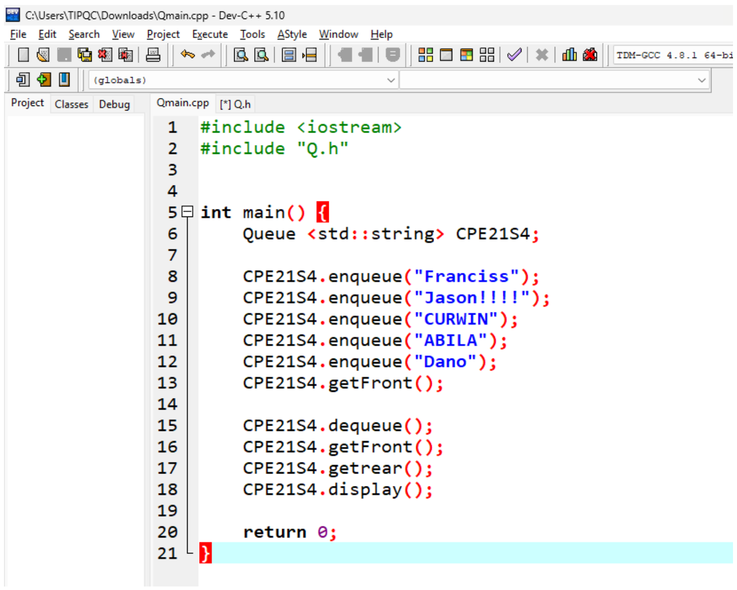
Activity No. <5.1>	
<Queue - Linked List Application>	
Course Code: CPE010	Program: Computer Engineering
Course Title: Data Structures and Algorithms	Date Performed: 9/9/25
Section: CPE21S4	Date Submitted: 9/9/25
Name(s): Alcantara, Jason P.	Instructor: Engr. Jimlord Quejado

6. Output:

Syntax:

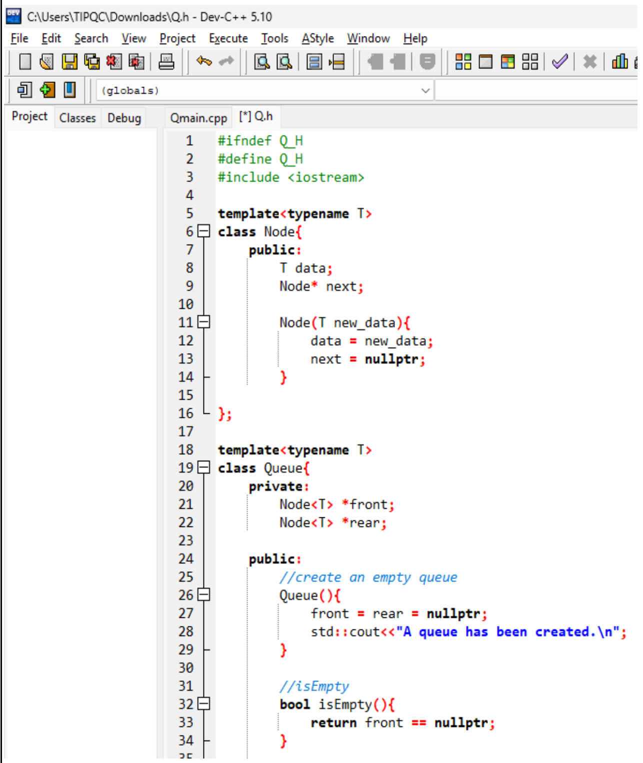
Main cpp:



C:\Users\TIPQC\Downloads\Qmain.cpp - Dev-C++ 5.10
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.8.1 64-bi
(globals)
Project Classes Debug
Qmain.cpp [*] Q.h

