CS2413: Assignment 2

Total Points: 100

Primary objectives of this assignment

Be able to define class, play with private member variable through public member function and class pointer.

Be able to dynamically construct a linked list through input, with each element being a class object.

Be able to implement deletion and insertion on a linked list.

Be able to implement binary search algorithm based on a sorted sequence stored in a linked list.

Project Overview

In this assignment, we will input a sorted sequence of integers (of arbitrary size and values), create a doubly linked list to store it, and selectively perform the following tasks

- apply binary search to find a search key in the list and return its position or failure

- delete the key from the list (if it was in the list), or insert the key in the list (if it was not in the list). After deletion or insertion, the list should remain sorted.

Program Input and Output

Our program should take three sets of input:

(1) A task selection integer

- if 0, perform binary search and output position of the key or -1 if the key is not found.[1]

- if 1, perform key deletion or insertion (automatically decide), and output the updated sequence.

(2) A search key

(3) The input sequence of integers, ended with a non-integer character – we suggest using 's'.

Figure 1 shows an example input format. In the first line, the first integer selects task 0, and the second integer is search key 11. In the second line, we input an arbitrary sequence of integers, and end it with 's'.

Given inputs in Figure 1, the correct output should be 6.

<div align="center">

0 11

1 2 4 7 8 9 11 15 s

</div>

Fig. 1. Example Input with Task Selection Integer being 0

---

[1] Position index should start from 0, i.e., 1st element is 0, 2nd element is 1, . . .

Figure 2 shows another example input, and this time we select task 1. Based on these inputs, the correct output should be 1 2 4 7 8 9 15, because key 11 is deleted from the sequence.

<div align="center">

1 11
1 2 4 7 8 9 11 15 s

</div>

Fig. 2. Example Input with Task Selection Integer being 1

Additional Requirements and Tips

You need to implement this assignment using the following data structures. Otherwise, your submitted code will be voided, even if it gives the right input and output on gradescope.

- The input sequence must be stored using a doubly linked list, where each element stores an integer.

- Each element must be implemented as an object of a class 'Course', which contains the following three private member variables:

    - an integer 'capacity', used to store one integer in the input sequence

    - a Course pointer 'ptr_prev', used to store address of the previous object in the list

    - a Course pointer 'ptr_next', used to store address of the next object in the list

Figure 3 shows an example diagram of the doubly linked list. Note a pointer stores the address of a class object, not its member variable. Also, the head and tail elements will each have a pointer being NULL.
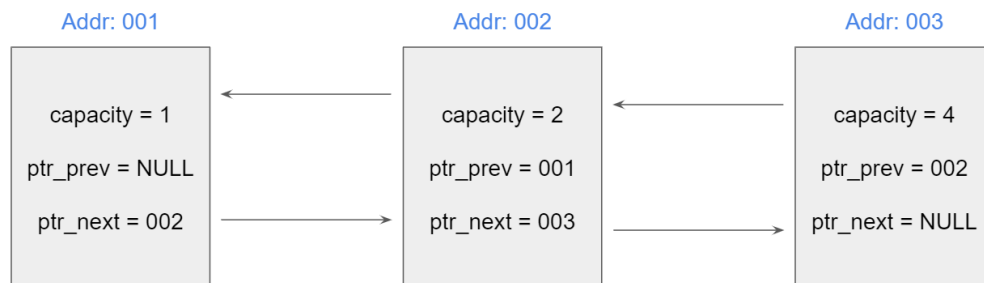
Fig. 3. Diagram of doubly linked list.

Finally, be aware that your input sequence should be of arbitrary size, and you probably need to use dynamic memory allocation. (We will cover these practices in the following lectures.)

Please name your submitted code as `cs2413_hw2.cpp`.

Rubrics

- Binary search based on doubly linked list: 40 points.

- Deletion or insertion on doubly linked list: 50 points.

- Documentation: 10 points. (Comments in your submitted code)