

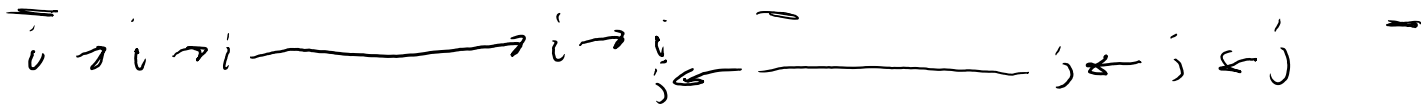
Review Questions

Thursday, May 3, 2018 8:55 AM

CS 211 Review Questions

1. [3] Show the result of quicksort after one iteration of the quicksort algorithm (until $I \geq J$ and pivot is swapped back).

0	1	2	3	4	5	6	7	8	9	10	11	12	13
3	9	12	1	2	15	11	8	10	17	4	6	5	13
3	5	6	1	2	4	8	13	10	17	15	12	9	11



2. [3] Perform radix sort on the following numbers:

101, 98, 27, 15, 333, 234, 531, 503, 122, 432, 199, 200, 155, 188, 79, 631

	10^0	10^1	10^2
0	200	200, 101, 503	15, 27, 79, 98
1	101, 531, 631	15	101, 122, 155, 188, 199
2	122, 432	122, 27	200, 234
3	333, 503	531, 631, 432, 333, 234	333
4	234,		432
5	15, 155	155	503, 513
6			631
7	27	79	
8	98, 188	188	
9	199, 79	98, 199	

3. [3] Order the following sorting algorithms from worst to best case algorithmic complexity:

Merge Sort, Bubble Sort, Radix Sort, Quick Sort, Shell Sort, Insertion Sort, Heap Sort, Selection Sort, Tree Sort, Shaker Sort

Bubble Sort (N^2), Insertion Sort (N^2), Shaker Sort (N^2), Selection Sort (N^2)

Heap Sort ($N \log N$), Tree Sort ($N \log N$), QuickSort ($N \log N$), Merge Sort ($N \log N$), Shell Sort ($N \log N$)

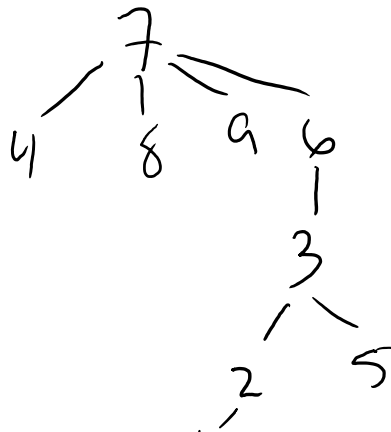
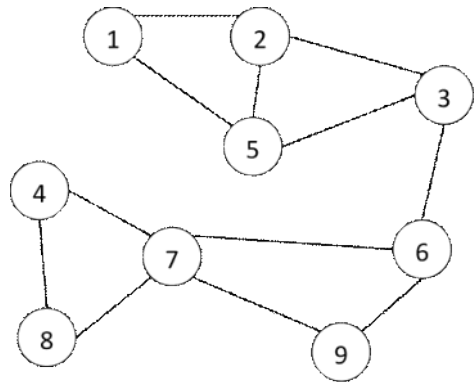
Radix Sort ($N \cdot M$)

4. [3] Draw the BFS search tree for the following graph starting at vertex 7.

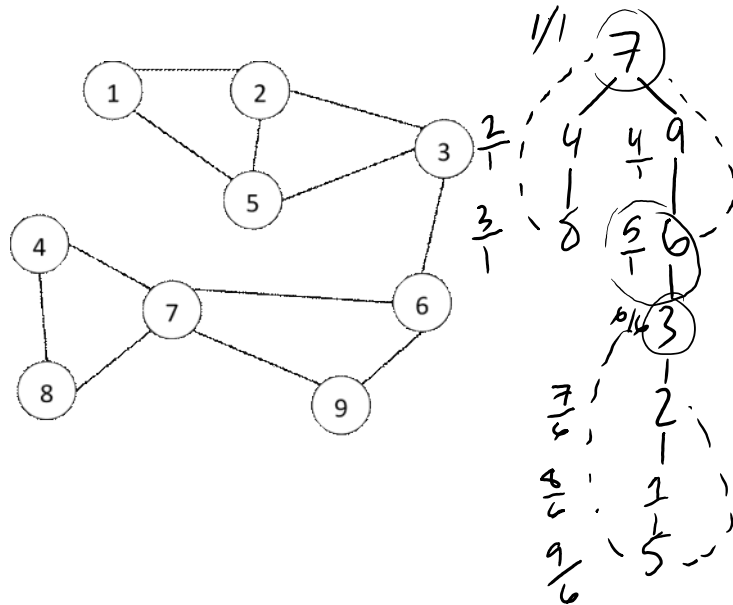


7.

