Lab 7

# Problem 1

Algorithm reverse(S)  
 Input A String S  
 Output A String with each element is reverted  
  
 stack <- new Stack  
 for i <- 0 to S.length - 1 do  
 stack.push(S.charAt(i))  
  
 reverseStr <- ""  
  
 while !stack.isEmpty() do  
 reverseStr <- reverseStr + stack.pop()  
  
 reverseStrArr = reverseStr.split(" ")  
  
 result <- ""  
  
 for i <- reverseStrArr.length - 1 to 0 do  
 result <- result + reverseStrArr[i] + " "  
  
 return result

# Problem 2

The result after running test

49 ms -> MergeSort

121 ms -> MyBST

255 ms -> MergeSortPlus

2331 ms -> BubbleSort2

2489 ms -> BubbleSort

2928 ms -> BubbleSort1

The MyBST sort is run faster than all other sorting algorithm but MergeSort

The implement of MyBST as following

public int[] sort(int[] arr) {  
 for (int i = 0; i < arr.length; i++) { // O(n)  
 insert(arr[i]);  
 }  
   
 countNode(root); // O(log n)  
 sortedArr = new int[count]; // O(1)  
   
 sort(root); // O(log n)  
 return sortedArr;  
}

private void sort(Node t) {  
 if (t != null) {  
 sort(t.left);  
 sortedArr[pos++] = t.element;  
 sort(t.right);  
 }  
}

T(n) = O(n) + O(log n) + O(1) + O(log n) = O(n log n)

# Problem 3

|  |  |
| --- | --- |
| 1 | Yes |
| 2 | No |
| 3 | Yes |
| 4 | No |
| 5 | No |
| 6 | No |
| 7 | Yes |

# Problem 4

|  |  |
| --- | --- |
| 1 | No |
| 2 | Yes |
| 3 | No |
| 4 | Yes |
| 5 | Yes |
| 6 | No |
| 7 | No |