D1265154 曾郁珊 Mina assignment4 report

In this report, I'll share my journey of programming and developing a queue data structure implementation in C++. The assignment tasked me with implementing enqueue, dequeue, and other operations on a queue, along with creating a driver program to test the functionality.

Implementation:

I began by creating a Node class, representing individual elements of the queue. Each node contained an integer element and pointers to the next and previous nodes.

The core of the assignment lay in implementing the IQueue class, providing various queue operations. These included enqueue, dequeue, headElem, isEmpty, getSize, getHead, getTail, and printHeadToTail.

In the enqueue operation, I inserted an element at the end of the queue. I dynamically created a new node to hold the element. If the queue was empty, the new node became both the head and tail. If not, I appended the new node after the current tail node and updated the tail pointer.

Conversely, the dequeue operation removed and returned the element at the front of the queue. If the queue was empty, the operation returned 0 indicating failure. If not, I removed the head node, updated the head pointer to the next node, and deallocated memory for the old head node.

Testing:

To ensure the correctness of my implementation, I developed a driver program to thoroughly test the functionality of the queue operations. I conducted randomized tests with varying trial counts, each involving enqueue and dequeue operations on the queue. I verified the correctness of operations such as enqueue, dequeue, isEmpty, getSize, and printing from head to tail through the driver program.

Challenges Faced:

Several challenges emerged during implementation, particularly regarding proper memory management with dynamic memory allocation and deallocation in enqueue and dequeue operations. Handling cases where the queue was empty or contained only one element in the dequeue operation required special attention. Debugging and ensuring proper traversal of the queue while printing its contents were also notable challenges.