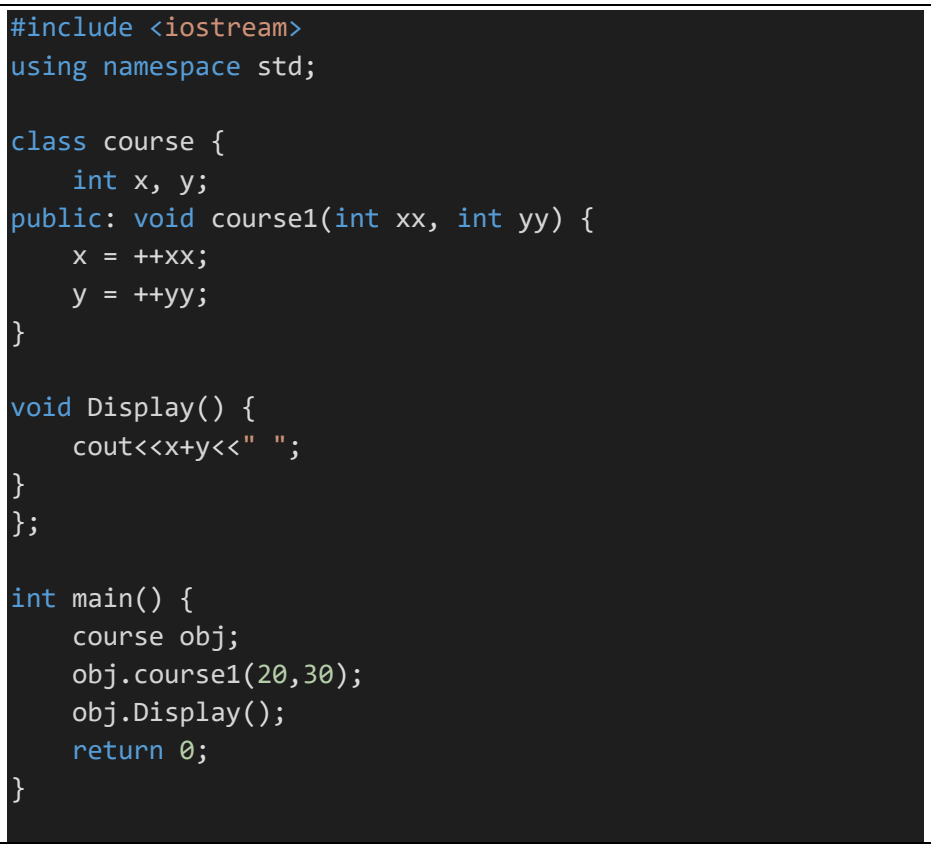
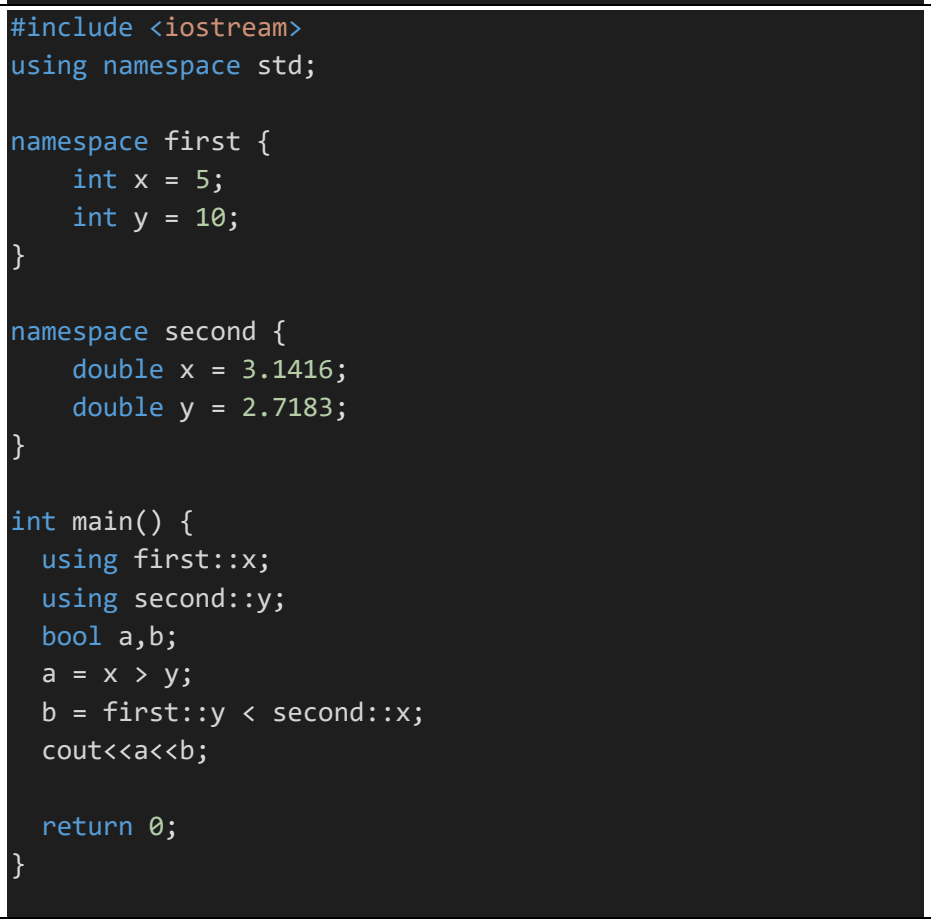
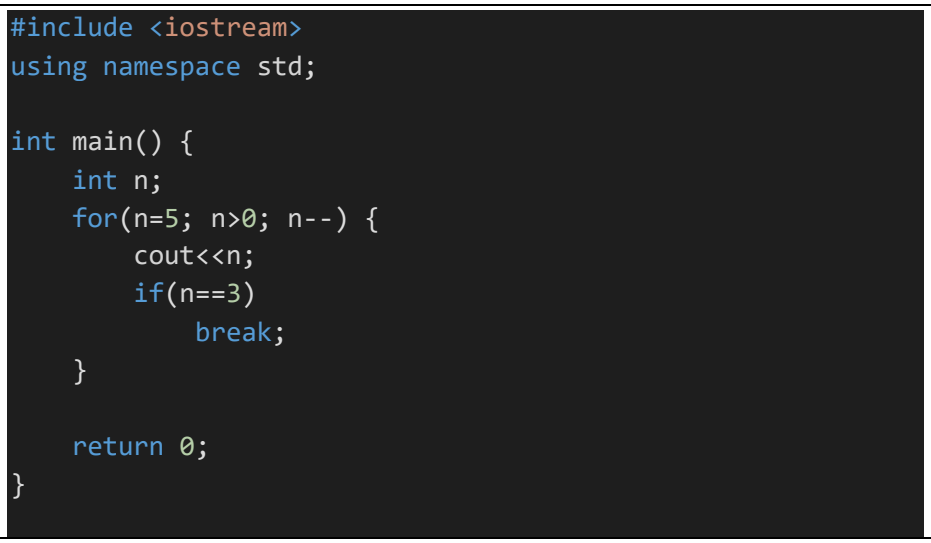
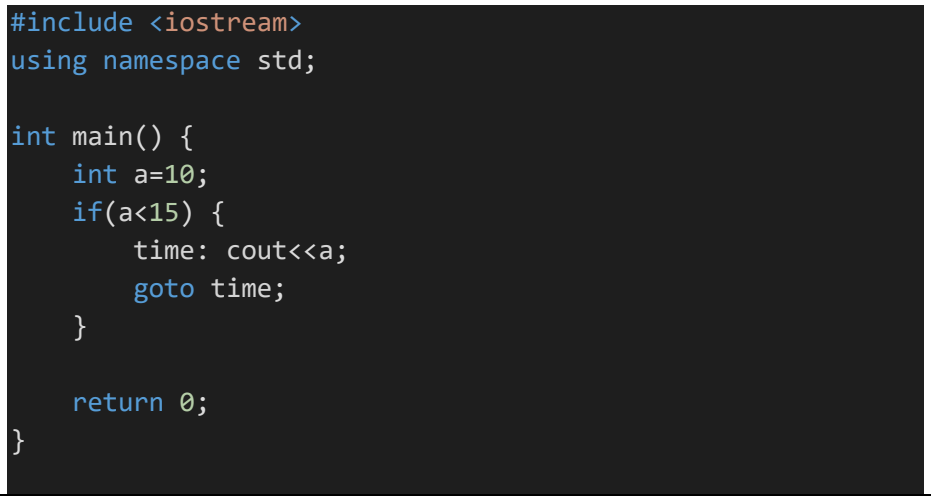
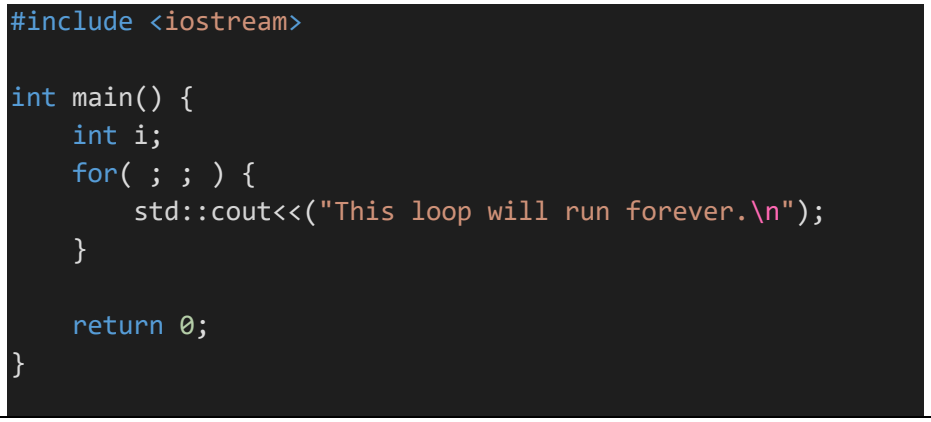
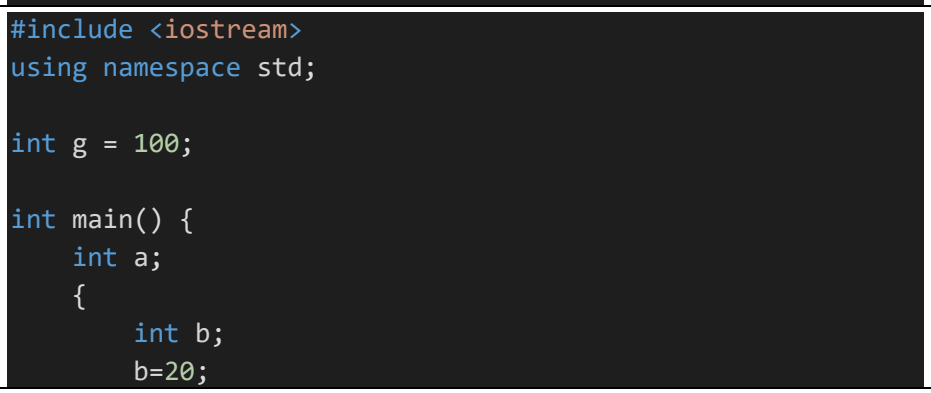
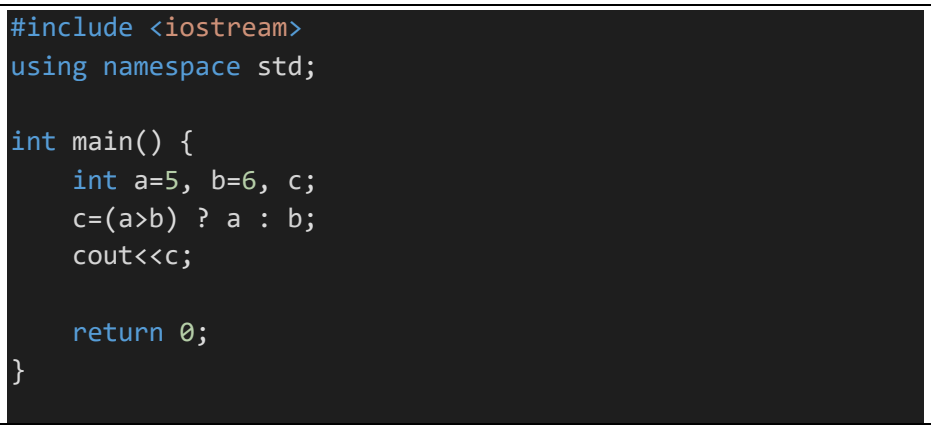
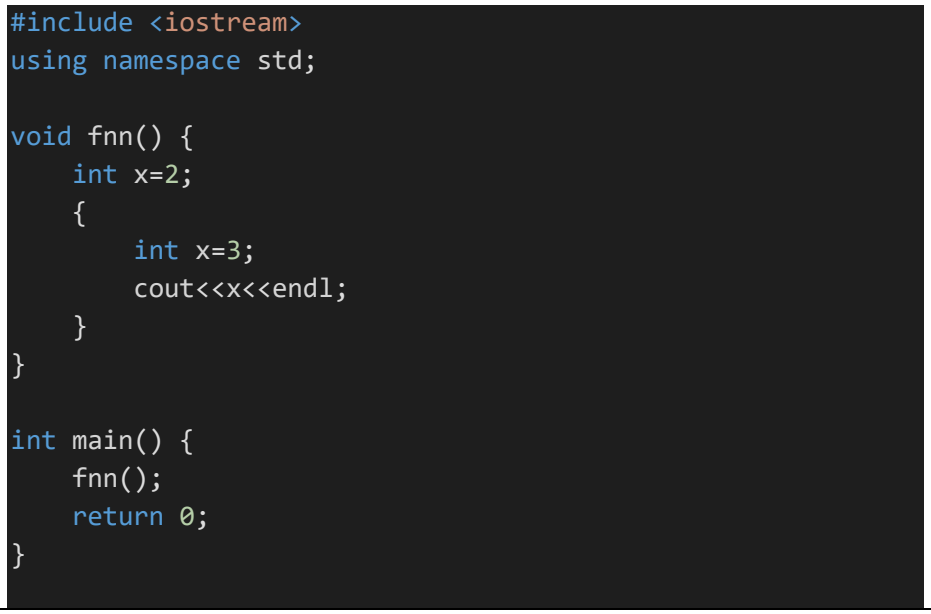
1.	 #include <iostream> using namespace std; class course { int x, y; public: void course1(int xx, int yy) { x = ++xx; y = ++yy; } void Display() { cout<<x+y<<" "; } }; int main() { course obj; obj.course1(20,30); obj.Display(); return 0; }	52
2.	 #include <iostream> using namespace std; namespace first { int x = 5; int y = 10; } namespace second { double x = 3.1416; double y = 2.7183; } int main() { using first::x; using second::y; bool a,b; a = x > y; b = first::y < second::x; cout<<a<<b; return 0; }	10
3.	 #include <iostream> using namespace std;	52

