EL9343 Homework 2

All problem/exercise numbers are for the third edition of CLRS text book

.

---

1. For the maximum subarray problem, if we use divide-conquer, but instead of dividing the array into two halves, we equally divide it into three segments, how should the algorithm be modified? what is the running time of the new algorithm?

2. Using Figure 2.4 as a model, illustrate the operation of insertion sort and merge sort on the array $[10, 1, 6, 20, 16, 8, 33, 15]$

3. Using Figure 6.4 as a model, illustrate the operation of HEAPSORT on the array $[30, 18, 26, 3, 37, 5, 16, 29]$

4. Similar to Figure 7.1, show the steps of Hoare's Partition on the array of

$$[1, 7, 9, 3, 100, 13, 12, 5, 14]$$

5. 5.1 Merge two **sorted** linked lists, and return it as a new list. The new list should be made by splicing together the nodes of the first two lists.
For example:
Input: $1 \to 2 \to 5$, $3 \to 6$.
Output: $1 \to 2 \to 3 \to 5 \to 6$.

Implement pseudo code for function Merge-Two(l_1, l_2). Type of l_1 and l_2 are ListNode. (Definition of linked list can be found at https://en.wikipedia.org/wiki/Linked_list).
The format of the pseudo code is suggested to be similar to Max-Heapify(A,i) shown in Chapter 6.2 on page 154 of CLRS text book.

5.2 What if there are k sorted linked lists. Please design an algorithm to combine the k linked lists.
Input:
[
$1 \to 4 \to 5$,
$1 \to 3 \to 4$,
$2 \to 6$
]
Output: $1 \to 1 \to 2 \to 3 \to 4 \to 4 \to 5 \to 6$.

Implement pseudo code for function Merge-K(l) where l is $[l_1, l_2, ..., l_k]$. l_1 to l_k are linked lists.

5.3 Assume the average length of each linked list is n, what is the complexity of your solution of (6.2)? Can you improve it to $knlog(k)$.
(Hint: Use priority queue assuming all operations related to priority queue (page 38 and 39 of lecture slides) are available.)

5.4 Can you use divide and conquer method to solve question (6.2).
(Hint: Re-use solution of (6.1)).