Data Structures
Sample Final Exam

1. (32 pts) (Midterm Exam Problem) Mark by T(=True) or F(=False) each of the followings. You don't need to prove it.

   (1) (4 pts) Suppose T is a ***proper binary tree*** and let e and i denote the numbers of external and internal nodes, respectively. Then $e = i + 1$.

   (2) (4 pts) The height of a binary search tree with n keys is $O(\log n)$ in worst case.

   (3) (4 pts) Suppose a binary search tree T is established by inserting a series of entries. Then, the order of the entries does not impact the shape of the resulting binary search tree.

   (4) (4 pts) A graph T with n vertices is a ***tree*** if it is connected and has no cycles.

   (5) (4 pts) The maximum possible number of edges in a directed graph is $\frac{n(n+1)}{2}$.

   (6) (4 pts) A heap can be constructed in linear time.

   (7) (4 pts) Let T be a B-tree of order 5, the possible degrees of internal nodes (except the root) are 2, 3, 4, and 5.

   (8) (4 pts) When inserting a node to an AVL tree, the height-balance property may not be restored with at most one restructuring.

2. (9 pts) Mark by T(=true) or F(=False) each of the following statements. You don't need to prove it. Let G be a directed graph and Q an undirected graph. We use m and n to denote the numbers of edges and vertices, respectively.

   (1) (3 pts) If G is strongly connected, then $m \geq 2(n-1)$.

   (2) (3 pts) For Q, the maximum possible value of m is $n(n+1)$.

   (3) (3 pts) If Q is connected and $m = n - 1$ edges, then Q has no cycle.

3. (9 pts) Recall the skip list data structure. Suppose that we use only two levels, *i.e.*, two linked lists, for n elements. Each element is in linked list S_0 and some elements are also in the other linked list S_1. Please answer each of the following statements.

   (1) (3 pts) (True or False) The search cost can be minimized to $O(\sqrt{n})$ which is better than $O(\log n)$.

   (2) (3 pts) Please draw the skip list structure when the minimized search cost occurs.

   (3) (3 pts) (True or False) When there are $\log n$ linked lists, the skip list becomes like a binary search tree.

4. (15 pts) Indicate whether the following statements are true or false:

   (1) (5 pts) If e is a minimum-weight edge in a connected weighted graph, it must be among edges of each minimum spanning tree of the graph.

   (2) (5 pts) If edge weights of a connected weighted graph are all distinct, the graph must have exactly one minimum spanning tree.

(3) (5 pts) If edge weights of a connected weighted graph are not all distinct, the graph must have more than one minimum spanning tree.

5. (10 pts) Draw the 11-entry hash table that results from using the hash function, $h(i) = (2i + 5) \bmod 11$, to hash the keys 34, 22, 2, 88, 23, 72, 11, 39, 20, 16, and 5, assuming collisions are handled by the following approaches respectively.

   (a) (5 pts) *chaining*

   (b) (5 pts) *double hashing*, second hash function: $h'(k) = 7 - (k \mod 7)$

6. (10 pts) Please answer each of the following problems shortly and concisely.

   (1) (5 pts) Solve the continuous-knapsack problem for the following weights(w_i), profits(p_i), and knapsack capacity(M). Please show your work **step by step**.

$$w_1 = 12, w_2 = 15, w_3 = 20, w_4 = 15;$$

$$p_1 = 4, p_2 = 3, p_3 = 6, p_4 = 8;$$

$$M = 50.$$

   (2) (5 pts) A *k-ary* tree is a tree of which each node has at most k children. What is the maximum number of node is a k-ary tree of height h? Prove your answer

7. (10 pts) Consider the red-black tree in Figure **??**.

   (1) (5 pts) Please construct a corresponding (2,4)-tree.

   (2) (5 pts) Use the (2,4)-tree derived in (1) and perform the following sequence of insertions: `insert(63)`, `insert(28)` and `insert(18)`. `insert(63)` 、 `insert(28)` 、 `insert(18)`

You need to mark the actions taken at each step. No partial credits will be given to a wrong input for each subproblem.

8. (15 pts) Consider the edge-weighted connected graph $G = (V, E)$ in Figure 1 where V is the vertex set and E is the edge set of G respectively.

Figure 1: The edge-weighted connected graph G for Problem 8

   (1) (5 pts) Please use the *adjacency matrix* to represent G in Figure 1.

   (2) (5 pts) Please find a minimum spanning tree of G by *Prim*'s algorithm **starting from vertex** d. Show your work step by step.

   (3) (5 pts) Please find a minimum spanning tree of G by *Kruskal*'s algorithm and show your work step by step.