**Lab – 3**

Array of objects, Passing and returning objects, Static and const

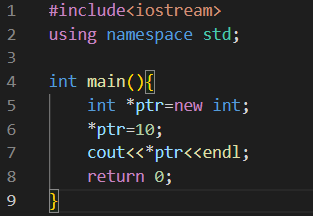
1. Implement array of objects with and without pointers. Here is how to use pointer for dynamic integer allocation and deletion.

// Creating and deleting int pointer

    int\* ptr1 = new int;

delete ptr1;

Code:



Output:



// Allocate and deallocate Heap memory

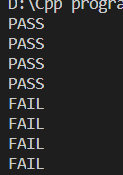
      int\* array = new int[10];

           delete[] array;

Code:

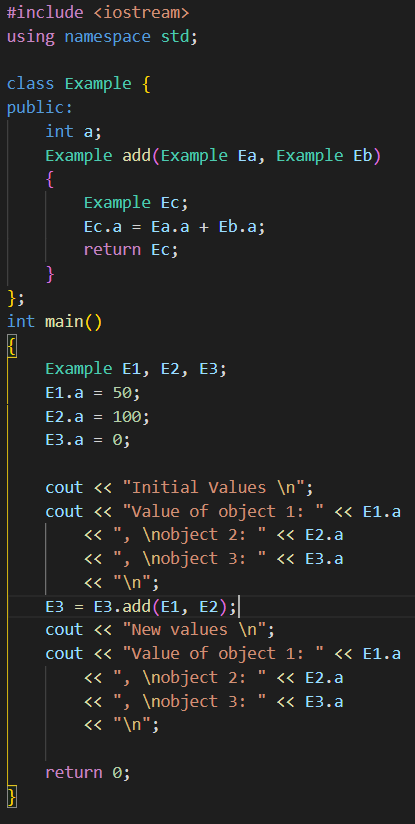


Output:

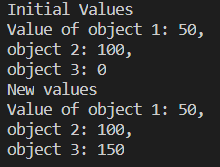


1. Write a program to pass an object as an argument and returning the object from a function.

Code:

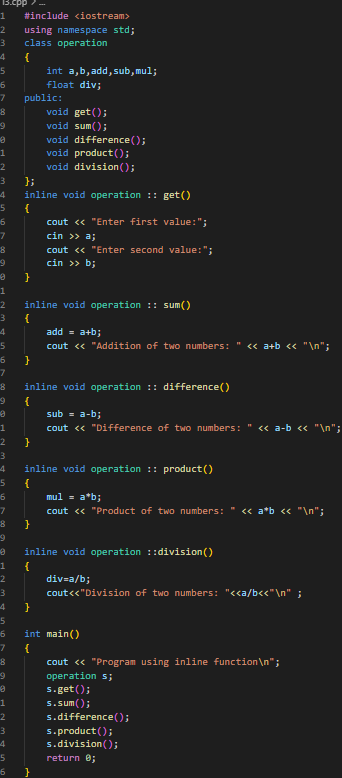


Output:

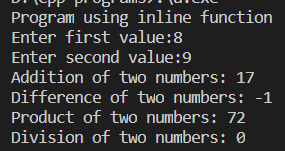


1. Practice inline function using a C++ program.

Code:



Output:



1. Complete the following program with both types of main functions separately in (a) and (b). Discuss the details with your lab instructor.

class Try {

public:

Try() {

// …constructor

}

~Try() {

// …destructor

}

void fun(){

cout << “In function” <<endl;

}

};

int main(){

Try\* try1 = new Try();

// … access fun() using try1 object

Try try2;

