Assignment 7

Samundra karki

April 2019

Problem 7.2

(a)

Pseudocode of insitu-algorithm revering the lnked list.

```
reverseList(list)
if list == NULL then
  exit the program
else
  create three pointer *next, *previous and *current of same datatype.
  previous = next = NULL
  current = list
  while current != NULL do
    next = current->next
    prev = current
    current = next
  end while
  list = prev
```

This algorithm is in-situ because it doesnot creates new linked-list in order to reverse it i.e. it does not need whole new array equal to the size of the list. It just creats three node of same type, in order to reverse i.e. O(1) for space complexity. This is why it is called in-situ algorothm.

(b)

Time complexity to convert binarySearchTree to Linked list.

Here we first recursively queried all the right children of the tree then left children of the tree. time take to add at the front of the list is O(1). So time taken is to recursively query all the elements in the tree where the number of the element is 'n'. So, we can conclude that the time complexity is O(n).

(c)

Time complexity to convert binarySearchTree to Linked list.

We know that the time taken to insert a element in a tree is O(h) = O(lgn). Also, we are inserting the elements in the loop. So, total time taken can be computed as O(nlogn).
We, also know that the time taken to search a element in a binary search tree is O(logn). We for a sorted we need to search in a loop for the element until the last element. So the search time-complexity of binary search tree is less than in a sorted linked list.
Codes can be found in the zip file.
Reference:- 7.2(b) Implentation of binary search tree was seen in several sites from Internet.