國立臺北大學 103 學年度碩士班一般入學考試試題

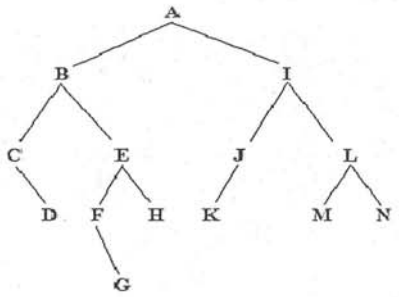
系（所）組別：電機工程學系乙組(電腦工程組)

科　　目：資料結構

☐可 ☑不可使用計算機

1. Let $F(n) = 2^6 + n^4 \log(n^3) + n^{1.6} + 100\, n^4$,
   $G(n) = 150n^4 + 10n^3 + 5n^2$,
   $H(n) = 150n^4 + 2n^2$.
   Write down the best asymptotic ("big-O") characterization of the following functions:
   (a) (5 %) F(n)
   (b) (5 %) G(n) + H(n)
   (c) (5 %) G(n) - H(n)
   (d) (5 %) G(n) · H(n)

2. Here is a small binary tree:



   Write the order of the nodes visited in:
   (a) (5 %) A pre-order traversal
   (b) (5 %) Ain-order traversal
   (c) (5 %) A post-order traversal

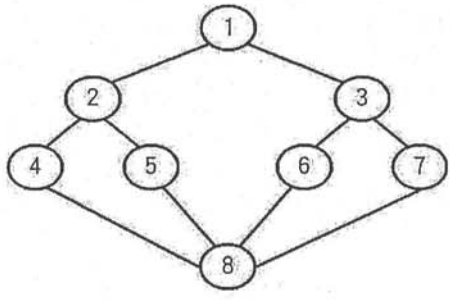
3. Consider a hash table of size 13 storing entries with integer keys. Suppose the hash function is h(k) = k mod 13. Insert, in the given order, entries with keys
   10, 3, 6, 16, 17, 19
   into the hash table using:
   (a) (5 %) Linear probing to resolve collisions. Show all the work and fill in the blanks below.

0	1	2	3	4	5	6	7	8	9	10	11	12
												-

   (b) (5 %) Double hashing to resolve collisions with secondary hash function
   h(k) = 7 - (k mod 7)
   Show all the workand fill in the blanks below.

0	1	2	3	4	5	6	7	8	9	10	11	12
-												

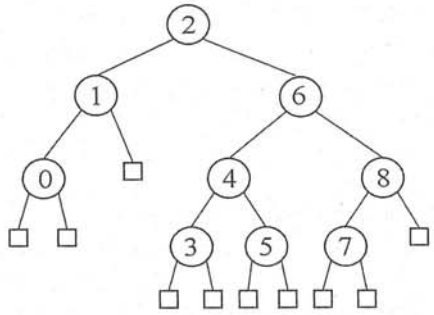
4. Given the following graph:



(a) (10%) Perform a depth-first traversal of the graph shown above, starting with vertex1.List the vertices in the order in which they are visited.Draw the depth -first search tree which results from running depth -first search on the graph.

(b) (10 %) Perform a breadth-first traversal of the graph shown above, starting with vertex1.List the vertices in the order in which they are visited.Draw the breadth-first search tree which results from running breadth-first search on the graph.

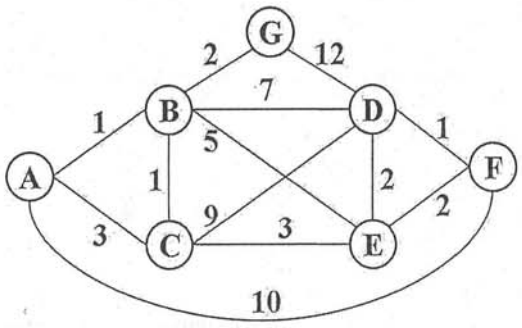
5. Consider the following AVL tree.



(a) (10%) Insert an additional key value 6 into the tree and re-balance if needed. Draw the final tree and all intermediate trees that you need.

(b) (10%) Remove a key value 1 from the original tree and re-balance if needed. Draw the final tree and all intermediate trees that you need.

6. Consider the following undirected, weighted graph:



(a) (10%) Step through Dijkstra's algorithm to calculate the single-source shortest paths from nodeA to every othernodes.

(b) (5%) Indicate the lowest-cast path from node A to node F.