CSE 2320-002 Name ______________________________________
Test 1
Spring 2012 Last 4 Digits of Student ID # ___________________

Multiple Choice. Write your answer to the LEFT of each problem. 3 points each

1. The time to run the code below is in:

```
for (i=n; i>=0; i-=2)
  for (j=0; j<n*n; j++)
    sum+=i+j;
```

A. $\Theta(n)$ B. $\Theta(n \log n)$ C. $\Theta(n^2)$ D. $\Theta(n^3)$

2. The time to fill in the dynamic programming matrix when computing the LCS for sequences of lengths m and n is:

A. $\Theta(n)$ B. $\Theta(m + n)$ C. $\Theta(n \log n)$ D. $\Theta(mn)$

3. What is indicated when `find(i)==find(j)` while maintaining disjoint subsets?

A. `i` and `j` are in the same subset
B. `i` and `j` are leaders for different subsets
C. `i` and `j` are leaders for the same subset
D. `i` is the ancestor of `j` in one of the trees

4. Suppose $H_n = \frac{137}{60}$. What is the value of n?

A. 4 B. 5 C. 6 D. 7

5. The function $2\log n^2 + \log n$ is in which set?

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

6. Which technique allows interfacing a prority queue with a dictionary?

A. Array of handles
B. Binary search
C. Binary search tree
D. `PQchange`

7. What is n, the number of elements, for the largest table that can be processed by binary search using no more than 7 probes?

A. 31 B. 63 C. 64 D. 127

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

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

9. The goal of the Huffman coding method is:

   A. Another type of binary search tree.
   B. Maximize the compression for every string.
   C. Minimize the expected bits per symbol.
   D. Store a string within the leaves of a binary tree.

10. When solving the fractional knapsack problem, the items are processed in the following order.

   A. Ascending order of weight
   B. Ascending order of $$$/lb
   C. Descending order of weight
   D. Descending order of $$$/lb

11. The purpose of the binary searches used when solving the longest (monotone) increasing subsequence (LIS) problem is:

   A. to assure that the final solution is free of duplicate values
   B. to determine the longest possible increasing subsequence terminated by a particular input value
   C. to search a table that will contain only the LIS elements at termination
   D. to sort the original input

12. Suppose you are using the substitution method to establish a Θ bound on a recurrence $T(n)$ and you already know that $T(n) \in \Omega(\log n)$ and $T(n) \in O(n^3)$. Which of the following cannot be shown as an improvement?

   A. $T(n) \in O(1)$ B. $T(n) \in O(\log n)$ C. $T(n) \in \Omega(n^2)$ D. $T(n) \in \Omega(n^3)$

13. The time to run the code below is in:

```
sum=0;
for (i=0; i<n; i=i+sum)
  sum++;
```

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

14. The expected time for insertion sort for n keys is in which set? (All $n!$ input permutations are equally likely.)

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

15. The number of calls to `mergeAB` while performing `mergesort` on n items is:

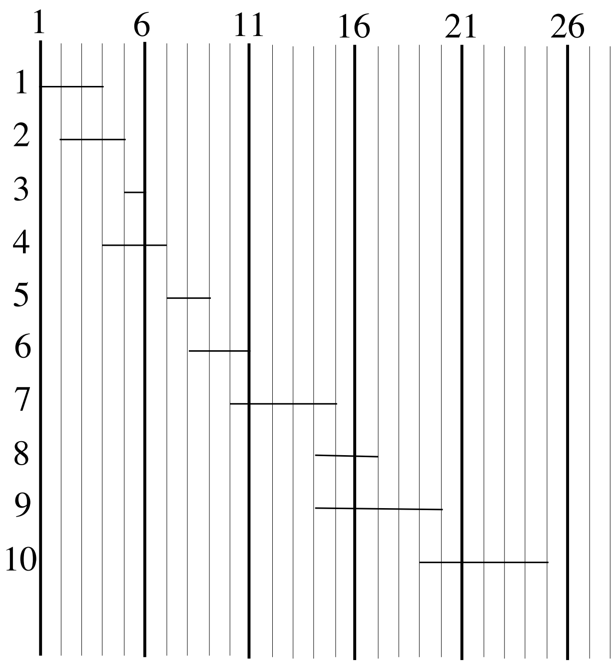
A. $\Theta(\log n)$ B. $\Theta(m+n)$ C. $\Theta(n)$ D. $\Theta(n \log n)$