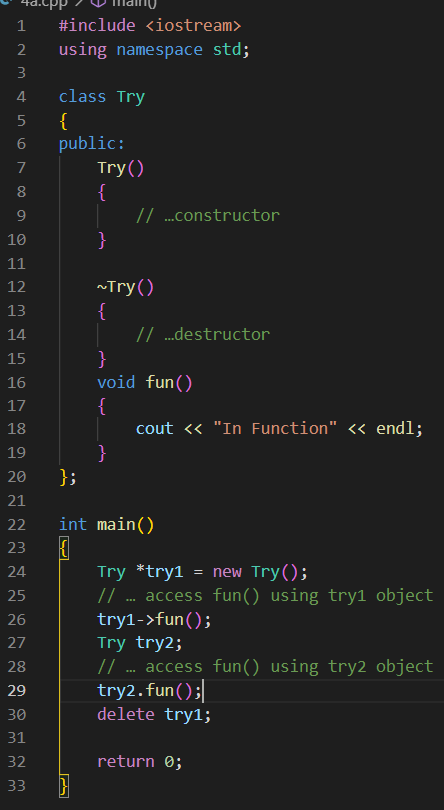
// … access fun() using try2 object

delete try1;

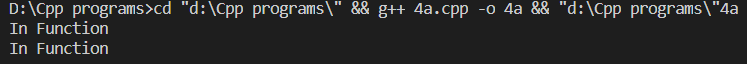
return 0;

}

Code:



Output:



int main(){

Try\* try = new Try[3];

try[0].fun();

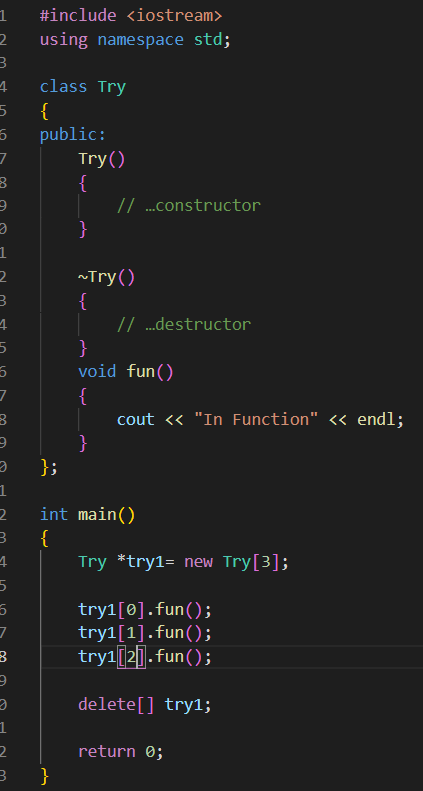
// … more like this

delete [] try;

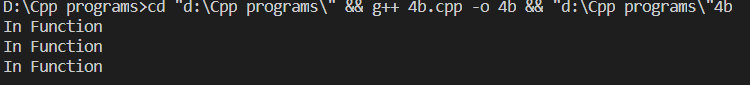
return 0;

}

Code:



Output:



1. Implement *static* data member and the member function using a C++ program.

Code:

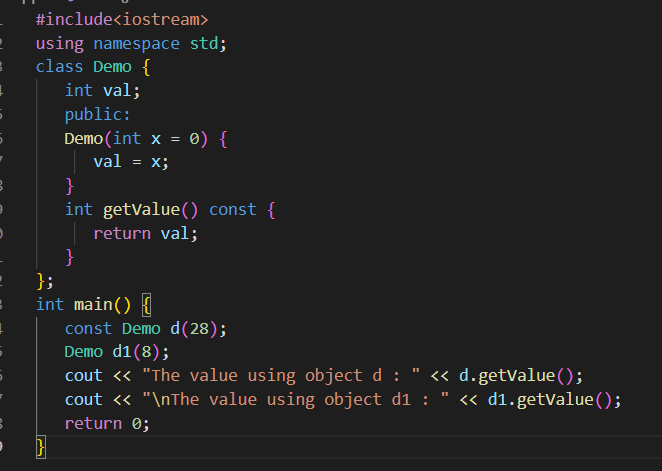


Output:



1. Implement *const* member function in C++ program.

Code:



Output:

