INTRODUCTION TO ALGORITHMS (CELEN086)
EXTRA PRACTICE PROBLEMS (4)

TOPIC: *Sorting Lists, Binary Trees*

University of Nottingham
UK | CHINA | MALAYSIA

Note: ALL algorithms must come with proper headings. Also, <u>trace your algorithms!!</u>

1. Write a recursive algorithm **`find(x,L)`** that takes a list and an element x and returns the position of that element in the list.. For example: **`find(11,[10,11,12,4,5,6])=2,  find(3,[1,6,4,5,0,9])=0`**

2. Apply the divide & conquer scheme of **mergesort** to the following unsorted lists and obtain a sorted list at the end. You must show each step clearly, i.e. *split – sort – merge.*

   i. **`L=[10,11,12,0,3,4,2,1,5]`**
   ii. **`L=[10,9,8,7,6,5,4,3,2,1]`**

3. Apply the divide & conquer scheme of **quicksort** to the given unsorted lists above and obtain a sorted list at the end. You must show each step clearly, i.e. *partition – sort – merge.*

4. Apply the scheme of insertion sort to the unsorted lists given in question 2 and obtain a sorted list at the end. You must show each step clearly, i.e. *insert – sort.*

5. Write a recursive algorithm **`plus(x,binT)`** that takes a binary tree and a value x and then adds x to every node value, hence returns a new binary tree with updated node values. Trace your algorithm for **`plus(10,T)`** where T is the following binary tree:

   `T= node(node(leaf,40,node(node(leaf,15,leaf),30,leaf)),10,node(node(leaf,20,leaf),0,leaf))`

6. Write a recursive algorithm **`sumNodes(binT)`** that takes a binary tree and returns the sum of all the node values. Trace your algorithm for the binary tree given in question 5.