Short Answer:

1. Use the greedy method of Notes 6 for unweighted interval scheduling of the set of intervals in the first long-answer problem. You should simply list the numbers of the chosen intervals. (5 points)

Long Answer

1. Use dynamic programming to solve the following instance of weighted interval scheduling. Be sure to indicate the intervals in your solution and the sum achieved. 10 points



v_i	p_i	$m(i)$
1	0	
5	0	
3	2	
8	1	
1	4	
2	4	
3	5	
1	6	
4	6	
1	8	

2. Use the recursion-tree method to show that $T(n) = 2T\left(\frac{n}{4}\right) + \sqrt{n}$ is in $\Theta\left(\sqrt{n} \log n\right)$. 10 points

3. Use the substitution method to show that $T(n) = 2T\left(\frac{n}{4}\right) + \sqrt{n}$ is in $\Theta\left(\sqrt{n} \log n\right)$. 10 points

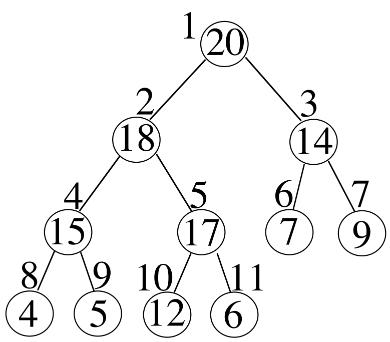
4. Complete the following instance of the optimal matrix multiplication ordering problem, including the tree showing the optimal ordering. 10 points

```
p[0]=8
p[1]=6
p[2]=4
p[3]=3
p[4]=5
p[5]=7
```

```
          1           2           3           4           5
 1     0     0   192     1   216     1   336     3   ???     ?
```

2	-------	0 0	72 2	162 3	303 3
3	-------	-------	0 0	60 3	189 3
4	-------	-------	-------	0 0	105 4
5	-------	-------	-------	-------	0 0

5. Show the result after performing `PQdelmax` twice on the following maxheap. 10 points



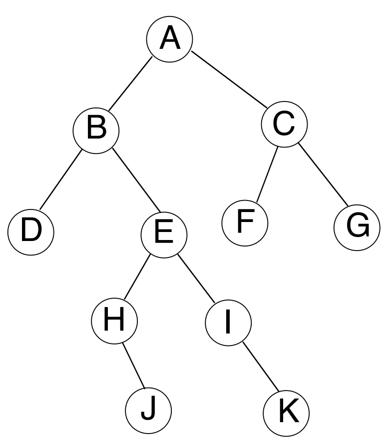
CSE 2320-002 Name ______________________________________
Test 2
Spring 2012 Last 4 Digits of Student ID # ___________________

Multiple Choice. Write your answer to the LEFT of each problem. 3 points each

1. In a binary search tree, which element does not have a predecessor?

   A. any one of the leaves B. the maximum C. the minimum D. the root

2. Suppose the tree below is a binary search tree whose keys are not shown. Which node will contain the key that is the predecessor of the key stored at H?



A. A
B. B
C. C
D. G

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

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

4. What is the worst-case time to perform MAXIMUM(L) for an unordered, doubly-linked list with n nodes?

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

5. Given a pointer to a node, the worst-case time to delete the node from a doubly-linked list with n nodes in ascending order is:

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

6. What is the worst-case time to find the predecessor of a key in an unbalanced binary search tree storing n keys? Assume that parent pointers are available.

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

7. An array with 150 unique elements is subscripted starting with 0. You would like to iteratively use PARTITION to find the thirty largest values, but there is no requirement that the thirty largest values be ordered. Which of the following is not correct?

   A. If 131 is returned from PARTITION, we must continue.
   B. If 120 is returned from PARTITION, we are done.
   C. If 119 is returned from PARTITION, we are done.
   D. If 118 is returned from PARTITION, we must continue.

8. In which situation will a sentinel be inappropriate?

   A. Binary search for a key in an ordered table, to simplify and speed-up code
   B. Search for a key in an unordered table, to simplify and speed-up code
   C. Search for a key in an unordered linked list, to simplify and speed-up code
   D. Red-black tree, to simplify code

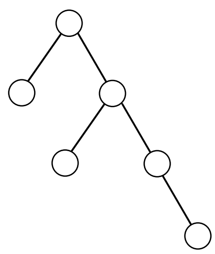
9. In a red-black tree holding n keys, what is the total number of left and right pointers that will be set to `nil` (the sentinel)?

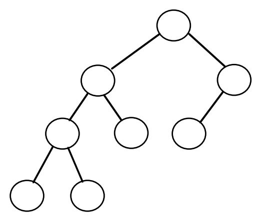
   A. $n - 1$
   B. n
   C. $n + 1$
   D. None of the above

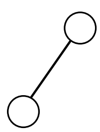
10. How should the successor of a node without a right child in an unbalanced binary search tree be found?

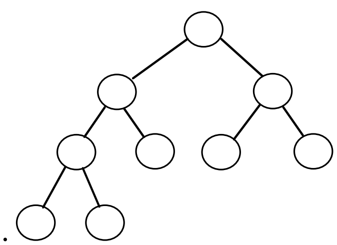
    A. Examine the ancestors of the node
    B. Go left, then proceed to the right
    C. Go right, then proceed to the left
    D. Preorder traversal

11. Which of the following binary trees has *multiple* legal colorings as a red-black tree?

A. 

B. 

C. 

D. 

12. Recursion is often an alternative to using which data structure?

    A. Linked list   B. Queue   C. Stack   D. 2-d array

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

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

14. Which binary tree traversal corresponds to the following recursive code?

```
void traverse(node x)
{
if (x==null)
  return;
traverse(x->left);
// process x here
traverse(x->right);
}
```

    A. inorder   B. postorder   C. preorder   D. search for key x

15. Suppose that only numbers in 1 . . . 1000 appear as keys in a binary search tree. While searching for 500, which of the following sequences of keys could not be examined?

A. 700, 200, 600, 550, 500
B. 200, 700, 600, 300, 400, 500
C. 100, 1000, 200, 800, 300, 900, 500
D. 300, 400, 900, 800, 500

Long Answer

1. Give the unbalanced binary search tree that results when the keys 50, 100, 80, 70, 60, 90, 120 are inserted, in the given order, into an initially empty tree. (5 points)

2. Describe the three situations that can occur for deletion from an unbalanced binary search tree. An example for each situation will be fine. (10 points)

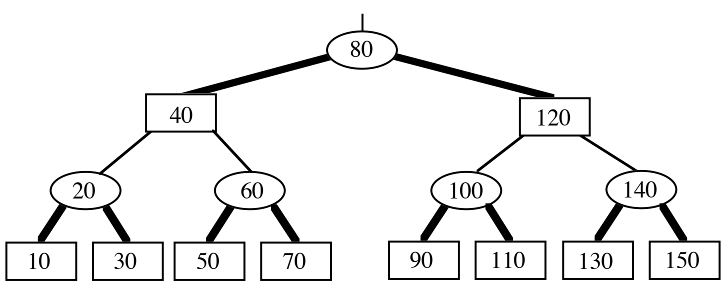
3. A billion integers in the range 0 . . . 999,999 are to be sorted by LSD radix sort. How much faster will this be done if radix 0 . . . 999 is used rather than decimal (0 . . . 9) radix? Show your work. (10 points)

4. Show the result after PARTITION manipulates the following subarray. Be sure to circle which version of PARTITION you applied. (10 points)

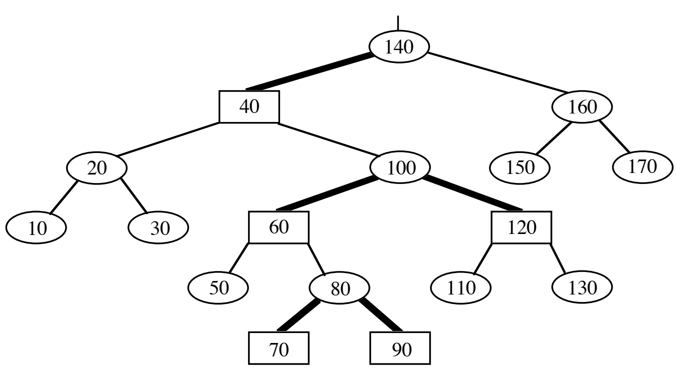
8	2	5	3	4	1	9	0	7	6

Version: 1 2/Sedgewick

5. Insert 55 into the given red-black tree. Be sure to indicate the cases that you used. (10 points)



6. Insert 95 into the given red-black tree. Be sure to indicate the cases that you used. (10 points)



Name ________________________________

Spring 2012 Last 4 Digits of Student ID # ______________________

Multiple Choice. Write the letter of your answer to the LEFT of each problem. 2 points each

1. Suppose a depth-first search on a directed graph yields a path of tree edges from vertex X to vertex Y and a path of tree edges from vertex X to Z. If there is also an edge from Y to Z, then its type will be:

   A. Tree B. Back C. Cross D. Forward

2. During a breadth-first search, the status of a black vertex is:

   A. It has been completely processed.
   B. It is in the FIFO queue.
   C. It is in the priority queue.
   D. It is undiscovered.

3. Suppose a directed graph has a path from vertex X to vertex Y, but no path from vertex Y to vertex X. The relationship between the finish times is:

   A. finish(X) > finish(Y) B. finish(X) < finish(Y)
   C. finish(X) = finish(Y) D. could be either A. or B.

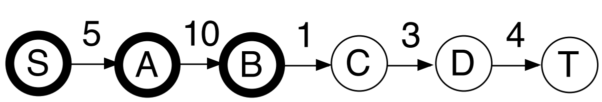
4. The capacity of any cut is:

   A. An upper bound on the maximum flow. B. A lower bound on the maximum flow.
   C. The same as the capacity of all other cuts. D. The same as the maximum attainable flow.

5. Suppose a maximum bipartite matching with *k* edges is found using Edmonds-Karp. Which of the following does not hold?

   A. The capacity of the minimum cut is *k*. B. There will be *k* + 1 breadth-first searches.
   C. All residual network capacities are zero or one. D. Every augmenting path uses three edges.

6. The capacity of the following cut is ______. (S vertices are bold.)



   A. 1 B. 10 C. 15 D. 23

7. Dijkstra's algorithm, when implemented with a heap, is most suitable for:

   A. Finding the minimum spanning tree of a dense graph.
   B. Finding the minimum spanning tree of a sparse graph.
   C. Finding the shortest paths from a designated source vertex in a dense graph.
   D. Finding the shortest paths from a designated source vertex in a sparse graph.

8. Before searching for a minimum cut in a network, it is useful to do the following:

A. Find one augmenting path.
B. Determine the type of each edge using depth-first search.
C. Find and record augmenting paths until none remains.
D. Perform a breadth-first search on the input network.

9. When finding the strongly connected components, the number of components is indicated by:

A. The number of cross edges found during the second depth-first search.
B. The number of back edges found during the first depth-first search.
C. The number of restarts for the second depth-first search.
D. The number of restarts for the first depth-first search.

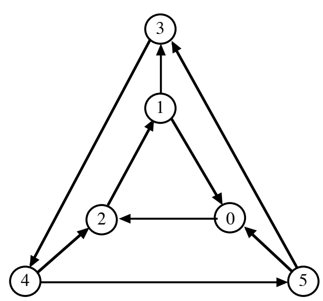
10. The fastest method for finding the diameter of a tree (where distance is measured in "hops") is to:

A. Use breadth-first search.
B. Use Dijkstra's algorithm.
C. Use the Floyd-Warshall algorithm.
D. Use the Ford-Fulkerson algorithm.

11. During depth-first search on an undirected graph, a cycle is indicated by which edge type?

A. Back B. Cross C. Forward D. Tree

12. What is the number of strongly connected components in this graph?



A. 1 B. 2 C. 3 D. 4

13. Which edge is chosen in a phase of Kruskal's algorithm?

A. The unprocessed edge (x, y) of smallest weight such that `find(x)==find(y)`
B. An edge of maximum-weight in a cycle (to be excluded)
C. An edge that is on a shortest path from the source
D. The unprocessed edge (x, y) of smallest weight such that `find(x)!=find(y)`

14. Suppose a depth-first search is performed on an undirected graph with *n* vertices. The graph is a free (i.e. unrooted) tree if:

A. all edges are tree edges
B. both C and D
C. there are no forward edges
D. there are no cross edges

15. Suppose a double hash table has $\alpha = 0.8$ (without deletions), the upper bound on the expected number of probes for unsuccessful search is:

A. 1.2 B. 2 C. 5 D. 10

16. Suppose that there is only one path from vertex 5 to vertex 10 in a directed graph: $5 \rightarrow 7 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 10$. During the scan of which column will Warshall's algorithm record the presence of this path?

A. 2 B. 3 C. 4 D. 7 E. 8 F. 10

Problems 17 and 18 refer to the following hash table whose keys are stored by double hashing using $h_1(\text{key})$ = key % 13 and $h_2(\text{key})$ = 1 + (key % 12).

0	1	2	3	4	5	6	7	8	9	10	11	12
			120	186	187	162		122	110			194

17. 266 would be inserted into which slot of the given table?

A. 0 B. 1 C. 2 D. 7 E. 10 F. 11

18. 313 would be inserted into which slot of the given table? (266 has not been inserted)

A. 0 B. 1 C. 2 D. 7 E. 10 F. 11

Problems 19 and 20 refer to the following hash table whose keys are stored by linear probing using h(key) = key % 13.

0	1	2	3	4	5	6	7	8	9	10	11	12
			94		122	110	20	86	87	62		

19. 148 would be inserted into which slot of the given table?

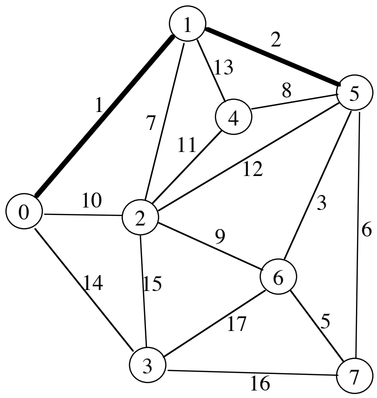
A. 0 B. 1 C. 2 D. 4 E. 11 F. 12

20. 133 would be inserted into which slot of the given table? (148 has not been inserted)

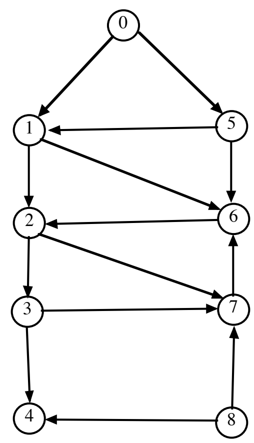
A. 0 B. 1 C. 2 D. 4 E. 11 F. 12

Long Answer

1. What are the entries in the heap (for Prim's algorithm) before <u>and</u> after moving the next vertex and edge into the minimum spanning tree? DO NOT COMPLETE THE ENTIRE MST!!! Edges already in the MST are the thick ones. Edges currently not in the MST are the narrow ones. You do <u>not</u> need to show the binary tree for the heap ordering. 10 points.

