Algorithms and Data Structures: Linked lists and Sorting

Exercise - 2

Dr Nagarajan Ganapathy

Version 1

Datum: 25.09.2023

Write the most efficient algorithm for the following problems in C++ programming language and mention the Time and Space Complexity of your algorithms in the comments (at the end):

1. **Create your own linked list using class or struct, implement the following functions: [10 pts]**
   - **Insertion**
   - **Deletion**
   - **Search**
   - **Display**
   - **Head**
2. **Create your own doubly linked lists using class or struct, implement the following functions: [10 pts]**
   - **Insertion**
   - **Deletion**
   - **Search**
   - **Display**
   - **Head**
   - **Tail**
3. **Harry is fond of palindromes. He tries to find palindromes everywhere. He has now been given a list and he is curious to find out whether it is a palindrome or not. So now help Harry verify whether a list is palindrome or not. [15 pts]**

   Testcases:

a). List: 1->4->2->2->1; Output: False.

b). List: 1->2->3->2->1; Output: True.

c). List: (Empty List); Output: True.

d). List: 1->3->3->0; Output: False.

4. **Larry is a classmate of Harry, he has been given a linked list of integers and his instructor has asked to compute the middle element and extreme element of the list but in a single iteration and without using the length of the list. Larry got confused about how it is possible. So now help Larry in finding the middle element of the linked list. [15 pts]**

Testcases:

a). List: 1->2->3->2->1; Output: 1,3,3.

b). List: 1; Output: 1.

c). List: (Empty List); Output: -1. d).

List: 1->2->3->4; Output: 1,2,4.

5. **Mr X has stored a path of all the visible numbers he has seen while traveling from IIT Hyderabad to IIT Madras in Doubly Linked Lists (DLL). He has given this DLL to his student and asked him to predict the DLL he would have made if he would have traveled from IIT Madras to IIT Hyderabad. You being a friend of his student can you evaluate the DLL for him. [15 pts]**

Testcases:

a). DLL: 2 ⇌ -3 ⇌ 0 ⇌ 2

Output: 2 ⇌ 0 ⇌ -3 ⇌ 2

b). DLL: 9 ⇌ 3 ⇌ 5 ⇌ 7 ⇌ 2

Output: 2 ⇌ 7 ⇌ 5 ⇌ 3 ⇌ 9

6. **Implement the following algorithms: [20 pts]**
   i. **Bubble Sort**
   ii. **Insertion Sort**
   iii. **Selection Sort**

iv. **Merge Sort**

v. **Quick Sort**

7. **Larry has been given a list. But he hates to have duplicate characters in them. So now update the linked list such that all duplicate characters from the lists are removed and the list is also sorted. [15 pts]**

Testcases:

a). List: 1->8->4->3->1->11->3->1

Output: 1->3->4->8->11.

b). List: 6->6->5->6

Output: 5->6.

c). List: -1

Output: -1.

d). List: 1->4->5->2->3

Output: 1->2->3->4->5.