CISC 320 Midterm Exam – with solutions

Friday, Mar. 17, 2000

Name: ________________________________

The first 40 questions count 2 points each. The remaining three questions count 9, 3, and 8 points for a total of 100 points.
Multiple Choice: write the letter of the best choice in the blank.

1. _ a _
   (a) On a path from the root to a leaf in a red-black tree there are never two red edges in a row.
   (b) On a path from the root to a leaf in a red-black tree there are never two black edges in a row.
   (c) Both (a) and (b) are true.
   (d) None of the above.
2. ____ In a B-tree with t = 2, there are at most ___ keys stored in a node.
   (a) 1
   (b) 2
   (c) 3
   (d) 4
3. ____ In a B-tree with t = 2, an internal node has *at most* ___ child nodes.
   (a) 1
   (b) 2
   (c) 3
   (d) 4
4. ____ In a B-tree with t = 2, an internal node has *at least* ___ child nodes.
   (a) 1
   (b) 2
   (c) 3
   (d) 4
5. ____ Which of the following functions are $\Theta(n \lg(n))$?
   (a) Worst case time of insertionSort.
   (b) Average case time of insertionSort.
   (c) Best case time of insertionSort.
   (d) None of the above.

6. ____ Which of the following functions are $O(n \lg(n))$?

   (a) Worst case time of insertionSort.
   (b) Average case time of insertionSort.
   (c) Best case time of insertionSort.
   (d) None of the above.

7. ____ Which of the following functions are $\Theta(n \lg(n))$?

   (a) Worst case time of RandomizedQuickSort.
   (b) Expected case time of RandomizedQuickSort.
   (c) Both (a) and (b) are true.
   (d) None of the above.

8. ____ Which of the following functions are $\Theta(n \lg(n))$?

   (a) Worst case time of heapSort.
   (b) Average case time of heapSort.
   (c) Both (a) and (b) are true.
   (d) None of the above.

9. ____ Introspective sort is a combination of

   (a) insertionSort, mergeSort, and heapSort.
   (b) insertionSort, mergeSort, and quickSort.
   (c) mergeSort, heapSort, and quickSort.
   (d) insertionSort, heapSort, and quickSort.

10. ____ In introspectiveSort (STL's sort), which method is used to sort the short sequences?

   (a) insertionSort.
   (b) mergeSort.
   (c) heapSort.
   (d) quickSort.

11. ____ In introspectiveSort, what is gained by switching methods if the recursion depth gets too deep?

   (a) Avoid function call overhead.
   (b) Avoid sorting already sorted subsequences.
   (c) Guarantee $\Theta(n \lg(n))$ worst case time complexity.
   (d) None of the above.

12. ____ In mergeSort, there are two recursive calls to mergeSort. They occur

   (a) before the merge step.
   (b) after the merge step.
   (c) one before and one after the merge step.
   (d) during the merge step.

13. ____ Consider the merge step used in mergeSort. When two sorted sequences of length $n/2$ are being merged, the worst case time complexity of the merge operation is

(a) $\Theta(\lg(n))$.
(b) $\Theta(n)$.
(c) $\Theta(n \lg(n))$.
(d) None of the above.

14. ____ In quickSort, there are two recursive calls to quickSort. They occur

(a) before the partition step.
(b) after the partition step.
(c) one before and one after the partition step.
(d) during the partitioning.

15. ____ Consider the partition step used in quickSort. When a sequence of length n is partitioned, the worst case time complexity of the partitioning operation is

(a) $\Theta(\lg(n))$.
(b) $\Theta(n)$.
(c) $\Theta(n \lg(n))$.
(d) None of the above.

16. ____ QuickSort can be effectively randomized.

(a) The randomization is to determine which side of the partition is sorted first.
(b) The randomization is to make the array nearly sorted, whereupon insertion sort may be used to finish up.
(c) The randomization is to determine the partition element in a way independent of the input data.
(d) None of the above.

17. ____ The median of a sequence of n items may be found using

(a) $\Theta(\lg(n))$ comparisons .
(b) $\Theta(n)$ comparisons.
(c) $\Theta(n \lg(n))$ comparisons.
(d) None of the above.

18. ____

(a) The max and min of a sequence of n items may be found efficiently without any rearranging of the items in the sequence.
(b) The median of a sequence of n items may be found efficiently without any rearranging of the items in the sequence.
(c) Both (a) and (b) are true.
(d) None of the above.