	`` int f(int p, int q) { if(p > q) return p; else return q; } int main() { int a=5, b=10; int k; bool x=true; bool y=f(a,b); k = ((a*b)+(x+y)); cout<<k; } ```	
4.	``` #include <iostream> using namespace std; int main() { int i,j; j = 10; i = (j++, j +100, 999+j); cout<<i; return 0; } ```	1010
5.	``` #include <iostream> using namespace std; class Room { public: double length; double breadth; double height; double calculateArea() { return length * breadth; } double calculateVolume() { return length * breadth * height; } }; int main() { Room room1; room1.length = 4.5; room1.breadth = 3.8; ```	17.1 157.32

	`` room1.height = 9.2; cout<<room1.calculateArea()<<room1.calculateVolume(); return 0; } ```	
6.	``` #include <iostream> using namespace std; #define PI 3.14159 int main() { float r = 2; float circle; circle = 2 * PI * r; cout<<circle; return 0; } ```	12.5664
7.	``` #include <iostream> using namespace std; int main() { int a = 10; if(a<10) { for(i=0;i<10;i++) cout<<i; } return 0; } ```	error
8.	``` #include <iostream> using namespace std; int x=1; void fun() { int x = 2; { int x = 3; cout<<::x<<endl; } } int main() { fun(); return 0; } ```	1

9.		543
10.		infinitely print 10
11.		This loop will run forever infinite times
12.		2035655065

	`` a=35; g=65; cout<<b<<a<<g; } a=50; cout<<a<<g; return 0; } ```	
13.	``` #include <iostream> using namespace std; int main() { int i; if(cout<<"0") i=3; else i=5; cout<<i; return 0; } ```	3
14.	``` #include <iostream> int main() { int i=0,x=0; for(i=1;i<10;i*=2) { x++; std::cout<<x; } std::cout<<x; return 0; } ```	12344
15.	``` #include <iostream> using namespace std; int main() { int i=3; int l=i/-2; int k=i%-2; cout<<l<<k; return 0; } ```	-1 1