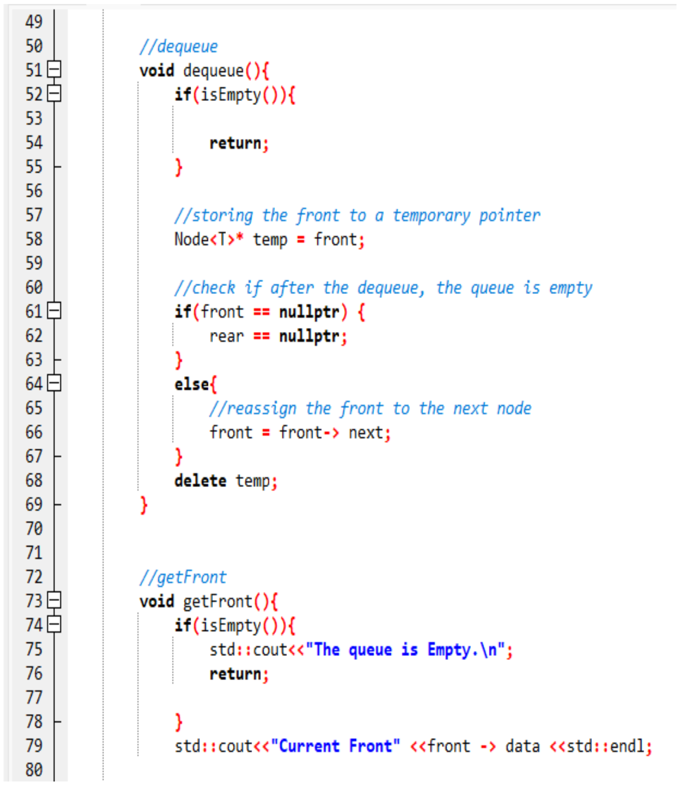
```cpp
#include <iostream>
#include "Q.h"


int main() {
    Queue <std::string> CPE21S4;

    CPE21S4.enqueue("Franciss");
    CPE21S4.enqueue("Jason!!!!");
    CPE21S4.enqueue("CURWIN");
    CPE21S4.enqueue("ABILA");
    CPE21S4.enqueue("Dano");
    CPE21S4.getFront();

    CPE21S4.dequeue();
    CPE21S4.getFront();
    CPE21S4.getrear();
    CPE21S4.display();

    return 0;
}
```

Q.h file:

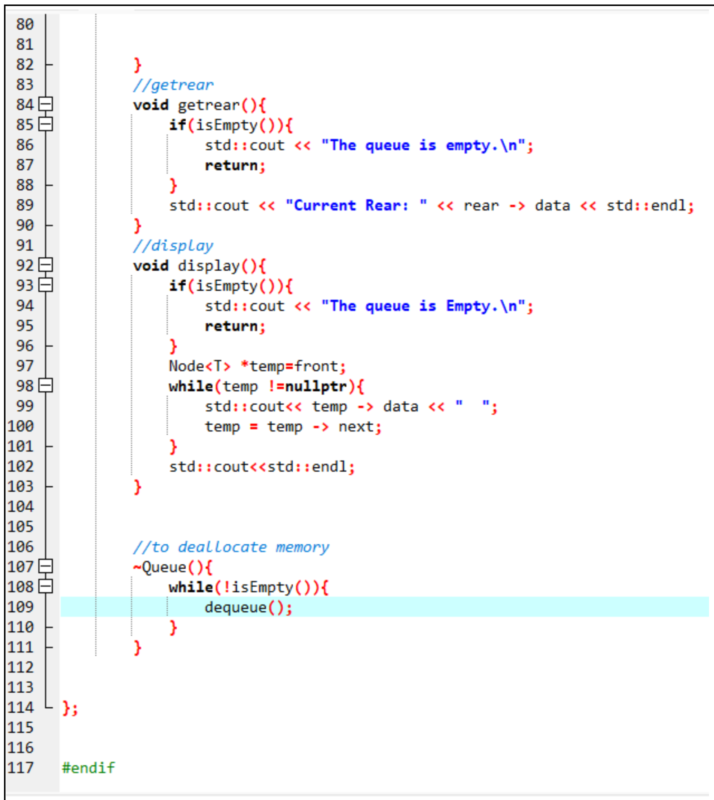


C:\Users\TIPQC\Downloads\Q.h - Dev-C++ 5.10
File Edit Search View Project Execute Tools AStyle Window Help
(globals)
Project Classes Debug
Qmain.cpp [*] Q.h

```cpp
#ifndef Q_H
#define Q_H
#include <iostream>

template<typename T>
class Node{
    public:
        T data;
        Node* next;

        Node(T new_data){
            data = new_data;
            next = nullptr;
        }

};

template<typename T>
class Queue{
    private:
        Node<T> *front;
        Node<T> *rear;

    public:
        //create an empty queue
        Queue(){
            front = rear = nullptr;
            std::cout<<"A queue has been created.\n";
        }

        //isEmpty
        bool isEmpty(){
            return front == nullptr;
        }
```

