BIL 212
Summer 2021

Lab 3

Part 1

Implement swap and sort methods similar to the ones you implemented in Lab 1 using PositionalList/Position. You still cannot modify values inside the nodes but this time you can create new nodes. You will implement these methods outside of your list class but you can modify the list class to implement Positions. Create SwapSortDriver, construct a new list with random values in the main method of this class and test these methods on this list. You cannot access Nodes directly in your driver code, you must use the Position to access and modify them.

Part 2

In this question, you are asked to determine if a given DoublyLinkedList (of chars) is palindrome or not using 2 different methods:

- Use only get(int index) method while accessing nodes of the list. You cannot use next or previous (except inside the get method itself).

- Use an iterator and an empty stack (and the size of the list) to find if the given DoublyLinkedList is a palindrome.

Implement these methods outside of your list class, name it PalindromeDriver, this file will also contain the main method. Construct large lists (both palindrome and not palindrome) with varying sizes (like 20000, 40000 and 80000 etc.) and analyze the running times of these methods, similar to what you've done in Lab 2. Explain the difference and write down your explanation in a file named palindrome-analyze.txt.

You can write your own code or use the classes posted on Piazza, but you cannot use the Java's built-in data structures. Make sure to check your section before attending. Please upload all your .java and .txt files without compression to uzak.etu.edu.tr before 10.06.2021 Thursday 11:00.