16.		6
17.		3
18.		findcourse
19.		Compiler error

	`` class A { public: void f(){ cout<<"A::f()"<<endl; } }; class B:public A{ public: void fb(){ cout<<"A::fb()"<<endl; } }; class C:public A{ public: void fc(){ cout<<"A::fc()"<<endl; } }; class D: public B,public C{ public: void fd(){ cout<<"A::fd()"<<endl; } }; int main() { D obj; obj.f(); return 0; } ```	
20	``` #include <iostream> using namespace std; class B { int b; public: B(int i) { b=i; } }; class C { B b;public:C(int i) { b=B(i); ```	error

	`}` `friend void show();` `};` `void show()` `{` `C c(10);` `cout<<"value of b is:"<<c.b.b<<endl;` `int main();` `return 0;` `}`	
21.	`#include <iostream>` `using namespace std;` `int main()` `{` `int a=5,b=6,c;` `c=(a>b)?(cout<<"a is greater"):(cout<<"b is greater");` `cout<<c;` `return 0;` `}`	error: cannot convert 'std::basic_ostream<char>' to 'int' in assignment c=(a>b)?(cout<<"a is greater"):(cout<<"b is greater");
22.	`#include <iostream>` `using namespace std;` `int main()` `{` `int n;` `for(n=5;n>0;n--)` `{` `cout<<n;` `if(n==3)` `break;` `}` `return 0;` `}`	543
23.	`#include <iostream>` `using namespace std;` `int main()` `{` `int n=1;` `cout<<"the numbers are:";` `do {` `cout<<n<<endl;` `n++;` `}` `while(n<=100);` `cout<<endl;` `return 0;` `}`	Print natural numbers 1 to 100
24.	`#include <iostream>` `using namespace std;` `int main()` `{`	3

	`` int i; if (cout<<"0:")i=3; else i=5; printf("%d",i); return 0; } ```	
25.	``` #include <iostream> using namespace std; class Box { private: int length; public: Box() { length(0) {} friend int printlength(Box); }; int printlength(Box b) { b.length += 10; return b.length; } int main() Box b; cout << printlength(b) << endl; return 0; } ```	error
26.	``` #include <iostream> using namespace std; int main() { int a = 5, b = 6, c; c = (a > b) ? a : b; cout << c; return 0; } ```	6
27.	``` #include <iostream> using namespace std; class B { int b; public: B(int i) { b=i; } }; class C ```	error

	{ B b;public:C(int i) { b=B(i); } friend void show(); }; void show() { C c(10); cout<<"value of b is:"<<c.b.b<<endl; int main(); { show(); } return 0; }	
28.	#include <iostream> using namespace std; namespace ns1 { int a=4; } namespace ns2 { int a=8; } int main() { int a=12; ns1::a,ns2::a; cout<<ns1::a<<endl; cout<<ns2::a<<endl; cout<<a; return 0; }	4 8 12
29.	#include <iostream> using namespace std; class Test { protected: int x; public: Test(int i):x(i) {} void fun() const { cout<<"fun()const"<<endl; } void fun() {	fun()….fun()const

```
        cout<<"fun()"<<endl;
    }
};
int main()
{
    Test t1(10);
    const Test t2(20);
    t1.fun();
    t2.fun();

    return 0;
}
```

30.

```
#include <iostream>
using namespace std;
class A
{
    int a;
    public:
    int assign(int i)
    const
    {
        a=i;
    }
    int return_value()
    const
    {
        return a;
    }
};
int main()
{
    A obj;
    obj.assign(5);
    cout<<obj.return_value();

    return 0;
}
```

error

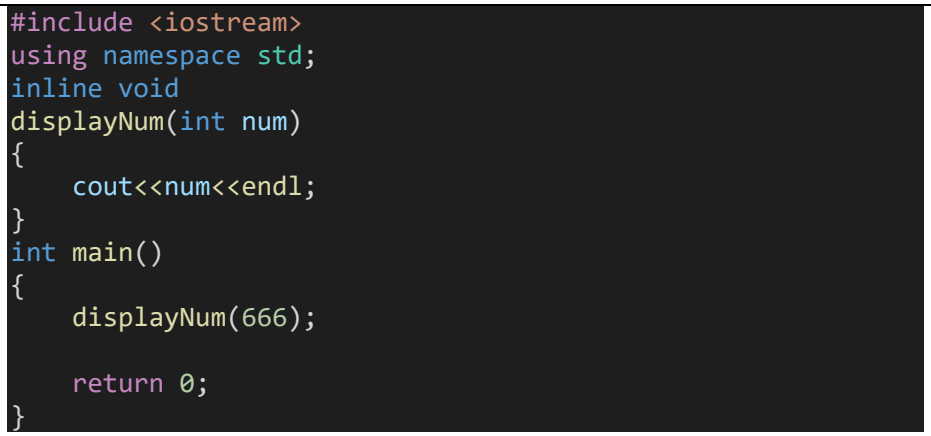
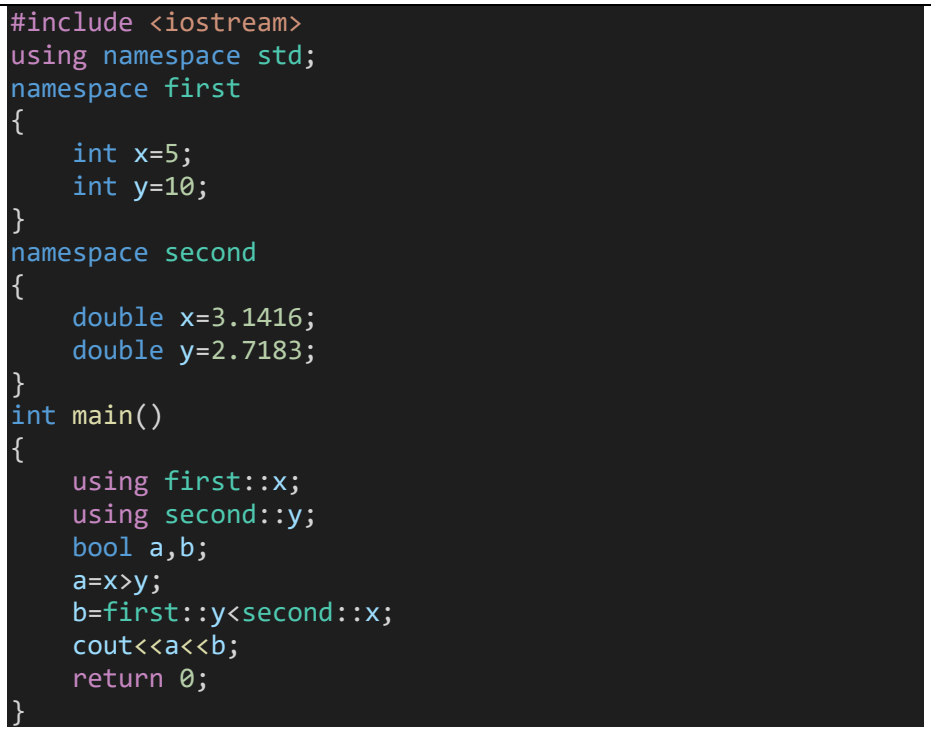
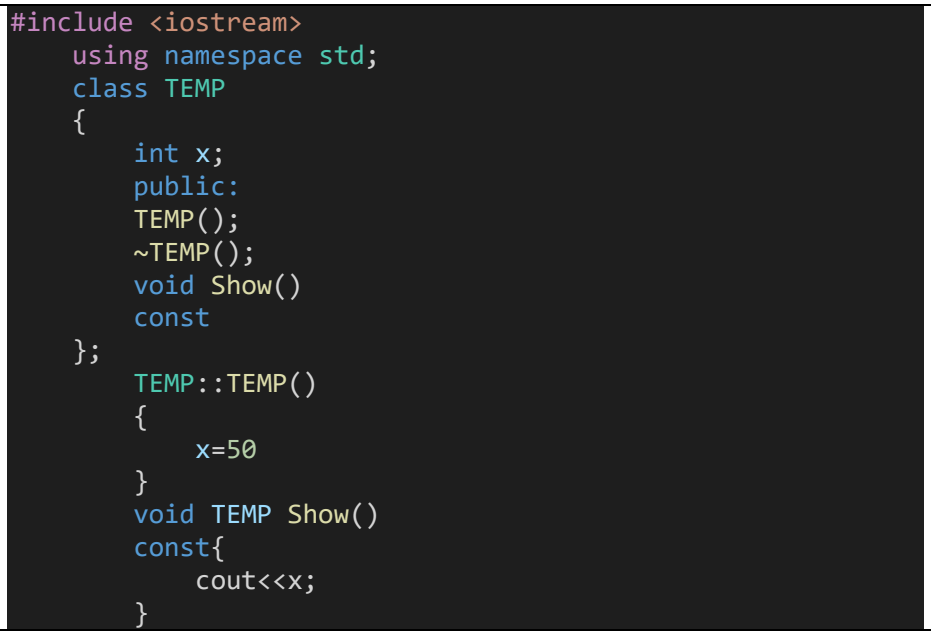
31.

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

int main()
{
   int n=1;
   cout<<"the num are";
   do
   {
       cout<<n<<endl;
       n++;
   }
   while(n<=100);
   cout<<endl;
```

Print natural numbers 1 to 100

	`` return 0; } ```	
32.	``` #include <iostream> using namespace std; #define PI int main() { float r=2; float circle; circle=2*PI*r; cout<<circle; return 0; } ```	Compile time error
33.	``` #include <iostream> using namespace std; int main() { int i,a=10; if(a<10){ for(i=0;i<10;i++) cout<<i; } else { cout<<i; } return 0; } ```	Garbage value
34.	``` #include <iostream> class Test { public: int i; void get(); }; void Test::get() { std::cout << "enter the value of i:"; std::cin >> i; } Test t; int main() { Test t; t.get(); std::cout << "value of i in local t:" << 'n'; ::t.get(); std::cout << "value of i in global t:" << ::t.i << 'n'; return 0; } ```	Compiles and run fine

35.		666
36.		10
37.		Compile time error

	`` int main() { TEMP obj; obj.Show(); return 0; } ```	
38.	``` #include <iostream> using namespace std; class ABC { public: ABC() { cout<<"find"; } ~ABC() { cout<<"course"; } }; int main() { ABC obj; return 0; } ```	findcourse
39.	``` #include <iostream> using namespace std; int main() { int a = 5, b = 6, c; c = (a > b) ? a : b; cout << c; return 0; } ```	6
40.	``` #include <iostream> using namespace std; int grades(int a = 0, int b= 0, int c) { return (a+b+c); } int main() { cout << grades(10); return 0; } ```	Compile time error
41.	``` #include <iostream> using namespace std; class A ```	error

