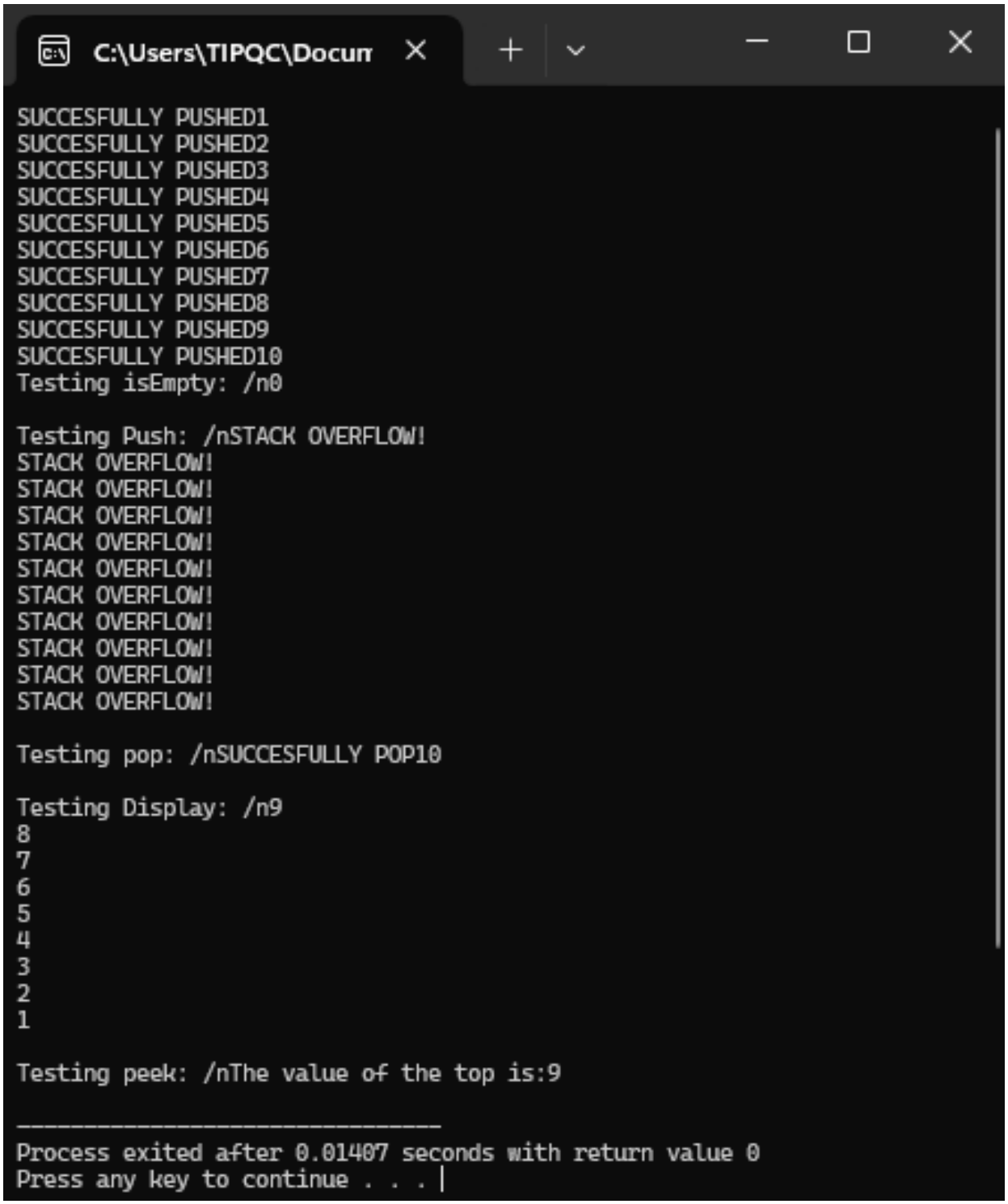
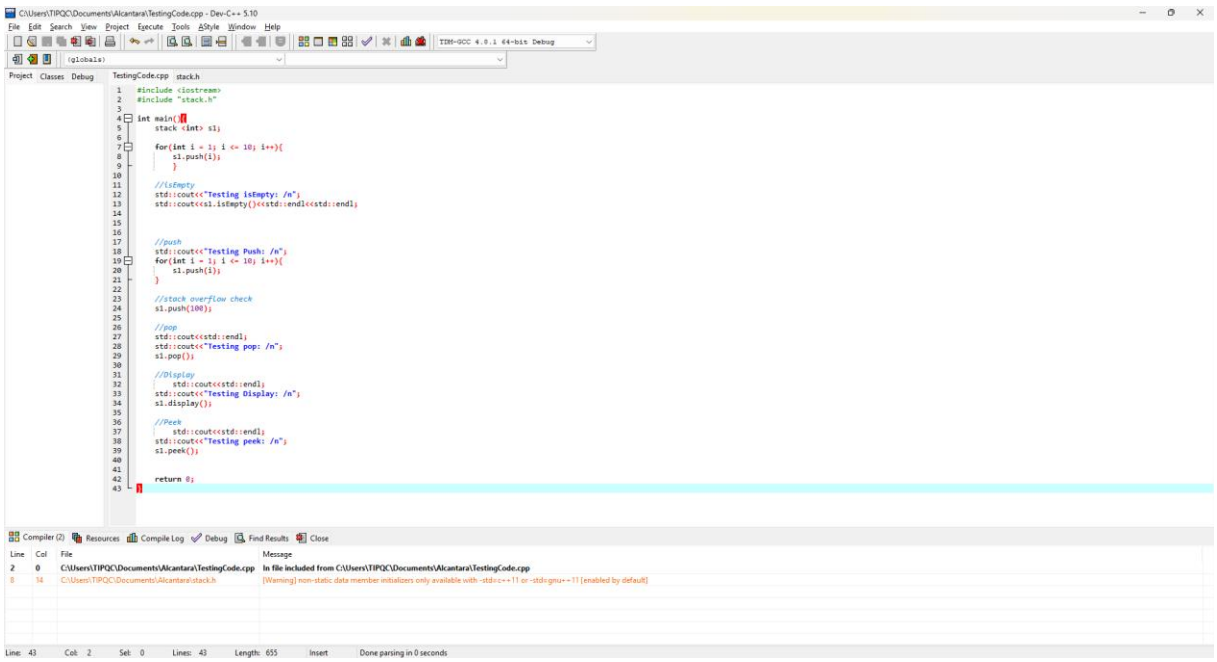
Activity No. <4.2>	
<Hands-on Activity 4.2 Stacks>	
Course Code: CPE010	**Program:** Computer Engineering
Course Title: Data Structures and Algorithms	**Date Performed: 8/28/25**
Section: CPE21S4	**Date Submitted: 8/28/25**
Name(s): Alcantara, Jason P.	**Instructor: Engr. Jimlord Quejado**

