CSE 2320-004
Test 2
Fall 2018

Name Key 1

Multiple Choice:

1. Write the letter of your answer on the line (_____) to the LEFT of each problem.
2. CIRCLED ANSWERS DO NOT COUNT.
3. 3 points each

1. Based on dictionary search performance alone, the best justification for ordering a linked list is:

A A. Many more misses than hits are expected
B. Sentinels are more effective in speeding up search
C. Many more hits than misses are expected
D. Less storage will be needed

2. What is the worst-case time to perform MINIMUM(L) for a sorted, doubly-linked list with n nodes?

A A. $\Theta(1)$ B. $\Theta(\log n)$ C. $\Theta(n)$ D. $\Theta(n \log n)$

3. Suppose a postfix evaluator has already processed `3 2 1 + * 4 5 +` (with more to follow). What will be the contents of the stack (shown bottom-to-top going left-to-right)?

B A. `3 2 1 4 5` B. `9 9` C. `3 3 4 5` D. `18`

3 2 1
3 9
9

4. Suppose that only numbers in 1 . . . 100 appear as keys in a binary search tree. While searching for 50, which of the following sequences of keys could not be examined?

A A. 10, 40, 70, 30, 50 B. 10, 30, 70, 60, 50 C. 1, 100, 20, 70, 50 D. 100, 20, 80, 30, 50

5. Which phase of counting sort actually "counts"?

B A. first B. second C. third D. fourth

6. The most accurate description of the time to perform a deletion in an unbalanced binary search tree with n keys and height h is:

C A. $\Theta(1)$ B. $\Theta(\log n)$ C. $\Theta(h)$ D. $\Theta(n)$

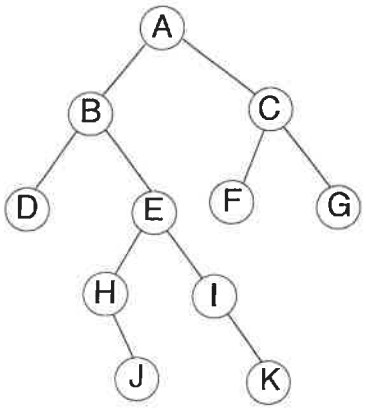
7. For which of the following sorts does the decision tree model not apply?

B A. Insertion B. LSD Radix Sort C. MERGE-SORT D. QUICKSORT

8. If POP is implemented as return stack[--SP], then PUSH of element X is implemented as:

C A. stack[++SP] = X B. return stack[SP++] C. stack[SP++] = X D. stack[--SP] = X

9. Suppose the tree below is a binary search tree whose keys and subtree sizes are not shown. Which node will contain the key with rank 8?



A

10. Which of the following will not be true regarding the decision tree for QUICKSORT for sorting n input values?

B

A. There will be $n!$ leaves.
B. Every path from the root to a leaf will have $O(n \log n)$ decisions.
C. There will be a path from the root to a leaf with $\Omega(n^2)$ decisions.
D. The height of the tree is $\Omega(n \log n)$.

11. The expected number of comparisons for finding the kth largest of n keys using PARTITION is in which asymptotic set?

B

A. $\Theta(\log n)$ B. $\Theta(n)$ C. $\Theta(n \log n)$ D. $\Theta(n^2)$

12. Recently, we considered an abstraction supporting the operations *allocate*, *allocateAny*, and *freeup* in constant time. How does the *allocateAny* operation detect that all items have already been allocated?

A

A. the header points at itself
B. the header points at `(-1)`
C. `next[n]!=0`
D. the recycling list is empty

13. Which of the following is a longest common subsequence for 0 1 2 0 1 2 and 0 0 1 2 1 2?

B

A. 0 0 1 1 B. 0 1 2 1 2 C. 0 0 1 2 D. 0 1 2 0

14. Suppose a (singly) linked list is used to implement a queue. Which of the following is true?

C

A. Like a circular queue, the maximum number of items is determined at initialization.
B. One node is always wasted.
C. The head points to the first element and the tail points to the last element.
D. The tail points to the first element and the head points to the last element.

15. Suppose a value k appears for p entries in the cost function table (C) for an instance of the longest monotonically increasing subsequence problem. Going left-to-right across the corresponding input sequence values (y_i), which statement is true?

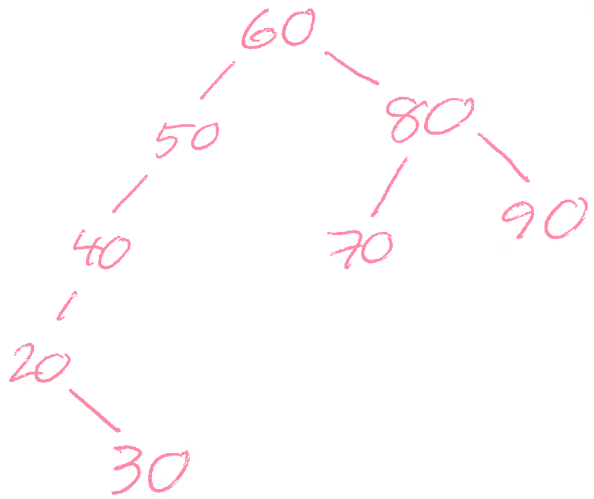
(Stated formally: For $i_1 < i_2 < \cdots < i_p$, suppose $C_{i_1} = C_{i_2} = \cdots = C_{i_p} = k$. Which statement is true regarding $y_{i_1}, y_{i_2}, \cdots, y_{i_p}$?)