4. [3] Draw the BFS search tree for the following graph starting at vertex 1.



5. [3] Articulation Points. Draw the DFS articulation tree for the following graph starting at vertex 7. Circle any articulation points in your tree.



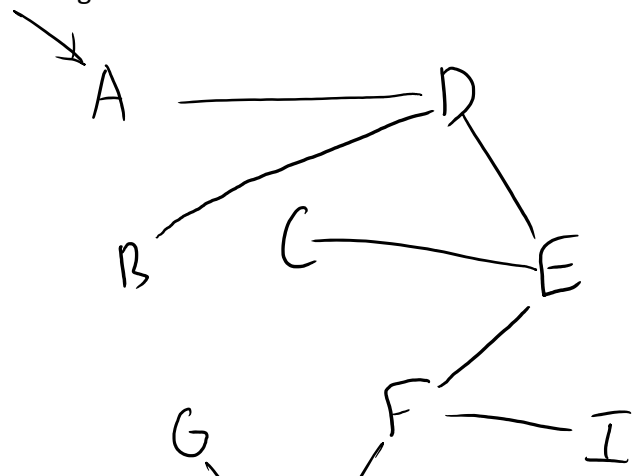
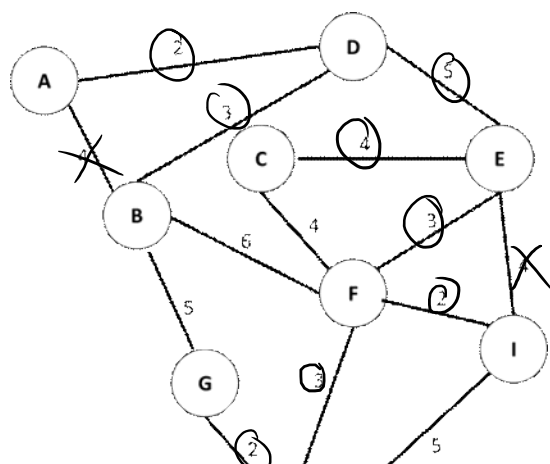
6. [1] What data structure allow us to perform a DFS on a graph?

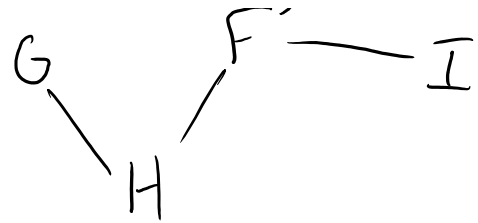
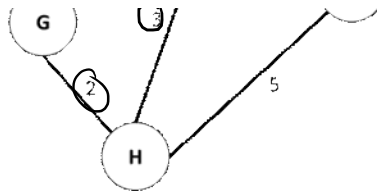
Stack

7. [1] What data structure allows us to perform a BFS on a graph?

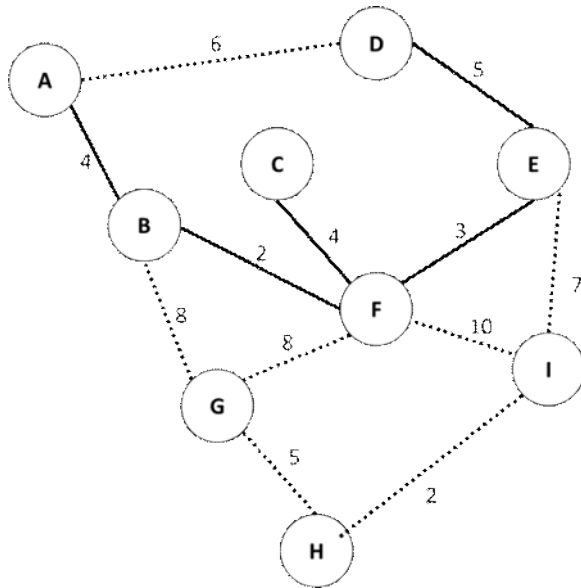
Queue

Construct an MST for the following graph starting at A





Consider the following partial MST (solid edges = accepted edges). What was the last edge added to the MST? What edge will be added next?



Also on the exam (probably)

- Comprehensive exam
- Algorithm analysis questions
 - What runtime complexity of code - fill in loop count (similar to E1 & E2)
 - How does the following data structure affect runtime complexity
- Tree questions
 - "Build this tree given 2 traversals" question
 - AVL rotation question
- Heap Questions
 - Work through an add / remove / merge on binary, binomial, skew heaps
- Hash table questions
 - Where will the next insert place value X in a linear, quadratic, cuckoo, hopscotch heap
- Graph questions
 - Build MST
 - Interpret or build DFS articulation tree
- Sorting questions
 - Given the following partially sorted graph, which sorting algorithm is currently running.
 - Conceptual questions related to algorithm analysis
 - E.g. What causes insertion sort to run in its best case time complexity?
 - E.g. What causes quicksort to run its worst case time complexity?