6. Output:



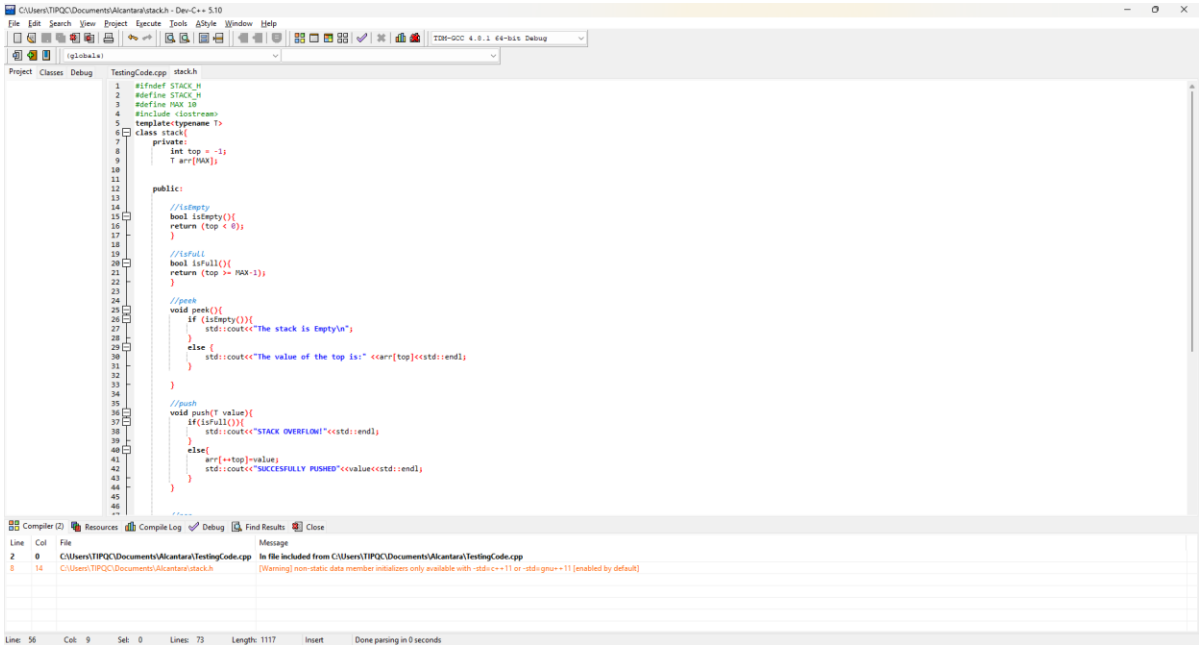
Syntax:



```
#include <iostream>
#include "stack.h"

int main(){
    stack <int> s1;

    for(int i = 1; i <= 10; i++){
        s1.push(i);
        }

    //isEmpty
    std::cout<<"Testing isEmpty: /n";
    std::cout<<s1.isEmpty()<<std::endl<<std::endl;


    //push
    std::cout<<"Testing Push: /n";
    for(int i = 1; i <= 10; i++){
        s1.push(i);
    }

    //stack overflow check
    s1.push(100);

    //pop
    std::cout<<std::endl;
    std::cout<<"Testing pop: /n";
    s1.pop();

    //Display
        std::cout<<std::endl;
    std::cout<<"Testing Display: /n";
    s1.display();

    //Peek
        std::cout<<std::endl;
    std::cout<<"Testing peek: /n";
    s1.peek();


    return 0;
}
```

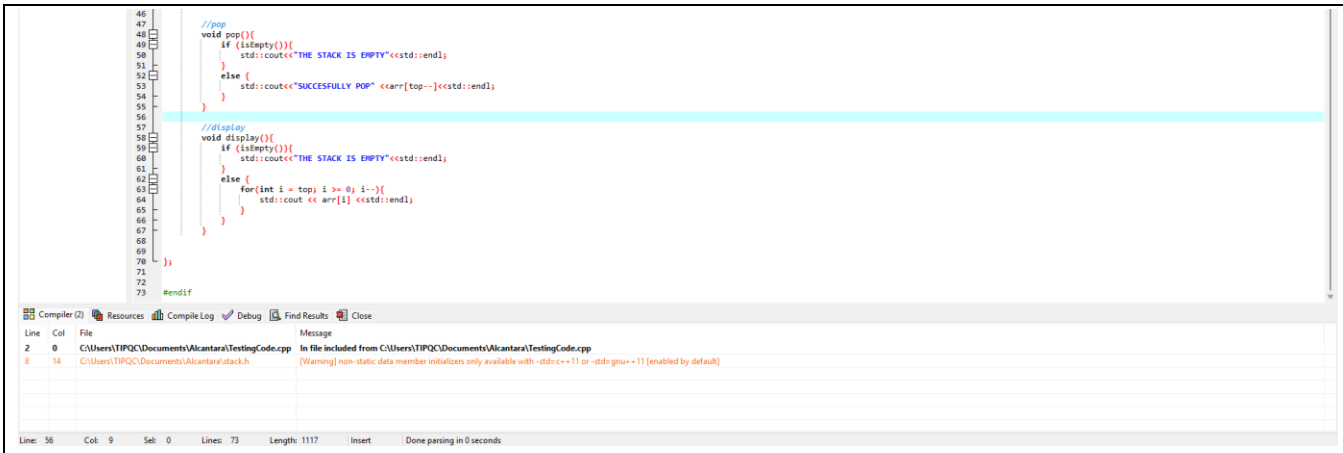
```
Compiler (2) | Resources | Compile Log | Debug | Find Results | Close

Line | Col | File | Message
2 | 0 | C:\Users\TIPQC\Documents\Alcantara\TestingCode.cpp | In file included from C:\Users\TIPQC\Documents\Alcantara\TestingCode.cpp
8 | 14 | C:\Users\TIPQC\Documents\Alcantara\stack.h | [Warning] non-static data member initializers only available with -std=c++11 or -std=gnu++11 [enabled by default]

Line: 43  Col: 2  Sel: 0  Lines: 43  Length: 655  Insert  Done parsing in 0 seconds
```