```
{
    int x;
    public:
    void setX(int i) { x = i;}
    void print() { cout << x;}
};

class B:public A
{
    public:
    B() {setX(10);}
};

class C:public A
{
    public:
    C() {setX(20);}
};

class D:public B, public C{

};

int main()
{
    D d;
    d.print();
    return 0;

}
```

42.

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

abstract class student
{
    public: int grades;
    calc_marks();
}
class student1:public student
{
    public : calc_marks()
    {
        return 20;
    }
};
class student2:public student
{
```

Option d(Class fail)

```
    public : calc_marks()
    {
        return 30;
    }
};
class fail {
    int grades;
};
```

43.

```
#include <iostream>
using namespace std;
int main()
{
    int arr[] = {10,20,30,40,50};
    int*p = arr;
    p+= 3;
    cout<<*p;
    return 0;
}
```

40

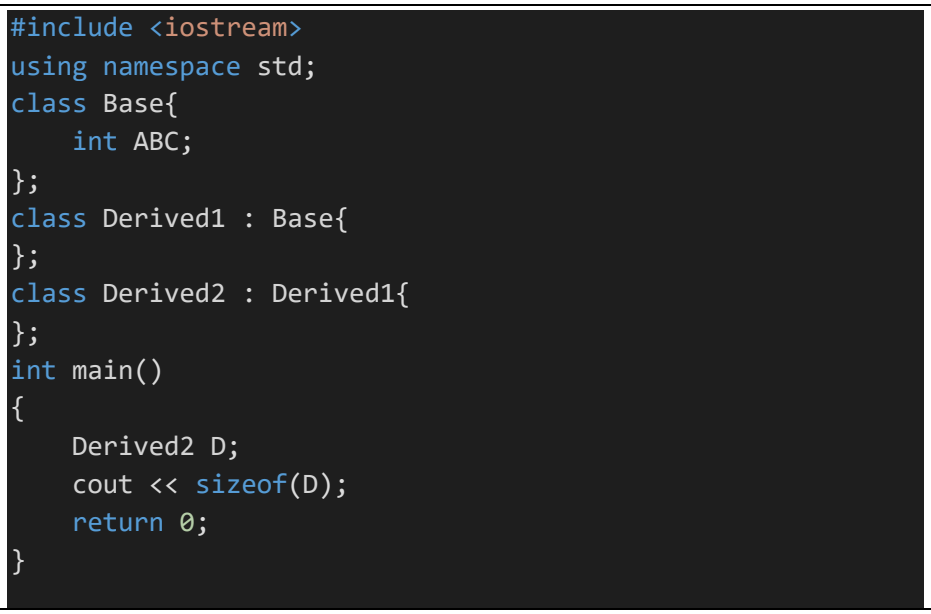
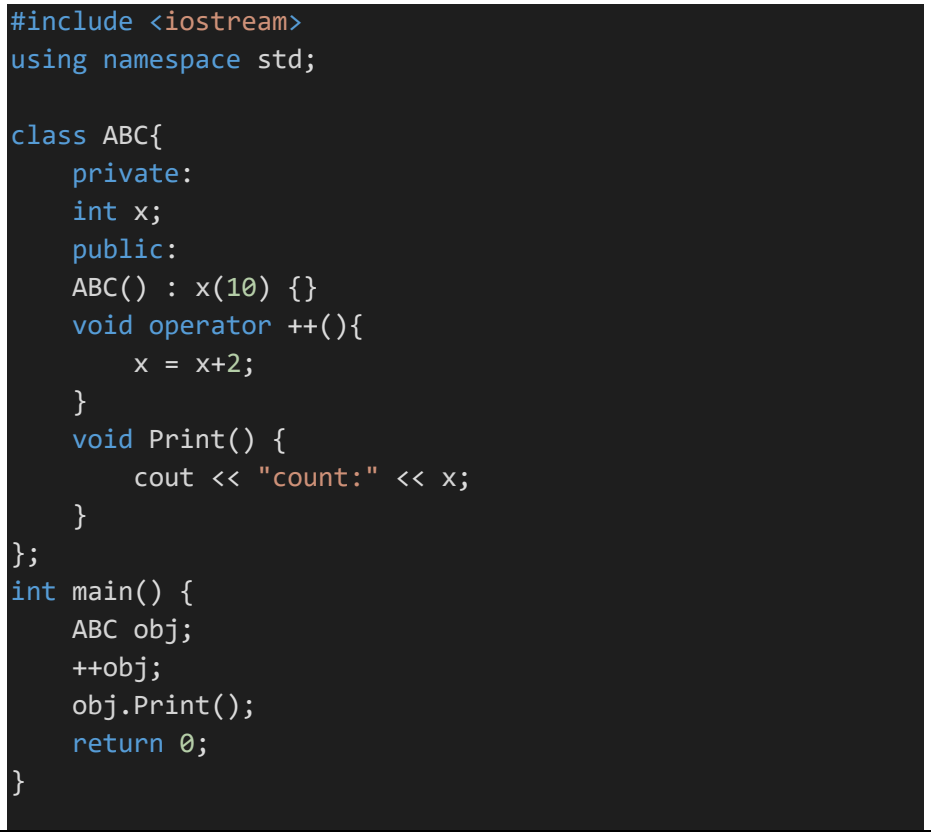
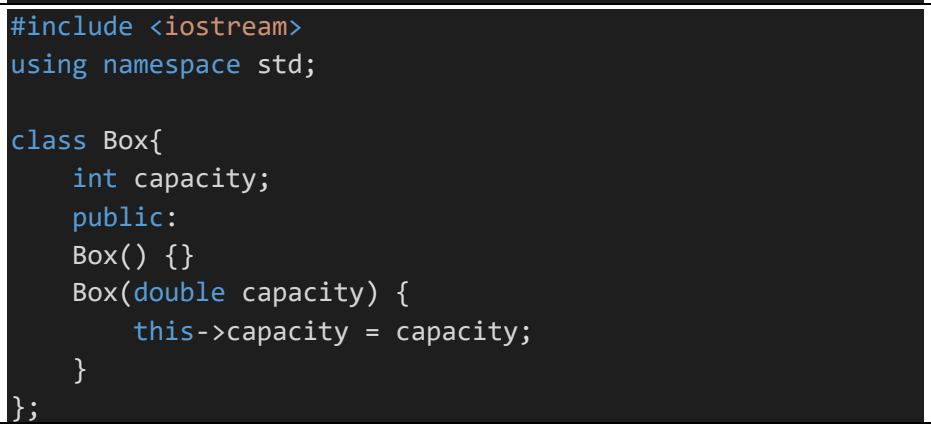
44.