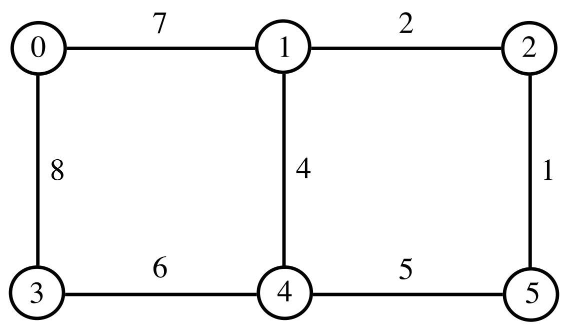


2. Perform depth-first search on the following graph, including start/finish times and edge types (T=tree, B=back, C=cross, F=forward.) Assume that the adjacency lists are ***ordered***. Write your answer in the tables below. 10 points

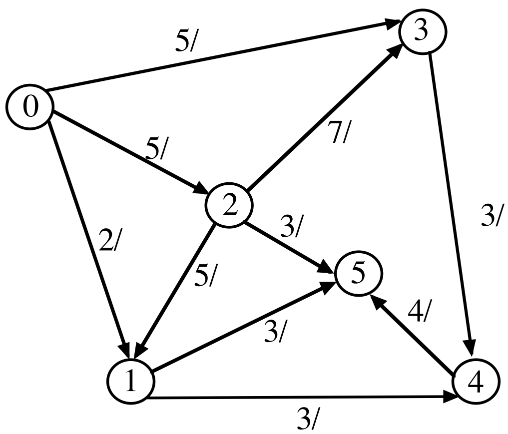


Vertex	Start	Finish	Edge	Type	Edge	Type
0	__1__	____	0 1	____	5 6	____
1	____	____	0 5	____	6 2	____
2	____	____	1 2	____	7 6	____
3	____	____	1 6	____	8 4	____
4	____	____	2 3	____	8 7	____
5	____	____	2 7	____		
6	____	____	3 4	____		
7	____	____	3 7	____		
8	____	____	5 1	____		

3. Show the ***compressed*** adjacency list representation this graph. (Answers using conventional adjacency lists will receive no credit.) 10 points.



4. Give augmenting paths for determining a maximum flow and give a minimum cut for the following network. 0 is the source and 5 is the sink. 10 points.

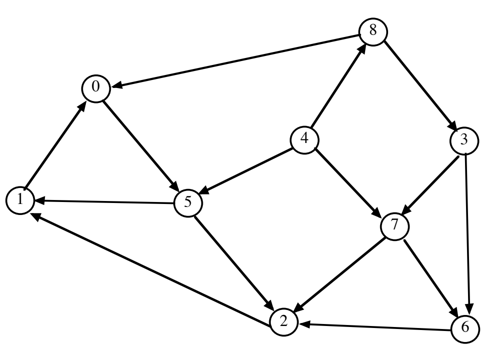


S vertices:

T vertices:

Augmenting Paths and Contribution to Flow:

5. Demonstrate, for the graph below, the algorithm that uses ***two*** depth-first searches to determine the strongly-connected components. 10 points



6. Give an algorithm (C code or pseudocode), based on the Floyd-Warshall algorithm with ***successors***, to determine for "all pairs" of starting and destination vertices the path that ***minimizes*** the sum of the intermediate vertex numbers. So, if there is an input edge from vertex i to vertex j, then the sum

would be zero. If there were a path 10 → 20 → 15 → 12 →3, its sum would be 20 + 15 + 12 = 47. As usual, vertex numbers start at 0, so the use of vertex 0 as an intermediate vertex is "free".
(10 points)