Student Name: _____   J#: _____

CSC 228 Data Structures and Algorithms, Spring 2019
Instructor: Dr. Natarajan Meghanathan

Quiz 6 and Quiz 7 (Take Home)   Max. Points: 50 each
Due on: April 12th @ 1 PM. <u>**Late submissions will NOT be accepted.**</u>
Print all the pages and answer in the space provided. Staple everything together and submit in class at the above time.
Clear label your answers for each of the sub parts (a) - (n), in the space provided.

Given an array of integers, do the following (SHOW ALL THE STEPS; just writing the final answer will get only ZERO).

Quiz 6

(a - 11 pts) Construct a max heap of the array. Show the initial essentially complete binary tree and the transformation of the binary tree to a max heap via the reheapify operations at the indices of the internal nodes (as shown in the slides).

(b - 11 pts) Sort the max heap version of the array obtained from (a) to obtain a sorted array of integers. Show the structural changes of the max heap in each iteration.

(c - 7 pts) Transform the max heap of (a) to a binary search tree.

(d - 7 pts) For the binary search tree obtained in (c), determine the average number of comparisons for a successful search and the average number of comparisons for an unsuccessful search.

(e - 7 pts) Use the sorted array of (b) to construct a binary search tree.

(f - 7 pts) For the binary search tree obtained in (e), determine the average number of comparisons for a successful search and the average number of comparisons for an unsuccessful search.

Quiz 7

(g - 6 pts) Construct a hash table of the given array using a hash function $H(K) = K \bmod 5$.

(h - 6 pts) For the hash table of (g), determine the average number of comparisons for a successful search and the worst case number of comparisons for an unsuccessful search.

(i - 9 pts) Consider the elements of the array assigned to you are known only one at a time. Construct a sequence of priority queues (as max heaps) with the insertion (enqueue) of one element at a time, as shown in the slides.

(j - 7 pts) Transform the binary search tree of (c) to a min heap.

(k - 6 pts) Transform the max heap of (a) to a min heap via reheapify operations at the internal nodes.

(l - 3 pts) Are the min heaps of (j) and (k) the same or not? What can you say about the distribution of the data in the min heaps of (j) and (k)?

(m - 7 pts) Determine whether the binary search trees of (c) and (e) are height-balanced or not?

(n - 6 pts) Write the pre-order, in-order and post-order traversal of the vertices in the binary search tree of (c).