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

class student {
    private:
    int age;
    public:
    student() {
        age = 20;
    }
    student(int a) {
        age = a;
    }
    int getAge() {
        return age;
    }
};
int main() {
    student stu1, stu2(25);
    cout<< "student1 Age = " << stu1.getAge()<< endl;
    cout<< "student2 Age = " << stu2.getAge()<< endl;

    return 0;
}
```

student1 Age=20
student2 Age=25

45.		4
46.		count:12
47.		Bool operator==(box b) { return this->capacity < b.capacity ? true : false; }

	`` int main() { Box b1(10); Box b2 = Box(14); if(b1 == b2) { cout<<"Equal"; } else{ cout<<"Not Equal"; } return 0; } ```	
48.	``` #include <iostream> using namespace std; class Test{ protected: int x; public: Test(int i):x(i) {} void fun() const {cout << "fun() const" << endl; } void fun() {cout << "fun()" << endl; } }; int main() { Test t1 (10); const Test t2 (20); t1.fun(); t2.fun(); return 0; } ```	fun() fun() const
49.	``` #include <iostream> using namespace std; namespace first{ int var=5; } namespace second{ double var=3.1416; } int main() { int a; a=first::var+second::var; cout<<a; ```	8

	`` return 0; } ```	
50.	``` #include <iostream> using namespace std; class A { private: int x,y; public:void A(int a, int b) { x=a; y=b; } }; int main() { A s; return 0; } ```	Compile time error
51.	``` #include <iostream> using namespace std; class Box { private: int length; public: Box():length(0) {} friend int printLength(Box); }; int printLength(Box b) { b.length +=10; return b.length; } int main() { Box b; cout<<printLength(b)<<endl; return 0; } ```	10
52.	``` #include <iostream> using namespace std; class Box { int capacity; public: Box(int cap) { capacity = cap; ```	value of capacity is :10

	`` } friend void show(); }; void show() { Box b(10); cout<<"Value of capacity is:"<< b.capacity<<endl; } int main() { show(); return 0; } ```	
53.	``` #include <iostream> using namespace std; class base { int val1 , val2; public: void get() { cout<<"Enter two values:"; cin>>val1>>val2; } friend float mean(base ob); }; float mean(base ob) { return float(ob.val1 + ob.val2) /2; }; int main () { base obj; obj.get(); cout<<"\n Mean value is:" << mean(obj); return 0; } ```	15
54.	``` #include <iostream> using namespace std; class Point { public: Point() { cout << " Constructor called"; } }; int main() { Point t1; return 0; } ```	Constructor called

55.	#include <iostream> using namespace std; class Box { int capacity; public: Box(int cap) { capacity = cap; } friend void show(); }; void show() { Box b(10); cout<<"Value of capacity is:"<< b.capacity<<endl; } int main() { show(); return 0; }	Value of capacity is: 10
56.		
57.	#include <iostream> using namespace std; class TEMP { int x; public: TEMP(); ~TEMP(); void Show() const; }; TEMP::TEMP() { x = 50; } void TEMP::Show() const{ cout<< x; } int main() { TEMP obj; obj.Show(); return 0; }	50
58.	#include <iostream>	10

```
using namespace std;

int main()
{
    int i;
    for(i=0; i<10;i++); {
        cout<<i;
    }

    return 0;
}
```

59.

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

class A
{ int x;
public:
void setX(int i){x=i;}
void print() {cout<<x;}
};

class B : public A {
public:
B() {setX(10);}
};

class C : public A {
public:
C() {setX(20);
};

class D : public B, public C {

};

int main() {
    D d;
    d.print();
    return 0;
}
```

10 20

60.

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

int fun (int x,int *py, int **ppz)
{
    int y,z;
```

31

	ppz += 2; z = **ppz; *py += 12; y = *py; x += 3; return x+y+z; } int main() { int c,*b,a; c=4; b=&c; a=&b; cout<<fun(c,b,a); return 0; }	
61.	#include <iostream> using namespace std; class calculate { private: int val; public: calculate ():val(5){} void operator ++() { ++val; } void display() { cout <<"Calculated values is: "<<val<<endl; } }; int main () { calculate cal1; ++cal1; cal1.display(); return 0; }	Calculated value is: 6
63.	#include <iostream> using namespace std; class Test { int a; }; class Test1 {	Error : ambiguous call to void disp()

```
int x;
Test t2;
public:
operator Test() {return t2;}
operator int() {return x;}
};
void disp(int x){cout<<"disp(int) called";}
void disp (Test t){cout<<"disp(Test)called"; }
int main ()
{ Test1 t;
disp(t);
return 0;
}
```

64.