19. ____ When the randomized select function is used to find the median of a sequence of n items,

(a) the items in the sequence are not rearranged at all.
(b) the items in the sequence are rearranged somewhat.
(c) the items are rearranged into fully sorted order.
(d) None of the above.

20. ____ Which of the following is a means to avoid "hash clash" collisions, when inserting elements into a hash table?

(a) separate chaining.
(b) linear probing.
(c) double hashing.
(d) All of the above.

21. ____ Which "hash clash" collision avoidance mechanism is appropriate if there are to be more items stored that there are spaces in the hash table.

(a) separate chaining.
(b) linear probing.
(c) double hashing.
(d) All of the above.

22. ____ A tree with N nodes has

(a) N-1 edges.
(b) N edges.
(c) N+1 edges.
(d) It varies. The number of edges is not determined by the number of nodes.

23. ____ Which sorting technique uses more than $O(\lg(n))$ memory space aside from the given array?

(a) InsertionSort.
(b) Mergesort.
(c) Heapsort.
(d) Quicksort.

24. ____ Which statement is *false*?

(a) InsertionSort requires about $k * n^2$ comparisons, for some constant k, when the input array is already sorted.
(b) Quicksort requires about $k * n^2$ comparisons, for some constant k, when the input array is already sorted, and the partition element choice is the first element.
(c) The first phase of heapsort is to put the keys in (a particular implementation of) a priority queue.
(d) There is a faster way to find the median of an array of n keys than by sorting the array then taking the middle key.

25. ____ Quicksort might be chosen over heapsort by some people because

(a) quicksort is *guaranteed* to require no more than $3n\lg(n)$ comparisons.
(b) heapsort requires additional memory space for the heap.
(c) quicksort is easier to program.
(d) quicksort is more suited to desktop PC's.

26. ____ Which of the following numbers is *less than the minimum* number of keys a B-tree of depth 2 with t = 100 could have.

(a) 500
(b) 50,000
(c) Both (a) and (b) are less than the minimum.
(d) None of the above.

27. ____ Which of the following numbers is *greater than the maximum* number of keys a B-tree of depth 2 with t = 100 could have.

(a) 5,000,000
(b) 500,000,000
(c) Both (a) and (b) are greater than the maximum.
(d) None of the above.

28. ____ A split operation during insertion in a red-black tree involves

(a) at least one rotation.
(b) at most two rotations.
(c) Both (a) and (b) are true.
(d) None of the above.

29. ____

(a) If an array is in reverse sorted order, then it is a heap.
(b) If an array is a heap, then it is in reverse sorted order.
(c) Both (a) and (b) are true.
(d) None of the above.

30. ____ When using double hashing, if the first hash function is key mod M, then a good choice for the second hash function would be

(a) key mod 8 plus 1, if M is divisible by 4.
(b) key mod $M - 1$ plus 1.
(c) key mod $M/2$ plus 1.
(d) key mod M plus 1.

31. ____ Suppose on a particular machine that heapsort requires 1 second to sort an array of size 65536 ($= 2^{16}$). How long do you expect heapsort to take on that machine to sort an array of size 1048576 ($= 2^{20}$)?

(a) 2 seconds.
(b) 20 seconds.
(c) 6 or 7 minutes.
(d) More time than the age of the universe.

32. ____ Suppose on a particular machine that the `select` function requires 0.5 seconds to find the median of an array of size 65536 ($= 2^{16}$). How long do you expect select to take on that machine to find the median of an array of size 1048576 ($= 2^{20}$)?

(a) 1 second.
(b) 8 seconds.
(c) 1 minute or so.
(d) More time than the age of the universe.

33. ____ During the running of randomizedQuickSort, how many calls are made to the random number generator in the worst case?

(a) about $\lg(n)$.
(b) about n.
(c) about $n \lg(n)$.
(d) about n^2.

34. ____ A heap is a priority queue represented as a tree which is ingeneously stored in an array. Which of the following are true.

(a) The tree root is in the the first position in the array.
(b) If a tree node is in position i of the array and is not a leaf, then one of it's children is in position $2i$.
(c) Both (a) and (b) are true.
(d) None of the above.

35. ____ A heap is a priority queue represented as a tree which is ingeneously stored in an array. Which of the following are true.

