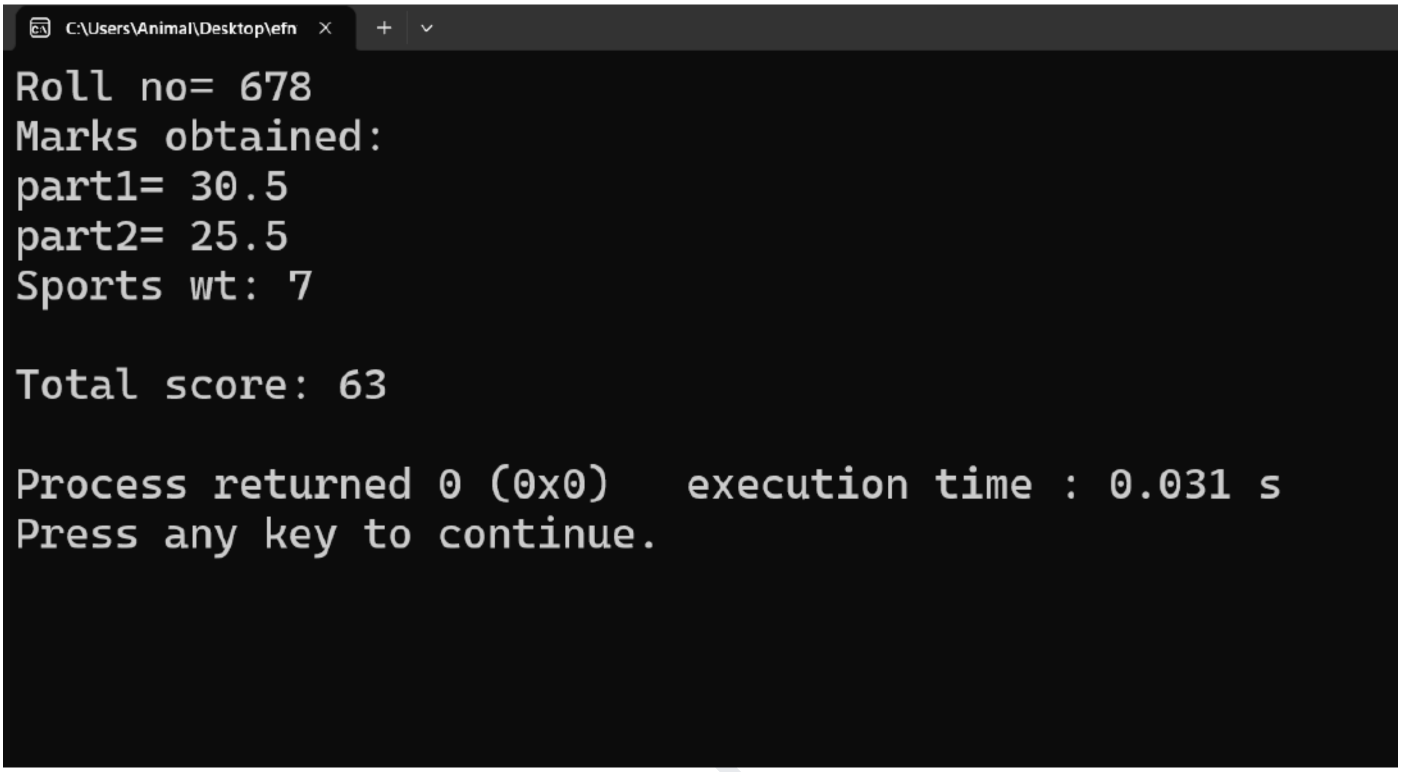


CODE:-

```
#include<iostream>
using namespace std;
class student
{
   protected:
      int roll_number;
   public:
      void get_number(int a)
      {
         roll_number = a;
      }
      void put_number(void)
      {
         cout<<"Roll no= "<<roll_number<<"\n";
      }
};
class test : virtual public student
{
protected:
   float part1, part2;
public:
   void get_marks(float x, float y)
   {
      part1 = x; part2 = y;
   }
   void put_marks(void)
   {

  cout<<"Marks obtained:"<<"\n"
```

```
        <<"part1= "<<part1<<"\n"
        <<"part2= "<<part2<<"\n";
    }
};
class sports : public virtual student
{
protected:
    float score;
public:
    void get_score(float s)
    {
        score=s;
    }
    void put_score(void)
    {
        cout<<"Sports wt: "<<score<<"\n\n";
    }
};
class result:public test, public sports
{
    float total;
public:
    void display(void);
};
void result :: display(void)
{
    total=part1+part2+score;
