Assignment4_D1228803 Adeline Chen 陳宣妤 Report

Programming Assignment 4: Queues Using Double-Linked Linear Lists

How I develop my assignment solution

The assignment use the project in Dev C which combines the file of .cpp .h and .dev

In this assignment, I discussed the implementation of IQueue data structure in C++. The IQueue data structure is designed to store integers in first in first out, allowing for operations such as enqueueing and dequeuing

The IQueue class manages the queue operations and maintains pointers to the head and tail of the queue. Such as enqueue, dequeue, headElem, isEmpty, getSize, getHead, getTail, and printHeadToTail. In enqueue operation I adds an integer element to the end of the queue. In dequeue operation I removes and returns the element at the front of the queue. In head element check I returns the value of the element at the head of the queue. In empty check I checks whether the queue is empty. In size retrieval I returns the number of elements currently in the queue. In head and tail pointer retrieve, I returns pointers to the head and tail nodes, respectively. In printing from head to tail I prints the elements of the queue from head to tail.

The Node have represents individual elements in the queue. From each node contains an integer data element and pointers to the previous and next nodes in the queue. It includes constructors to initialize nodes with or without data elements.

The main program demonstrates the usage of the IQueue by performing a series of enqueue and dequeue operations. By generating random numbers of elements to enqueue and dequeue in each trial.