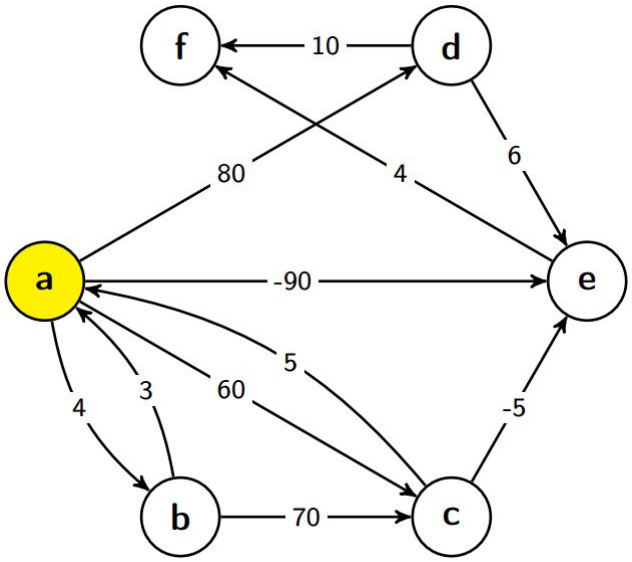
CSE 373 Section Handout #10: Final Review

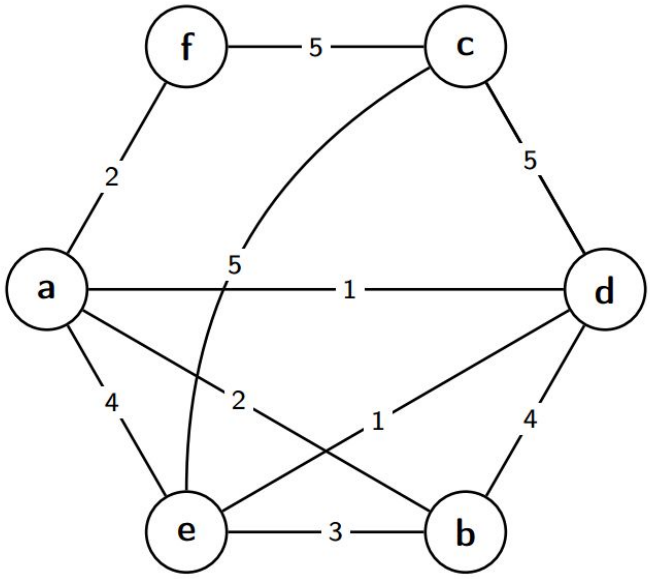
1. Dijkstra's:



a. Use Dijkstra's Algorithm to find the lengths of the shortest paths from a to each of the other vertices. For full credit, show your work.

b. Are any of the lengths you computed using Dijkstra's Algorithm in part (a) incorrect? For each length that is incorrect, explain what the correct answer is and why the answer from part (a) was incorrect.

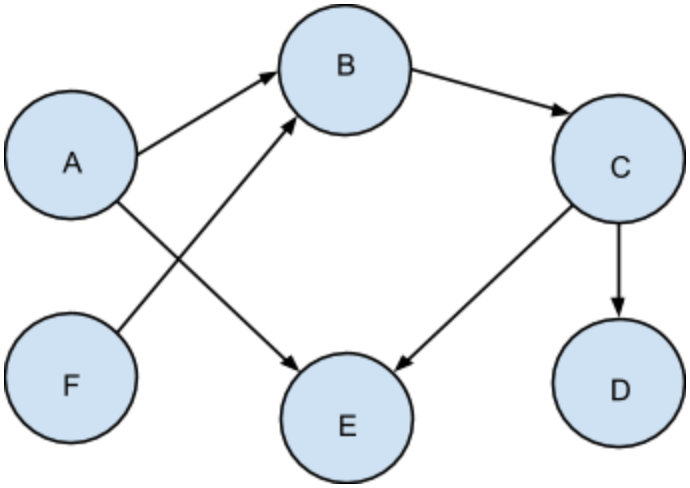
2. Kruskal's:



a. Use Kruskal's Algorithm to find two minimum spanning trees of the above graph.

b. Imagine that the above graph had some negative edges in it. Would Prim's Algorithm necessarily return a correct result? Explain your answer in 1-2 sentences.

3. Topological Sort:



   a. Use Topological Sort to provide a Topological Ordering for this graph.

   b. Are there multiple valid Topological Orderings for this graph? Why or why not?

4. Suppose we use radix sort to sort the numbers below, using a radix of 10. Show the state of the sort after each of the first two passes, not after the sorting is complete. If multiple numbers are in a bucket, put the numbers that "come first" closer to the top.
   Numbers to sort (in their initial order):

   17, 45, 877, 31, 7, 222, 42, 43, 301, 2525, 9, 27, 64

   Result after the first pass:

0	1	2	3	4	5	6	7	8	9

Result after the second pass:

0	1	2	3	4	5	6	7	8	9

5. Sorting: (Assume that array sun[] has indices: 0 to size-1)

```
SunnySort(int[] sun) {
    for (int i = 1; i < size; i++) {
        int j;
        int temp = sun[i];
        for (j = i; j > 0 && temp < sun[j-1]; j--) {
            sun[j] = sun[j-1];
        }
        sun[j] = temp;
    }
}
```

a. This is actually a sort mentioned in class. What sort is this?

b. Describe the best and worst case data set for this sort. (If all cases behave similarly, please state that.) What is the big-O running time of those two data sets?

Best Case data set?

Best Case running time?

Worst Case data set?

Worst Case running time?

c. Is it an in-place sort? Why or why not? (no credit without a reason or a definition of in-place, for partial credit define in-place sorting)

d. Is it a stable sort? Why or why not? (no credit without a reason or definition of stable, for partial credit define stable sorting)

Interview-esque Questions:

6. You are teaching a course, and need to keep track of students' grades. You also want to be able to print <StudentName, grade>, in alphabetical order. During the first few weeks, students may register and drop the course. What would be a good data structure to store this information in? Why? Are there any downsides to your choice?

7. You want to create a new ADT that has all of the properties of a List, but with the added benefit of disallowing duplicates. How would you implement this? Which data structure(s) would you use, and how? Explain the tradeoffs of your decision.

8. The Maximum Spanning Tree Problem is: given a connected, undirected graph with weights on its edges, find a spanning tree that maximizes the weight of its edges.

   How can you use a Minimum Spanning Tree algorithm to find the maximum Spanning Tree of a graph? Write pseudocode for your algorithm

9. You have been tasked with storing mathematical expressions (eg. 2 + 4 * a - 3 / b). Come up with a good way to represent these expressions in memory. Don't worry about parsing the expression, just describe how you would store it.