    put_number();
    put_marks();
    put_score();

    cout<<"Total score: "<<total<<"\n";
}
int main()
{
    result student_1;
    student_1.get_number(678);
    student_1.get_marks(30.5,25.5);
    student_1.get_score(7.0);
    student_1.display();
    return 0;}
```

OUTPUT:-



```
Roll no= 678
Marks obtained:
part1= 30.5
part2= 25.5
Sports wt: 7

Total score: 63

Process returned 0 (0x0)   execution time : 0.031 s
Press any key to continue.
```

PRACTICAL 7

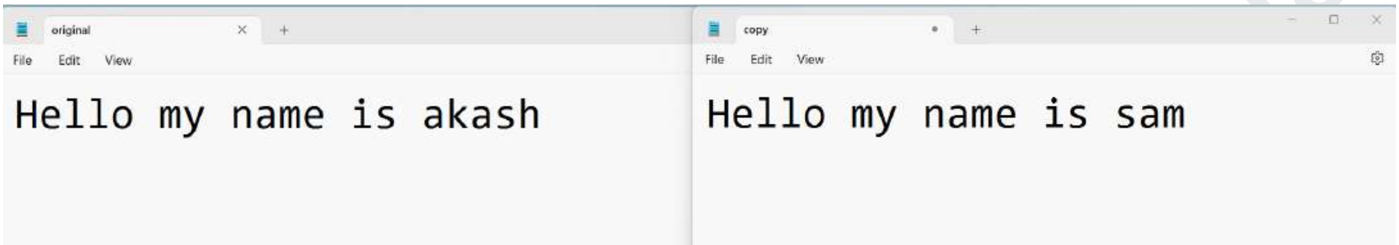
AIM:- Write a C++ Program to copy the contents of one file to another.

CODE:-

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

int main()
{
    string line;
    ifstream ini_file {"original.txt"};
    ofstream out_file {"copy.txt"};

    if(ini_file && out_file){

        while(getline(ini_file,line)){
            out_file << line << "\n";
        }

        cout << "Copy Finished \n";

    } else {
        printf("Cannot read File");
    }
    ini_file.close();
    out_file.close();