```
#include <iostream>
#include <string>
using namespace std;
class simcard
{
    int limit;

public:
    simcard() {}
    simcard(double limit)
    {
        this->limit = limit;
    }
    bool operator<(simcard b)
    {
        return b.limit < this->limit ? true : false;
    }
};

int main()
{
    simcard s1(10);
    simcard s2 = simcard(14);
    if (s1 < s2)
    {
        cout << "limit of simcard1 is less";
    }
    else
    {
        cout << "limit of simcard2 is less";
    }
    return 0;
}
```

limit of simcard2 is less

65.	```#include<iostream>
using namespace std;
int main()
{
int arr[]={10,20,30,40,50};
int*p=arr;
p+=3;
cout<<*p;
return 0;
}```	40
66.	```#include<iostream>
using namespace std;

int main()
{
    int arr[]={10,20,30,40,50};
    int*p=arr;
    p+=4;
    cout<<*p<<" ";
    --*p;
    cout<<*p;
    return 0;
}``` | 50 49 |
| 67. | ```#include<iostream>
using namespace std;
void disp(int x)
{
    cout <<x;
}
void disp(double y)
{
    cout<<y;
}
int main(void)
{
disp(8);
disp(500.263);
return 0;
}``` | 8500.263 |
| 68. | ```#include<iostream>
using namespace std;

class Box {
private :``` | 10 |

	int length; public: Box():length(0) {} friend int printLength(Box); }; int printLength (Box b) { b.length+=10; return b.length; } int main () { Box b; cout<<printLength(b)<<endl; return 0; }	
69.	#include<iostream> using namespace std; class Box { int capacity; public: Box(int cap) { capacity=cap; } friend void show(); }; void show() { Box b(10); cout<<"value of capacity is "<< b.capacity<<endl; } int main () { show(); return 0; }	Value of capacity is 10
70.	#include<iostream> using namespace std; int main() { int i, a = 10; if (a < 10) { for (i = 0; i < 10; i++) cout << i; }	0

	`` else { cout << i; } return 0; } ```	
71.	``` #include<iostream> using namespace std; namespace Box1 { int a =4; } namespace Box2 { int a =13; } int main () { int a=16; Box1::a; Box2::a; cout<< a; return 0; } ```	16
72.	``` #include<iostream> using namespace std; namespace ns1 { int a = 4; } namespace ns2 { int a = 8; } int main () { int a = 12; ns1::a; ns2::a; cout << ns1::a << endl; cout << ns2::a << endl; cout << a; return 0; } ```	4 8 12
73.	``` #include<iostream> using namespace std; class Box { private: int length; public: Box(): ```	10

	`` length(0) { } friend int printLength(Box); }; int printLength(Box b) { b.length += 10; return b.length; } int main() { Box b; cout << printLength(b) << endl; return 0; } ```	
74.	``` #include<iostream> using namespace std; class Box { int capacity; public: Box(int cap){ capacity = cap; } friend void show(); }; void show() { Box b(10); cout<<"Value of capacity is: "<< b.capacity<< endl; } int main() { show(); return 0; } ```	Value of the capacity is: 10
75.	``` #include <iostream> using namespace std; class B { int b; public: B(int i){ b = i; } }; class C{ B b; public: C(int i) { b= B(i); } friend void show(); ```	Error

	``}; void show() { C c(10); cout<<"value of b is: "<< c.b.b << endl; } int main(){ show(); return 0; }```	
76.	```#include<iostream> int main() { int i; for (; ;) { std::cout<<("This loop will run forever. \n"); } return 0; }```	This loop will run forever. infinitely
77.	```#include<iostream> using namespace std; int fun(int x=0, int y=0, int z) { return (x+y+z); } int main() { cout << fun(10); return 0; }```	Compiler error
78.	```#include <iostream> using namespace std; class Point { Point() { cout<<"Constructro called"; } }; int main() { Point t1; return 0; }```	Compiler error
79.	```#include <iostream> using namespace std; class course { int x,y; public: course(int xx){```	20 19

	`` x= ++xx; } void Display(){ cout<<--x<<" "; } }; int main() { course obj(20); obj.Display(); obj.Display(); return 0; } ```	
80.	``` #include <iostream> using namespace std; class TMP { int p; public:TMP(int xx, char ch){ p = xx + int(ch); cout<<p; } }; int main() { TMP obj(15,'A'); return 0; } ```	80
81.	``` #include <iostream> using namespace std; class A { static int x; public: static void Set(int xx){ x=xx; } void Display(){ cout<<x; } }; int A::x=0; int main() { A::Set(33); A::Display(); return 0; ```	The program will report compile time error

	`}`	
82.	`#include <iostream>` `using namespace std;` `class constt{` `public:` `int a, b;` `// Default Constructor constt() {a=10; b=20;}` `};` `int main() {` `// Default constructor called automatically` `// when the object is created constt c;` `constt c;` `constt c1(10,20);` `cout<<"a:"<<c.a<<endl<<"b:"<<c.b;` `return 1;` `}`	No output
83.	`#include <iostream>` `using namespace std;` `class A{` `private:` `int x, y;` `public:` `void A(int a, int b){` `x=a;` `y=b;` `}` `};` `int main() {` `A s;` `return 0;` `}`	Compile time error
84.	`#include <iostream>` `using namespace std;` `int i;` `class A{` `public:` `~A(){` `i=10;`	1

	`` } }; int foo(){ i=3; A ob; return 1; } int main() { cout<<foo()<<endl; return 0; } ```	
85.	``` #include <iostream> using namespace std; class course{ int x, y; public: course(int xx){ x=++xx; } void Display(){ cout<<--x<<" "; } }; int main() { course obj(20); obj.Display(); obj.Display(); return 0; } ```	20 19