CSCI 3323
Fall 2017
Exam 3

Last Name: ____________________

First Name: ____________________

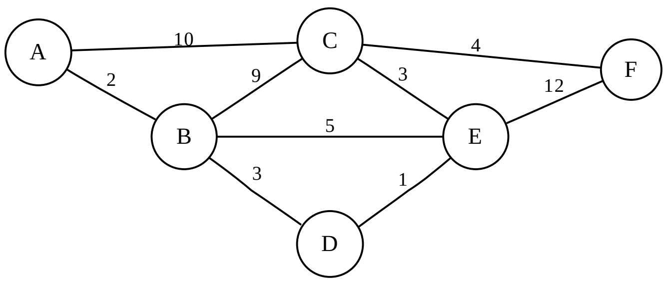
Unless instructed otherwise, write answers on the answer sheets provided.

1. (2 pts each) Name the complexity of each of these algorithms/operations.
   (a) deleteMax on heap
   (b) quicksort (worst case)
   (c) counting sort
   (d) mergesort
   (e) Dijkstra's single-source shorted path algorithm
   (f) DFS
2. (6 pts) Trace mergesort for this array assuming that the partition element is always selected to be the first element in the partition: $A =< 33, 17, 66, 24, 28, 56, 19, 62 >$
3. (6 pts) Trace counting sort for this array: $A =< 2, 1, 7, 2, 6, 3, 6 >$.
4. (4 pts) Explain why the implementors of Java's built-in sorts chose to use a mergesort when sorting arrays of objects and a quicksort when sorting arrays of simple types.
5. (4 pts) Briefly outline the proof that the best possible worst-case complexity for a comparison sort is $O(n \lg n)$.
6. (4 pts) Name two stable sorts.
7. (2 pts each) Briefly define each of the following terms:
   (a) heap
   (b) directed graph
   (c) complete graph

8. (8 pts) Write a method called `isHeap` that accepts (as parameters) an array of integer values along with the number of elements being used in the array. The method should return `true` if the array possesses all of the properties of a heap, and should return `false` otherwise. You may assume the existence of the following methods:

```
public static int left(int i) { return (2*i+1); }
public static int right(int i) { return (2*(i+1)); }
public static int up(int i) { return ((i-1)/2); }
```

9. Answer the following questions based on the graph below. When tracing an algorithm, always select the node that comes first in the alphabet if the order doesn't matter to the algorithm.



   (a) (4 pts) Show how the graph could be represented as an adjacency list. NOTE: There is no need to show edge weights … only adjacency.

   (b) (6 pts) Trace Prim's MST algorithm on the graph. Redraw the graph after every step and indicate the edges selected for inclusion in the MST by drawing them bold faced. Start with vertex `A`.

   (c) (6 pts) List order in which vertices in the graph would be visited if the graph were traversed using a BFS. Also show the resulting BFS tree. Start with vertex `A`.

   (d) (4 pts) Does a Hamiltonian circuit exist for this graph? If so, indicate the circuit.

10. (10 pts) Write a method named `hasEulerPath` that accepts (as parameters) an adjacency matrix along with the number of vertices in the graph represented by the matrix. The method should return `true` if the graph has an Euler path, and should return `false` otherwise.

11. (Up to 5 BONUS pts) Give a scheme that could be applied to any non-stable sorting method so as to make the resulting sort stable. State the amount of memory and time the "fix" adds to the original sort.