    return 0;
}
```

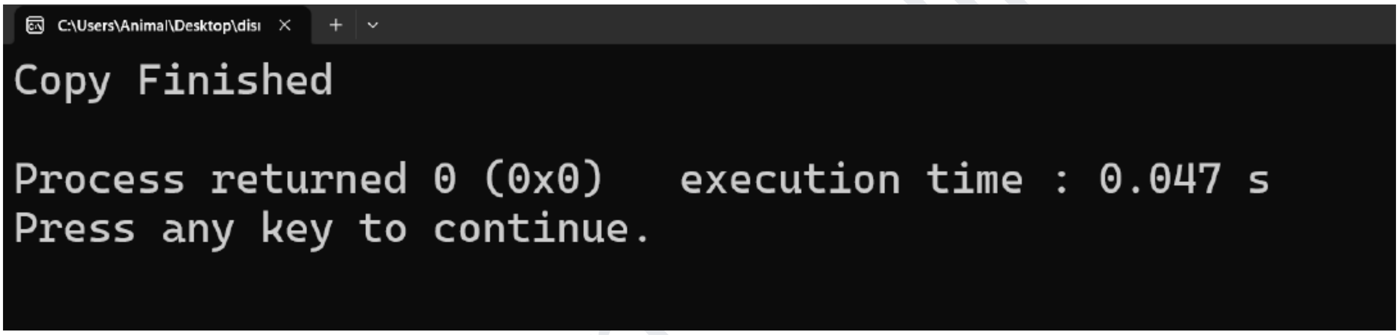
OUTPUT:-

Before:-



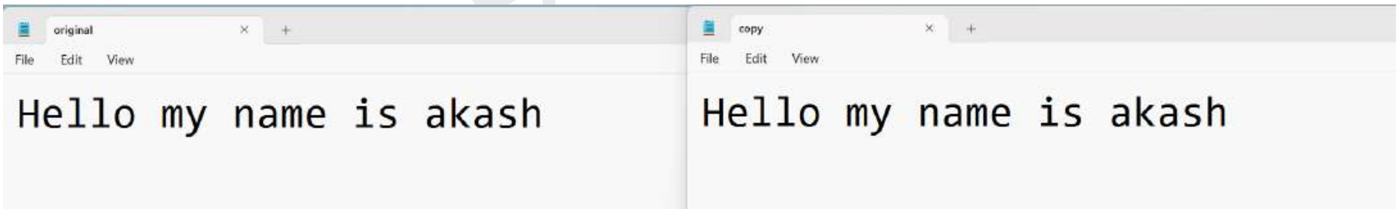
```
Hello my name is akash
```

```
Hello my name is sam
```



```
Copy Finished

Process returned 0 (0x0)   execution time : 0.047 s
Press any key to continue.
```

After:-



```
Hello my name is akash
```

```
Hello my name is akash
```

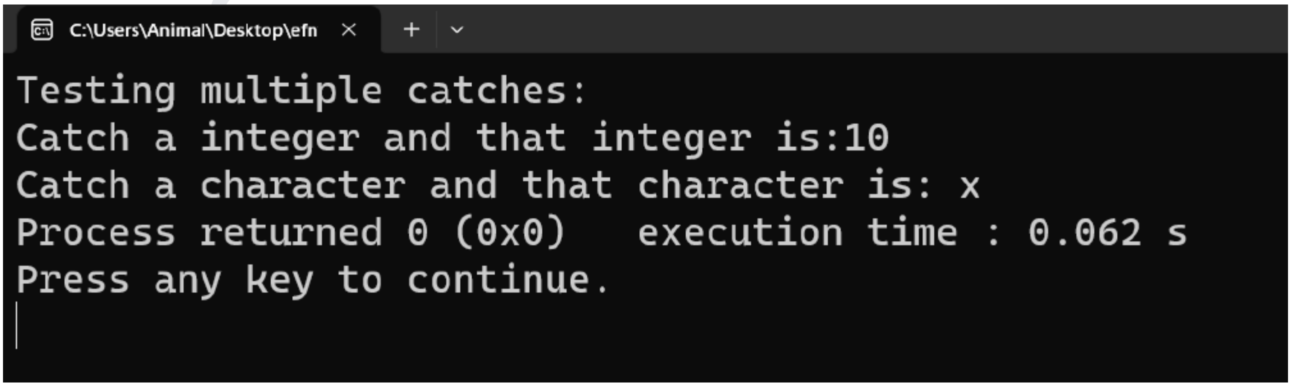
PRACTICAL 8

AIM:- Write a C++ program to implement the exception handling with multiple catch statements.

CODE:-

```
#include<iostream>

using namespace std;
void test(int x) {
    try {
        if (x > 0)
            throw x;
        else
            throw 'x';
    } catch (int x) {
        cout << "Catch a integer and that integer is:" << x;
    } catch (char x) {
        cout << "\nCatch a character and that character is: " << x;
    }
}
int main() {
    cout << "Testing multiple catches:\n";
    test(10);
    test(0);
}
```

OUTPUT:-



```
Testing multiple catches:
Catch a integer and that integer is:10
Catch a character and that character is: x
Process returned 0 (0x0)   execution time : 0.062 s
Press any key to continue.
```