

Here's a README file with the necessary steps and details about the functions and how to run the program.

Amei Ayuel
CS-303 Assignment 3

Overview

This program demonstrates solutions to the three tasks outlined in Assignment 3:

1. Queue Implementation (Template Class). Implements a queue data structure with common operations several different types of functions to manipulate the structure.
2. Recursive Linear Search - A recursive function to find the last occurrence of a target element in a vector.
3. Linked List with Insertion Sort: Implements a linked list and sorts it using the insertion sort algorithm.

Features and Functions

1. Queue Implementation (Question 1)

Functions:

- void push(T value) : Adds an element to the rear of the queue.
- void pop(): Removes the front element from the queue.
- T front(): Retrieves the front element of the queue.
- bool empty(): Checks if the queue is empty.
- int size(): Returns the number of elements in the queue.
- void move_to_rear(): Moves the front element to the rear of the queue.
- void display(): Prints all elements in the queue to the console.

In the main for Question 1

- Adds 10 integers to the queue.
- Displays the queue elements.
- Moves the front element to the rear and displays the updated queue.

2. Recursive Linear Search (Question 2)

Description:

- A recursive function that finds the last occurrence of a target element in a vector.
- Base case: Stops when the index becomes negative (no occurrence) or when the target is found.

- Recursive case: Decrements the index to continue searching from the end.

In the main Question 2

- Searches for the last occurrence of a target value in a sample vector and prints the index if found.

3. Linked List with Insertion Sort (Question 3)

Functions:

- void push(int value): Adds a new node with the given value to the end of the list.
- void insertionSort(): Sorts the linked list using insertion sort.
- void display(): Prints all elements of the list to the console.

In the main for Question 3

- Adds multiple integers to the linked list.
- Displays the original list.
- Sorts the list using insertion sort and displays the sorted list.

How to Run the Program

1. Compile the Program:

Ensure you have both `Assignment3.h` and `main.cpp` in the same directory. Compile using the following command:

2. Expected Output

```
h X CS-303-Assignment-3.cpp Solution Explorer
Microsoft Visual Studio Debug Console
Question 1: Queue Operations
Queue elements: 1 2 3 4 5 6 7 8 9 10
After moving front element to rear: 2 3 4 5 6 7 8 9 10 1

Question 2: Recursive Last Occurrence Search
Last occurrence of 3 is at index 6

Question 3: Linked List Insertion Sort
Original list: 3 1 4 2 5
Sorted list: 1 2 3 4 5

C:\Users\Amei\source\repos\CS-303-Assignment-3\x64\Debug\CS-303-Assignment-3.exe (process 22488) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```