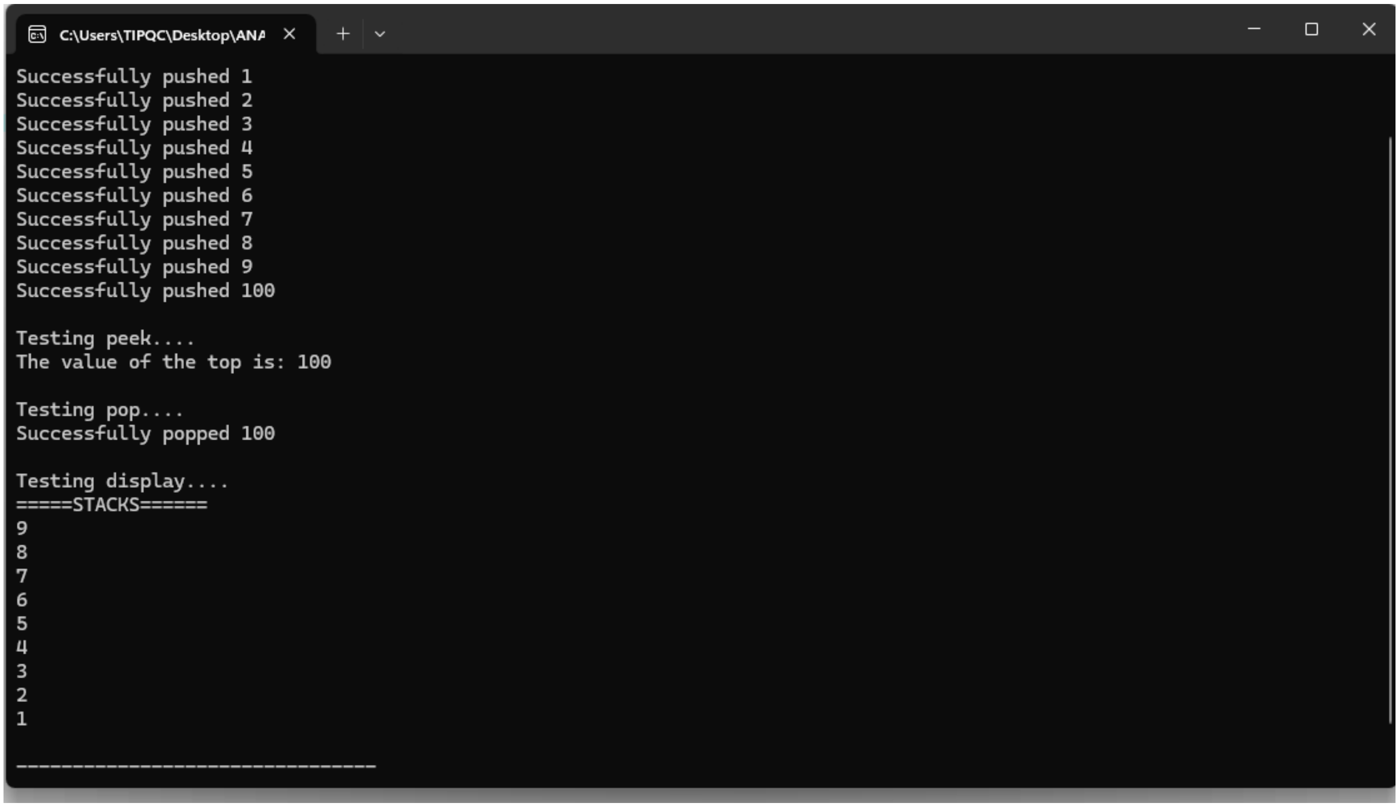
Activity No. <4.2>

<STACKS>

Course Code: CPE010	**Program:** Computer Engineering
Course Title: Data Structures and Algorithms	**Date Performed: 8/28/2025**
Section: CPE21S4	**Date Submitted: 8/28/2025**
Name(s): Anastacio, Lester Arvid P.	**Instructor: Engr. Jimlord Quejado**

