國立台灣科技大學九十五學年度碩士班招生試題

系所組別： 電機工程系碩士班丙一組

科　　目： 資料結構

【總分100分】

1. Indicate both the stability and time complexity for each of the following sorting methods:
   (a) Insert sort (4%)
   (b) Quick sort (4%)
   (c) Merge sort (4%)
   (d) Heap sort (4%)

2. Suppose we have the following key values: 7, 9, 16, 30, 49, 82, 5, 33, 31, 6, 2, 1.
   (a) Write out the max heap after each value is inserted into the heap. (4%)
   (b) Write out the min heap after each value is inserted into the heap. (4%)
   (c) Write out the result of ascending heap. (4%)
   (d) Write out the resultant max heap after deleting 9 and 49 from the max heap obtained in (a) above. (4%)
   (e) Write out the resultant min heap after deleting 9 and 49 from the min heap obtained in (b) above. (4%)

3. What would be the contents of queue Q after the following code is executed and the following data are entered? (14%)

```
1. Q = createqueue
2. S= createstack
3. loop (not end of file)
    1. read number
    2. if (number not 0)
        1. pushstack (S, number)
    3. else
        1. popstack (S, x)
        2. popstack (S, x)
        3. loop (not empty S)
            1. popstack (S, x)
            2. enterqueue (Q, x)
```

The data entered are: 5, 7, 12, 4, 0, 4, 6, 8, 67, 34, 23, 5, 0, 44, 33, 22, 6, 0.

國立台灣科技大學九十五學年度碩士班招生試題

系所組別： 電機工程系碩士班丙一組

科　　目： 資料結構

4. A ***B-tree of order n*** is a balanced order-*n* multiway search tree in which each nonroot node contains at least (n-1)/2 keys. The following figure shows a subtree of a B-tree of order 5. Give the subtrees after inserting 380, 530, and 510, respectively. (18%)

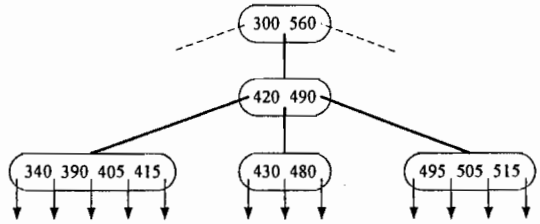

Figure of problem 4

5. For the following binary tree, give its related ***infix***, ***prefix*** and ***postfix*** by using ***traversal***. (12%)

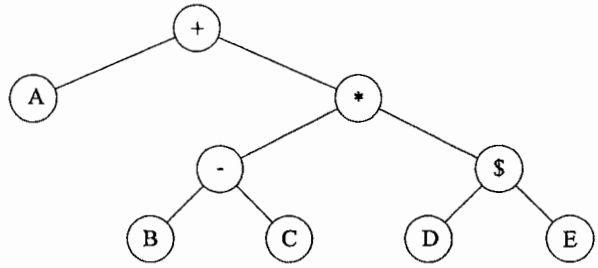

Figure of problem 5

6. Please construct a minimum cost spanning tree for the undirected connected graph ***G*** shown below, where ***V(G)***, ***E(G)***, and ***W(G)*** are the sets of vertices, edges, and weights, respectively.

   (a) Kruskal's algorithm without any constrain. (10%)

   (b) Kruskal's algorithm with the constrain that a branch contains at most two links. (10%)

   (Note that mark the sequence number beside each link)

   The graph ***G*** is given by:

   V(G) = {0, 1, 2, 3, 4, 5, and 6};

   E(G) = {(0,1), (0,3), (0,4), (1,2), (2,3) (2,6), (3,5), (3,6), (4,5), (5,6)} and the corresponding weight

   W(G) = {16, 12, 18, 10, 14, 20, 28, 22, 26, 24}

126