

```
// QueueLinked.h

#include <stdexcept>
#include <iostream>

using namespace std;

#include "Queue.h"

template <typename DataType>
class QueueLinked : public Queue<DataType> {
public:
    QueueLinked(int maxNumber = Queue<DataType>::MAX_QUEUE_SIZE);
    QueueLinked(const QueueLinked& other);
    QueueLinked& operator=(const QueueLinked& other);
    ~QueueLinked();

    void enqueue(const DataType& newDataItem) throw (logic_error);
    DataType dequeue() throw (logic_error);

    void clear();

    bool isEmpty() const;
    bool isFull() const;

    // Programming Exercise 2
    void putFront(const DataType& newDataItem) throw (logic_error);
    DataType getRear() throw (logic_error);

    // Programming Exercise 3
    int getLength() const;

    void showStructure() const;

private:
    class QueueNode {
    public:
        QueueNode(const DataType& nodeData, QueueNode* nextPtr);

        DataType dataItem;
        QueueNode* next;
    };

    QueueNode* front;
    QueueNode* back;
};
```