1	Agnew, Dennis	[1, 25, 4, 4, 30, 29, 5, 25, 6, 4, 12, 19]
2	Barner, Brandon	[12, 21, 12, 26, 11, 13, 26, 30, 12, 2, 3, 13]
3	Bell, Jailen	[2, 22, 26, 22, 29, 6, 14, 22, 23, 20, 16, 18]
4	Brown, Brandon	[6, 14, 9, 12, 21, 22, 3, 10, 26, 27, 12, 27]
5	Butler, Perry	[7, 18, 9, 7, 14, 3, 30, 16, 2, 14, 22, 29]
6	Butler, Travis	[26, 12, 20, 27, 9, 10, 22, 22, 1, 3, 28, 28]
7	Coffee, Jhakorri	[28, 20, 12, 16, 18, 24, 1, 11, 20, 28, 28, 3]
8	Dixon, Kobe	[5, 10, 21, 14, 7, 18, 8, 16, 4, 10, 26, 23]
9	Felix Jr, Christopher	[17, 29, 22, 8, 20, 30, 18, 1, 7, 14, 21, 21]
10	Fisher, Christina	[21, 30, 21, 10, 4, 2, 18, 9, 12, 23, 16, 17]
11	Ford, Kalil-Dan	[1, 30, 21, 12, 23, 21, 4, 5, 20, 20, 16, 2]
12	Garner, Kevin	[26, 25, 20, 7, 12, 26, 23, 22, 28, 2, 6, 17]
13	Garrett, Antornique	[23, 25, 8, 2, 7, 4, 5, 27, 14, 5, 30, 11]
14	Gaston, Danielle	[20, 9, 19, 27, 18, 13, 13, 30, 10, 19, 17, 18]
15	Green, Brandon	[15, 20, 3, 13, 3, 6, 25, 11, 17, 20, 15, 6]
16	Hall, Dawyne	[21, 29, 28, 18, 30, 3, 13, 20, 28, 11, 27, 25]
17	Horace-Herron, Kelsey	[19, 11, 24, 12, 3, 9, 21, 12, 14, 23, 19, 15]
18	Horner, Ashly	[10, 30, 21, 5, 11, 23, 17, 8, 15, 13, 1, 23]
19	Houston, Alecia	[5, 24, 30, 6, 20, 14, 12, 18, 10, 25, 17, 28]
20	Jackson, Robert	[16, 23, 24, 2, 27, 15, 18, 8, 4, 16, 4, 3]
21	Kitchems, Lesley	[7, 27, 28, 17, 4, 18, 16, 24, 21, 14, 20, 21]
22	Knotts, Trave'Elle	[11, 18, 7, 24, 9, 12, 19, 21, 1, 14, 1, 6]
23	Lenard, Jorian	[26, 7, 9, 2, 8, 12, 20, 3, 4, 11, 14, 20]
24	McWilliams, Caleb	[1, 18, 15, 9, 11, 4, 12, 2, 16, 9, 23, 7]
25	Nathaniel, Jonathan	[23, 3, 5, 14, 13, 24, 27, 12, 23, 24, 29, 15]
26	Patterson, Damian	[17, 28, 20, 5, 5, 12, 15, 28, 7, 27, 23, 23]
27	Redmond, Brandon	[12, 28, 4, 16, 14, 16, 30, 8, 23, 26, 25, 20]
28	Rhodes, Mark	[30, 21, 14, 25, 13, 3, 21, 4, 5, 10, 8, 25]
29	Roberts, Cambria	[4, 25, 17, 13, 25, 7, 12, 16, 28, 4, 24, 20]
30	Scott, Maiah	[12, 3, 10, 5, 21, 14, 4, 26, 11, 14, 10, 9]
31	Silas, Christopher	[2, 15, 30, 30, 27, 27, 2, 8, 22, 22, 11, 27]
32	Tchounwou, Hervey	[20, 11, 20, 12, 10, 19, 20, 18, 11, 10, 20, 8]
33	Thompson, Da Johne'	[18, 22, 14, 12, 6, 15, 5, 25, 17, 24, 27, 26]
34	Tim-Owhefere, Ovuzoyen Prince	[30, 27, 15, 12, 8, 28, 26, 15, 27, 1, 24, 29]
35	Torres, Rauvardo	[10, 12, 18, 6, 8, 9, 22, 21, 27, 14, 5, 12]
36	Walk, Malerie	[14, 18, 26, 3, 8, 5, 23, 20, 23, 14, 22, 17]
37	Wells, Alicia	[22, 1, 2, 21, 21, 24, 6, 12, 18, 27, 20, 29]
38	Williams, Rapheal	[30, 6, 29, 8, 14, 13, 19, 24, 23, 21, 22, 3]
39	Wilson, Corey	[24, 3, 30, 2, 30, 19, 29, 20, 6, 3, 5, 9]
40	Tran, Phat	[23, 17, 4, 14, 4, 8, 5, 29, 2, 3, 6, 16]

(a - 11 pts) Construct a max heap of the array. Show the initial essentially complete binary tree and the transformation of the binary tree to a max heap via the reheapify operations at the indices of the internal nodes (as shown in the slides).

(b - 11 pts) Sort the max heap version of the array obtained from (a) to obtain a sorted array of integers. Show the structural changes of the max heap in each iteration.

(c - 7 pts) Transform the max heap of (a) to a binary search tree.

(d - 7 pts) For the binary search tree obtained in (c), determine the average number of comparisons for a successful search and the average number of comparisons for an unsuccessful search.

(e - 7 pts) Use the sorted array of (b) to construct a binary search tree.

(f - 7 pts) For the binary search tree obtained in (e), determine the average number of comparisons for a successful search and the average number of comparisons for an unsuccessful search.

(g - 6 pts) Construct a hash table of the given array using a hash function H(K) = K mod 5.

(h - 6 pts) For the hash table of (g), determine the average number of comparisons for a successful search and the worst case number of comparisons for an unsuccessful search.

(i - 9 pts) Consider the elements of the array assigned to you are known only one at a time. Construct a sequence of priority queues (as max heaps) with the insertion (enqueue) of one element at a time, as shown in the slides.

(j - 7 pts) Transform the binary search tree of (c) to a min heap.

(j - 7 pts) Transform the binary search tree of (c) to a min heap.

(k - 6 pts) Transform the max heap of (a) to a min heap via reheapify operations at the internal nodes.

(l - 3 pts) Are the min heaps of (j) and (k) the same or not? What can you say about the distribution of the data in the min heaps of (j) and (k)?

(m - 7 pts) Determine whether the binary search trees of (c) and (e) are height-balanced or not?

(n - 6 pts) Write the pre-order, in-order and post-order traversal of the vertices in the binary search tree of (c).