Name: Covers chapter 1 – 8, 11 – 13, 22 – 23	CS3112 Spring 2019 Final Exam California State University, Los Angeles Instructor: Jungsoo Lim

I pledge by honor that I will not discuss the contents of this exam with anyone.
Signed by ______________________ Date ______________________

Part I. (30 pts)

1. (5 pts each) Prove or disprove each of the following statements.

   a. $f(n) = O(g(n))$ *implies* $g(n) = \Omega(f(n))$

   b. $f(n) = O(g(n))$ *and* $g(n) = \Omega(h(n))$ *implies* $f(n) = O(h(n))$

   c. $f(n) + o(f(n)) = \Omega(f(n))$

d. $f(n) = \theta(g(n))$ *implies* $2^{f(n)} = O\ (2^{f(n)})$

e. $f(n) = \theta(f(^{n}/_{2}))$

f. f(n) $=\Omega$(g(n)) implies g(n) $= \Theta$ (f (n)).

Part II (30)

2. (5 pts) Suppose you have an algorithm that operates on a set of data with n elements. If the recurrence formula that computes the time requirement for the algorithm is given by

$$T(n) = \begin{cases} 3T\left(\frac{5n}{8}\right) + D\sqrt[4]{n} & if\ n > 1 \\ C & if\ n = 1 \end{cases}$$

where D and C are constants, what is the solution of T(n)?

3. (5 pts) Suppose you have an algorithm that operates on a set of data with n elements. If the recurrence formula that computes the time requirement for the algorithm is given by

$$T(n) = \begin{cases} T\left(\frac{n}{2}\right) + T\left(\frac{n}{32}\right) + Dn & if\ n > 1 \\ C & if\ n = 1 \end{cases}$$

where D and C are constants, what is the solution of T(n)?

4. (5 pts) Suppose you have an algorithm that operates on a set of data with n elements. If the recurrence formula that computes the time requirement for the algorithm is given by

$$T(n) = \begin{cases} 9T\left(\frac{3n}{7} - 3\right) + Dn\sqrt[3]{n} & \text{if } n > 1 \\ C & \text{if } n = 1 \end{cases}$$

where D and C are constants, what is the solution of T(n)?

5. (5 pts) Suppose you have an algorithm that operates on a set of data with n elements. If the recurrence formula that computes the time requirement for the algorithm is given by

$$T(n) = \begin{cases} T(n-4) + Dn & \text{if } n > 1 \\ C & \text{if } n = 1 \end{cases}$$

where D and C are constants, what is the solution of T(n)?

6. (5 pts) Suppose you have an algorithm that operates on a set of data with n elements. If the recurrence formula that computes the time requirement for the algorithm is given by

$$T(n) = \begin{cases} 2T(n-2) + Dn & \text{if } n > 1 \\ C & \text{if } n = 1 \end{cases}$$

where D and C are constants, what is the solution of T(n)?

Part III (20 pts)

Please choose the best suitable data structure to insert/delete/search records for the following cases.

A. Max-Heap Tree
B. Min-Heap Tree
C. Red-black BST
D. Open-addressing hash table
E. Separate-chaining hash table
F. Perfect hashing table
G. None of the above

7. (5) Suppose you want to store priority queues to handle jobs with priorities, what data structure would work best for this case?

8. (5) Suppose you want to construct a compiler and need to store variables (or symbols) of a program, what data structure would work best for this case? Symbols are unique and they will not be changed throughout the lifetime of the program once an object code is created.

9. (5) Suppose you want to store over 100,000,000 customer records. The customer records will be constantly updated/purged/searched. What data structure would work best for this case?

10. (5) Suppose you need to develop a simulator to test the throughputs of 7G LTE, and you need to keep track of the events. When a timer expires, events are triggered accordingly. What data structure would work best to keep track of an event schedule?

Part IV (20 pts)

11. (20) Suppose a second-best minimum spanning tree G = (V, E) be an undirected, connected graph whose weight function is w: E → R, and |E| >= |V|, and all edge weights are distinct.
We define the second-best minimum spanning tree as follows. Let T be the set of all spanning trees of G, and let T' be a minimum spanning tree of G. Then, the second-best minimum spanning tree is a spanning tree T such that

$$min_{T'' \in T - \{T''\}}\{w(T'')\}$$

a. (10) Is minimum spanning tree (MST) unique if all edge weights are distinct?

Yes / No

Why? –

b. (10) Is the second-best MST unique if all edge weights are distinct?

Yes / No

Why? –