```
// Project: Linked List Implementation of Queue
// Author:  Niko Solihin

#include <iostream>
#include <string>
#include <sstream>
#include "IntQueue.h"
#include "IntQueue.cpp"

using namespace std;

int main () {

    // declare needed objects & variables
    IntQueue myQueue;
    int choice, value, frontItem;

    // initial message
    cout << "\nQueue Operations:";

    // loop until exit
    while (choice != 5)
    {
        // allow the user to decide what one wants to do
        cout << "\n\t1. Enqueue\n";
        cout << "\t2. Dequeue\n";
        cout << "\t3. Display Queue Length\n";
        cout << "\t4. Display Queue Contents\n";
        cout << "\t5. Exit\n\n";
        cout << "Enter your choice: ";
        cin >> choice;

        switch (choice)
        {
            case 1:
                // enqueue
                cout << "Enter a value: ";
                cin >> value;
                if ( myQueue.enqueue(value) )
                {
                    cout << "-----------------------------------------------------" << endl;
                    cout << "Succesfully enqueued " << value << " into the queue." << endl;
                    cout << "-----------------------------------------------------" << endl;
                };
                break;
            case 2:
                // dequeue
                if ( myQueue.dequeue(frontItem) )
                {
                    cout << "-----------------------------------------------------"     << endl;
                    cout << "Succesfully dequeued " << frontItem << " from the queue." << endl;
                    cout << "-----------------------------------------------------"     << endl;
                }
                // empty queue error
                else
                {
                    cout << "-----------------------------------------------------" << endl;
                    cout << "Empty queue, cannot dequeue." << endl;
                    cout << "-----------------------------------------------------" << endl;
                }
```

```
                break;
            case 3:
                // queue length
                    cout << "-----------------------------------------------------" << endl;
                    cout << "Currently there are " << myQueue.length() << " elements in the queue." << endl;
                    cout << "-----------------------------------------------------" << endl;
                break;
            case 4:
                // queue contents
                    cout << "-----------------------------------------------------" << endl;
                    cout << "Queue contains " << myQueue.display() << "." << endl;
                    cout << "-----------------------------------------------------" << endl;
                break;
            case 5:
                // exit
                cout << "Terminating..." << endl;
                break;
            default:
                cout << "Please enter a valid selection!" << endl;
        }
    }
    return 0;
}
```