Header File:



```
#ifndef STACK_H
#define STACK_H
#define MAX 10
#include <iostream>
template<typename T>
class stack{
    private:
        int top = -1;
        T arr[MAX];

    public:

        //isEmpty
        bool isEmpty(){
        return (top < 0);
        }

        //isFull
        bool isFull(){
        return (top >= MAX-1);
        }

        //peek
        void peek(){
            if (isEmpty()){
                std::cout<<"The stack is Empty\n";
            }
            else {
                std::cout<<"The value of the top is:" <<arr[top]<<std::endl;
            }

        }

        //push
        void push(T value){
            if(isFull()){
                std::cout<<"STACK OVERFLOW!"<<std::endl;
            }
            else{
                arr[++top]=value;
                std::cout<<"SUCCESFULLY PUSHED"<<value<<std::endl;
            }
        }
```

```
Compiler (2) | Resources | Compile Log | Debug | Find Results | Close

Line | Col | File | Message
2 | 0 | C:\Users\TIPQC\Documents\Alcantara\TestingCode.cpp | In file included from C:\Users\TIPQC\Documents\Alcantara\TestingCode.cpp
8 | 14 | C:\Users\TIPQC\Documents\Alcantara\stack.h | [Warning] non-static data member initializers only available with -std=c++11 or -std=gnu++11 [enabled by default]

Line: 56  Col: 9  Sel: 0  Lines: 73  Length: 1117  Insert  Done parsing in 0 seconds
```



```
    //pop
    void pop(){
        if (isEmpty()){
            std::cout<<"THE STACK IS EMPTY"<<std::endl;
        }
        else {
            std::cout<<"SUCCESFULLY POP" <<arr[top--]<<std::endl;
        }
    }

    //display
    void display(){
        if (isEmpty()){
            std::cout<<"THE STACK IS EMPTY"<<std::endl;
        }
        else {
            for(int i = top; i >= 0; i--){
                std::cout << arr[i] <<std::endl;
            }
        }
    }


};

#endif
```

Compiler (2) | Resources | Compile Log | Debug | Find Results | Close

Line	Col	File	Message
2	0	C:\Users\TIPQC\Documents\Alcantara\TestingCode.cpp	In file included from C:\Users\TIPQC\Documents\Alcantara\TestingCode.cpp
8	14	C:\Users\TIPQC\Documents\Alcantara\stack.h	[Warning] non-static data member initializers only available with -std=c++11 or -std=gnu++11 [enabled by default]

Line: 56 Col: 9 Sel: 0 Lines: 73 Length: 1117 Insert Done parsing in 0 seconds

7. Supplementary Activity

8. Conclusion:

In our class activity I learned how to make cpp file more simple and header files, so the code is more neat easier to understand. And we discussed all important use of all elements when doing stacks for example pop, push, isEmpty, Display and peek so we will be more prepared in our practical exam. But I still need to study more about it because I still get confused.

9. Assessment Rubric