D

A. They are monotonically decreasing
B. They are strictly increasing
C. They are monotonically increasing
D. They are strictly decreasing

Long Answer

1. Give the unbalanced binary search tree that results when the keys 60, 50, 40, 20, 80, 70, 90, 30 are inserted, ***in the given order***, into an initially empty tree. (5 points)

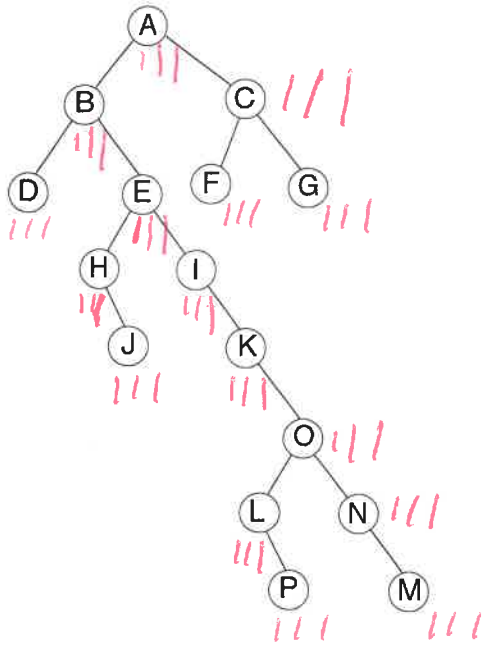


2. A billion integers in the range $0 \ldots 2^{30} - 1$ will be sorted by LSD radix sort. How much faster is this done using radix $0 \ldots 2^{10} - 1$ rather than $0 \ldots 2^{5} - 1$? Show your work. (10 points)

$0 \ldots 2^{10}-1$	$0 \ldots 2^{5}-1$
$K = 2^{10}$	$K = 2^5$
$d = 3$	$d = 6$
$\Theta(d(n+k))$	
$\Theta(3(1B + 1024))$	$\Theta(6(1B + 32))$

↑ Twice as fast

3. Give the inorder, postorder, and preorder traversals of the given binary tree. Be sure to label your traversals appropriately. (10 points)



Preorder
A B D E H J I K O L P N M C F G

Inorder
D B H J E I K L P O N M A F C G

Postorder
D J H P L M N O K I E B F G C A

```
AB 1    9    0    7    6    8    2    5    3    4
   1 AB 9    0    7    6    8    2    5    3    4
   1 A  9  B 0    7    6    8    2    5    3    4
   1    0 A  9  B 7    6    8    2    5    3    4
   1    0 A  9    7  B 6    8    2    5    3    4
   1    0 A  9    7    6  B 8    2    5    3    4
   1    0 A  9    7    6    8  B 2    5    3    4
   1    0    2 A  7    6    8    9  B 5    3    4
   1    0    2 A  7    6    8    9    5  B 3    4
   1    0    2    3 A  6    8    9    5    7  B 4
   1    0    2    3  < 4 >  8    9    5    7    6
```

4. Show the result after PARTITION (Version 1) manipulates the following subarray. Recall that both pointers start at the left end of the subarray. (10 points)

1	9	0	7	6	8	2	5	3	4

```
  i   S
-------
  0   0
  1   2
  2   3
  3   5
  4   7
  5  11
  i   C
-------
  0   0
  1   6
  2   1
  3   2
  4   6
  5   2
  6   6
  7   3
  8   3
  9   4
 10   3
 11   5
 12   4
 13   5
 14   4
Solution
  i   S
-------
  4   7
  3   5
  1   2
```

5. Use the dynamic programming solution for subset sums to determine a subset that sums to 14. Be sure to give the complete table that would be produced. (10 points, no points for solving by inspection)

i	0	1	2	3	4	5
S_i	0	2	3	5	7	11

6. Use dynamic programming to solve the following instance of the strictly longest increasing subsequence. Be sure to provide the table for the binary searches, along with the tables of lengths and predecessors for backtracing. (10 points, no points for solving by inspection)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
5	10	15	20	25	7	10	15	22	25	5	10	22	26	27
C 1	2	3	4	5	2	3	4	5	6	X	X	X	7	8
j 0	1	2	3	4	1	6	7	8	9	X	X	X	10	14

1: 5/1

2: 10/2 ~~7~~/6

3: 15/3 10/7

4: 20/4 15/8

5: 25/5 22/9

6: 25/10

7: 26/14

8: 27/15