(a) The heap property is that the key at any node is greater than the keys at it's children.
(b) The extract-max function is implemented by taking the item at the current-size position, then decreasing the current-size by one.
(c) Both (a) and (b) are true.
(d) None of the above.

36. ____ Just now gigahertz cpu processors are being produced. Chip design experts are predicting that we can expect processor speeds to increase by another factor of 1 million in the next thirty years (continuing Moore's law of speed doubling every 1.5 years). Such machines will be petahertz processors, executing 10^{15} instructions per second! Suppose I take a current gigahertz processor and run it on a problem continuously for the next 30 years. Then, 30 years in the future, I buy a petahertz machine and run it on the same problem. How long will it take the petahertz processor to catch up with the gigahertz processor that has been running for 30 years?

(a) A few seconds.
(b) A few minutes.
(c) A few days.
(d) Never catch up.

37. ____ Professor Genioso has just published a new comparison based sorting algorithm together with a proof that it uses just $O(n)$ comparisons. Professor Saunders' reaction is

(a) Ho hum, we already can do that.
(b) Oh, wow. I can't wait to read that paper (and present it in 320).
(c) Yea, but it won't beat STL's sort in practice.
(d) Forget it. Without even looking at the paper, I know Genioso is wrong.

38. ____ Which of these dynamic set implementations provides good performance on the operations `successor` and `predecessor` as well as on the `insert` and `search` operations?

(a) hash table.
(b) Red-Black tree.
(c) Both (a) and (b) are true.
(d) None of the above.

39. ____ Which of these dynamic set implementations guarantees that each `insert` and `search` operation takes $O(\lg(n))$ time, when n items are in the set?

(a) B-tree with t = 500.
(b) Red-Black tree.
(c) Both (a) and (b) are true.
(d) None of the above.

40. ____ Which of these dynamic set implementations guarantees that each `insert` and `search` operation takes $O(\lg(n))$ time, when n items are in the set?

(a) Open addressed hash table of size $2n$.
(b) Separate chained hash table of size $n/2$.
(c) Both (a) and (b) are true.
(d) None of the above.

41. (9 points) In the following situations a hash table is to be used to implement a reverse telephone directory. The keys will be phone numbers and for a given number the associated value to be inserted/searched/deleted is the user of that number. For each scenario explain which hash table variant you would use and why.

(a) Every number in the world is to be stored (say about a billion numbers). It is specified for speed that the hash table itself must fit in main memory (200 megabytes max), however additional data structure (including the records themselves) may reside on disk.

(b) Every number in Delaware is to be stored (say about 100,000 7 digit numbers). The hash table should be sized to hold no more than 200,000 keys. Because there are relatively few telephone exchanges in Delaware, you should try to minimize "clustering".

(c) Every office number on campus is to be stored (all and only the numbers in exchange 831-xxxx). You can choose the hash table size.

42. (3 points) Draw the B-tree that results when a record with key G is inserted in this B-tree. Assume the variant of B-tree which stores only keys and links (pointers) in internal nodes and stores full records in external nodes. Assume there is room for 7 keys and 8 links (t = 4) in a page used as an internal node and room for 4 records in a page used as an external node.

```
                -----------
                | D I     |
                -----------

-----------  -----------  -----------
| A B C   |  | D E F H |  | I J K   |
-----------  -----------  -----------
```

43. `quickSort(p,e)` sorts the elements in index range [p .. e). It uses `Partition(p,e)` which permutes the elements and returns an index q such that the partition element, which was *p, is now *q and the elements in range [p..q) are no more than this partition element and the elements in range [q..e) are no less than *q.

    (a) (4 points) Write `quickSort`. You may use `Partition` and do not have to write code for it.

    (b) (4 points) `RandomizedQuickSort` differs from `QuickSort` only in that it uses `RandomizedPartition` in place of `Partition`. In turn `RandomizedPartition` differs from `Partition` only in that it picks a randomly placed element to be partition element rather than always picking the first element. Asymptotically, the average case run time of `QuickSort` is the same as expected case run time of `RandomizedQuickSort`. Explain the difference between expected time and average time in this situation. Make clear what is the advantage of having a good expected time over having a good average time. Include in your explanation a discussion of the behaviour of `QuickSort` and of `RandomizedQuickSort` when the input array is already in sorted order.