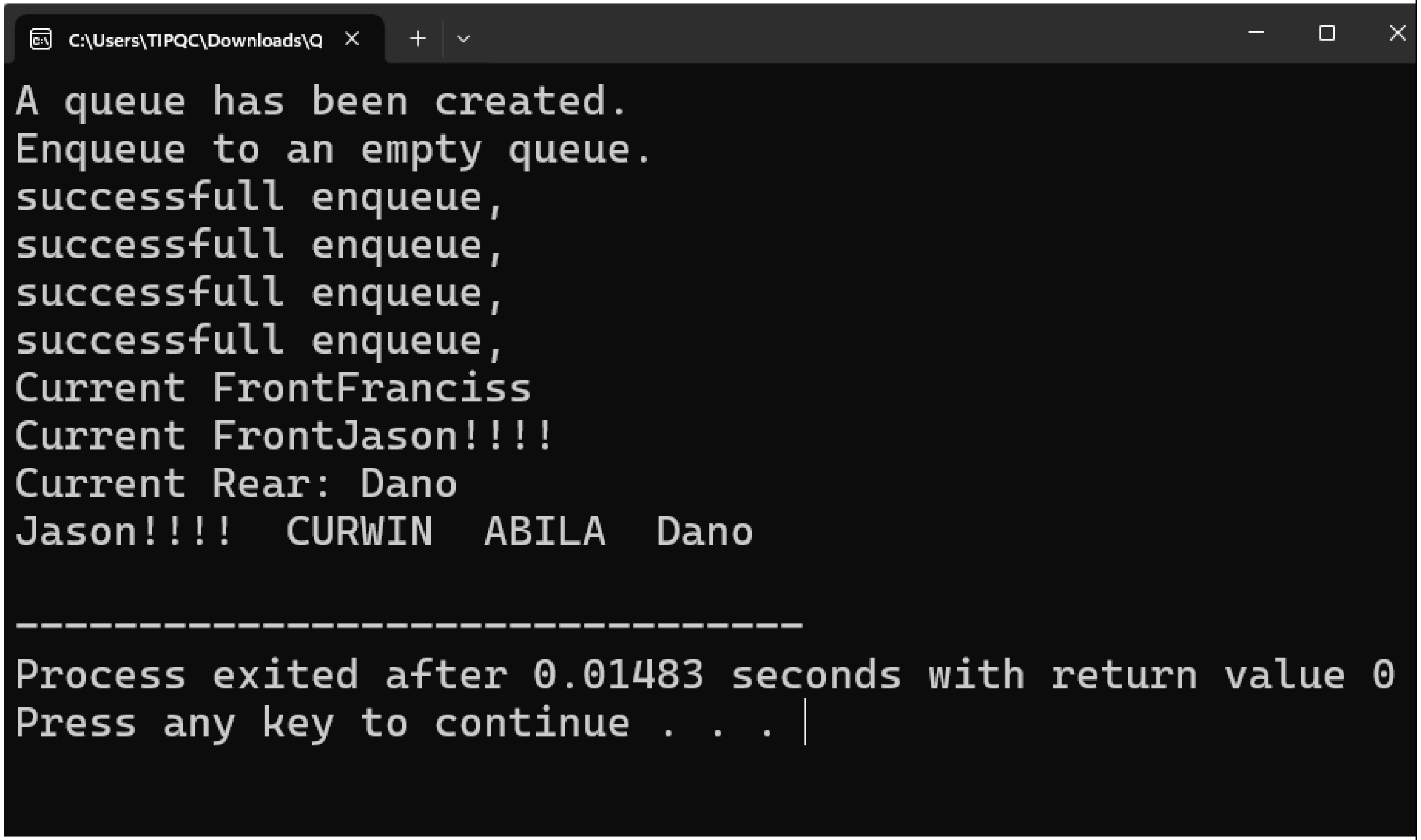


```cpp

        //dequeue
        void dequeue(){
            if(isEmpty()){

                return;
            }

            //storing the front to a temporary pointer
            Node<T>* temp = front;

            //check if after the dequeue, the queue is empty
            if(front == nullptr) {
                rear == nullptr;
            }
            else{
                //reassign the front to the next node
                front = front-> next;
            }
            delete temp;
        }


        //getFront
        void getFront(){
            if(isEmpty()){
                std::cout<<"The queue is Empty.\n";
                return;

            }
            std::cout<<"Current Front" <<front -> data <<std::endl;

```



```cpp


        }
        //getrear
        void getrear(){
            if(isEmpty()){
                std::cout << "The queue is empty.\n";
                return;
            }
            std::cout << "Current Rear: " << rear -> data << std::endl;
        }
        //display
        void display(){
            if(isEmpty()){
                std::cout << "The queue is Empty.\n";
                return;
            }
            Node<T> *temp=front;
            while(temp !=nullptr){
                std::cout<< temp -> data << "  ";
                temp = temp -> next;
            }
            std::cout<<std::endl;
        }


        //to deallocate memory
        ~Queue(){
            while(!isEmpty()){
                dequeue();
            }
        }


};


#endif
```

Output:



```
A queue has been created.
Enqueue to an empty queue.
successfull enqueue,
successfull enqueue,
successfull enqueue,
successfull enqueue,
Current FrontFrancis
Current FrontJason!!!!
Current Rear: Dano
Jason!!!!  CURWIN  ABILA  Dano

--------------------------------
Process exited after 0.01483 seconds with return value 0
Press any key to continue . . .
```

7. Supplementary Activity

8. Conclusion:

In this laboratory activity we learned the how to create a queue. We always need to check if the data is empty because if we don’t check the data if its empty the program will faced an error. And I learned how to used an temporary pointer and how to reassign the front node to the rear node. In my general though about this im still get confused a little bit that’s why I need to read, practice and understand more about this activity so next I will not get confused on how to created a queue but still in this day I learned a lot today.

9. Assessment Rubric