6. Output

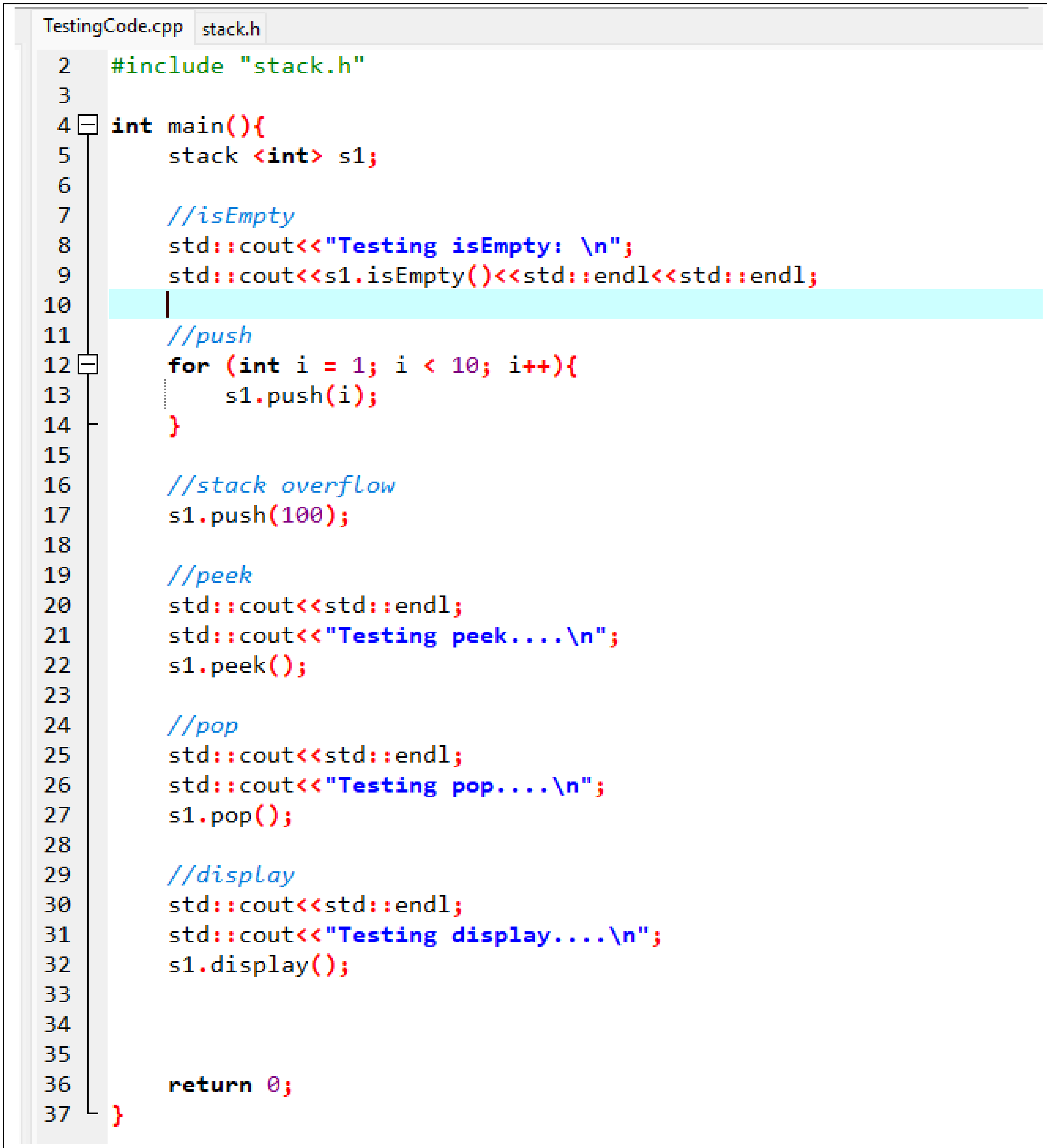
SCREENSHOTS:



```
Successfully pushed 1
Successfully pushed 2
Successfully pushed 3
Successfully pushed 4
Successfully pushed 5
Successfully pushed 6
Successfully pushed 7
Successfully pushed 8
Successfully pushed 9
Successfully pushed 100

Testing peek....
The value of the top is: 100

Testing pop....
Successfully popped 100

Testing display....
=====STACKS======
9
8
7
6
5
4
3
2
1

--------------------------------
```

