CS435 Algorithms Mid-block Notes

1. A logarithmic algorithm has a time complexity of O(log n).

2. A quadratic algorithm has a time complexity of ~~o(n~~~~3~~~~)~~ **O(n2)** [F]

3. Generally, an algorithm that runs in **O(n log n) time will take longer than** an algorithm that has **O(log n)** time complexity when n > n 0 .

4. When deciding between a List and an Array data structure, if the application will be frequently accessing the elements by rank and seldom inserting elements by rank then it is better to choose a **~~List~~** Array to store the elements. [F]

5. In the Queue ADT, the enqueue and dequeue operations run in ~~O(log n)~~ O(1) time.[F]

6. In a **proper** binary tree, **every internal node has two children**.

7. The height of a tree T is equal to the maximum depth of ~~an external~~ **any** node of T. (F)

8. Post-order traversal of a tree means the node is “visited” after the node's ~~parent~~ **descendants** is “visited”. (F)

9. \_\_\_\_\_\_A red-black tree that has 1000 internal nodes will have a height between 25 and 50.

10. \_\_T?\_\_\_\_Insertion-sort and selection-sort are best used only on sequences of a few hundred or fewer elements since faster sorting algorithms are available for larger sequences.

11. \_\_\_\_\_\_An unordered dictionary is inefficient for finding items since the entire dictionary might have to be scanned to find the key.

12. \_\_\_\_\_\_In Radix-sort, the key is divided into components and Bucket-sort is run on the input data using first the most-significant component, followed by Bucket-sorts using each

component in order.

Multiple choice. Circle the letter of the statement with the best answer.

11. An algorithm with O(n2) average case time complexity that takes 10 seconds to execute for an input size of 1000 elements will take how long to run when the input size is 10,000 elements.

b) from 50 up to 500 seconds (100)

12. What is the worst case time complexity of an insertion into a red-black tree of size n and why?

b) O(log n) – the insertion has to traverse all the levels of the tree.

13. What is primary benefit offered by hash tables?

**a)** They are very size efficient.

b) They expand automatically with no extra operations.

c) They handle various kinds of objects.

d) They are very fast for insertion and retrieval.

14. The Dictionary ADT includes what methods:

~~a) atRank(r), size(), isEmpty()~~

~~b) insertAfter(p), removeElement(), first()~~

c) findElement(k), insertItem(k,o), isEmpty()

~~d) find(l), keys(), isLast(p)~~

15. Which situation is Bucket-sort the best method to use for sorting?

a) When the input size of the data elements is less than a million.

b) When the keys are integers in a range less than the input size.

c) When the keys are very long and can be sub-divided evenly.

d) When the keys are short strings less than 32 characters.

16. The following hash table stores integer keys in the range of [0, 999] and uses the hash function and compression map, h(k) = k% 43. Collisions are handled using the quadratic probing strategy.

a) Into which slot will the integer key 58 be inserted? \_\_\_\_\_\_\_\_

b) Into which slot will the integer key 43 be inserted? \_\_\_\_\_\_\_\_

c) What is the load factor for the hash table as shown. \_\_\_\_\_\_\_\_\_

d) How is the delete operation handled in a hash table using quadratic probing strategy?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

17. What is the heap-order property for a min-heap?

a) The key stored at a node is greater than or equal to the key stored at the parent.

b) ~~The external nodes do not store keys or key-element pairs.~~

~~c) All the internal nodes on a level are “to the left” of the external nodes on the same level.~~

~~d) The last internal node of the tree stores the minimum key.~~

18. What is the primary advantage for implementing the Priority-Queue ADT using the heap data structure?

a) Heaps are more space-efficient than arrays or lists.

b) The heap provides random access to any key stored in the heap.

c) Inserting items in the heap always puts them on the end.

d) Removing the minimum item in the heap is faster than an array.

19. A total order relation for keys is necessary to satisfy the ....

a) object hierarchy.

b) comparison rule.

c) substitution property.

d) bin assignment.

20. The worst-case time complexity of Quick-sort is no better than Selection-sort or Insertion-sort.

Why is Quick-sort so widely used in applications?

a) The in-place Quick-sort is more memory efficient than the other sorts.

b) Recursive algorithms are very appealing to programmers.

c) Divide-and-conquer is a better approach for comparison-based sorting.

d) In the average case, Quick-sort is O(n log n).

21. Consider the following recurrence equation. Which statement about this equation is false?

a) The equation describes the running time of a recursive algorithm.

b) The equation can be reformulated to a closed-form expression.

c) The closed-form version is O(n log n).

d) The equation describes the running time of bottom-up heap construction.

22. In a Red-Black tree, the restructuring and recoloring operations...

a) ... keep the balance between red and black nodes so they are always the same.

b) ...cause updating to take twice as much time as searching.

c) ...are performed when searching, inserting and deleting key-element pairs.

d) ...are designed to maintain the depth so ordered dictionary searches are fast.