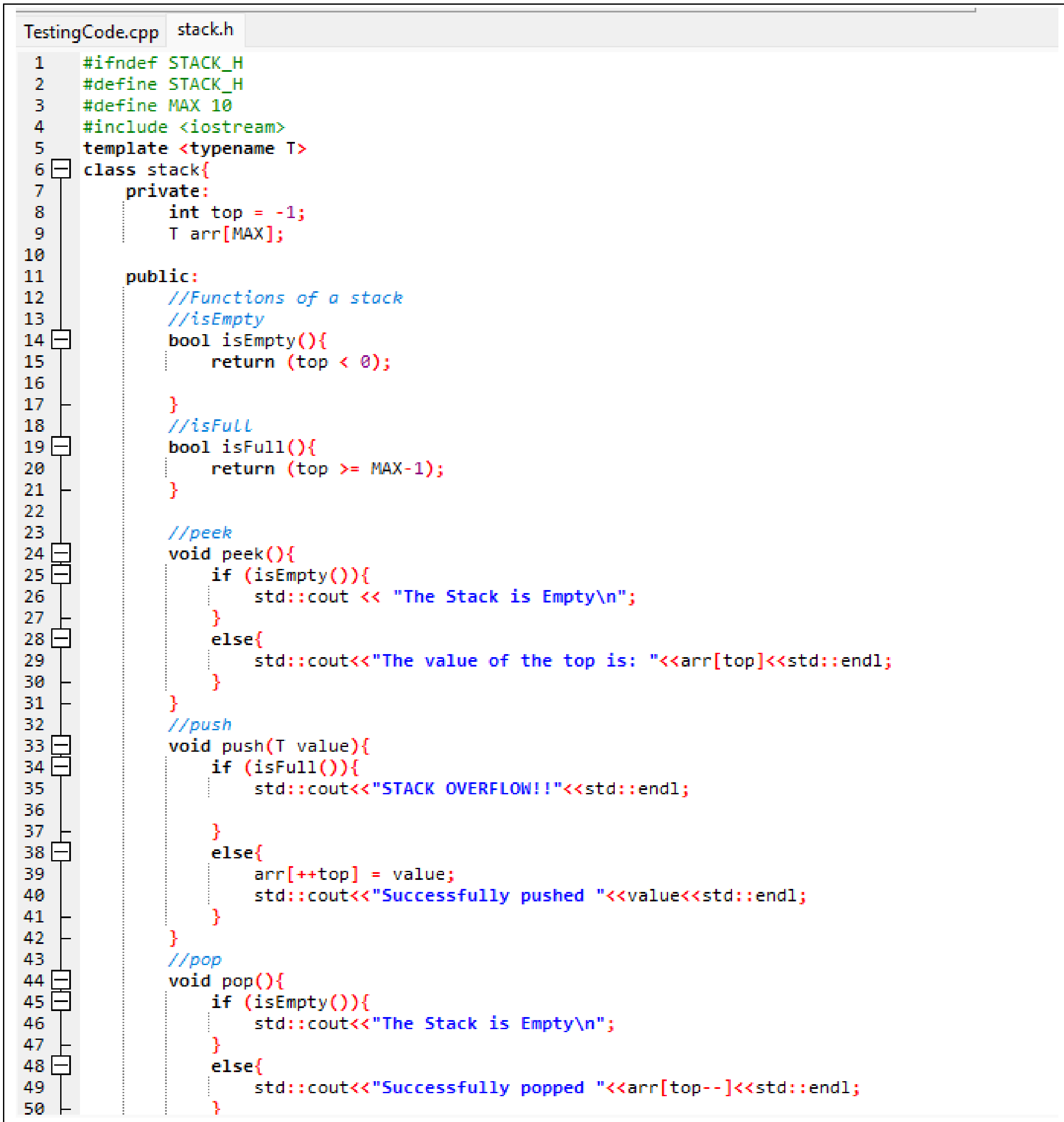
7. Supplementary Activity

Main File:

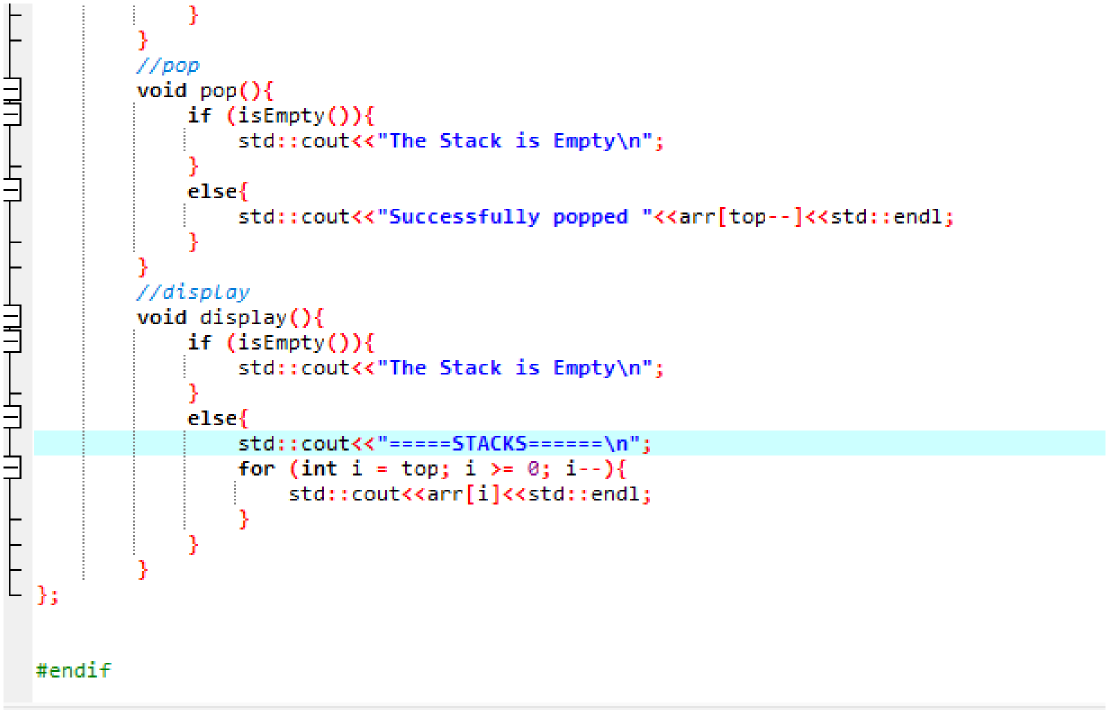


```cpp
#include "stack.h"

int main(){
    stack <int> s1;

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

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

    //stack overflow
    s1.push(100);

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

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

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



    return 0;
}
```

Header File:



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

    public:
        //Functions of a stack
        //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<<"Successfully pushed "<<value<<std::endl;
            }
        }
        //pop
        void pop(){
            if (isEmpty()){
                std::cout<<"The Stack is Empty\n";
            }
            else{
                std::cout<<"Successfully popped "<<arr[top--]<<std::endl;
            }
```



```
            }
        }
        //pop
        void pop(){
            if (isEmpty()){
                std::cout<<"The Stack is Empty\n";
            }
            else{
                std::cout<<"Successfully popped "<<arr[top--]<<std::endl;
            }
        }
        //display
        void display(){
            if (isEmpty()){
                std::cout<<"The Stack is Empty\n";
            }
            else{
                std::cout<<"=====STACKS======\n";
                for (int i = top; i >= 0; i--){
                    std::cout<<arr[i]<<std::endl;
                }
            }
        }
};


#endif
```

8. Conclusion

During this activity, I've learned how to properly create stacks from scratch and also learned the uses of header files, perimeters and templates to be able to use all the needed data types for the whole code to work especially the push function as it needed perimeters and such to actually add in the data that we will input onto the stack.

9. Assessment Rubric