# CS 211 Review Questions

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

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 3 | 9 | 12 | 1 | 2 | 15 | 11 | 8 | 10 | 17 | 4 | 6 | 5 | 13 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |

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

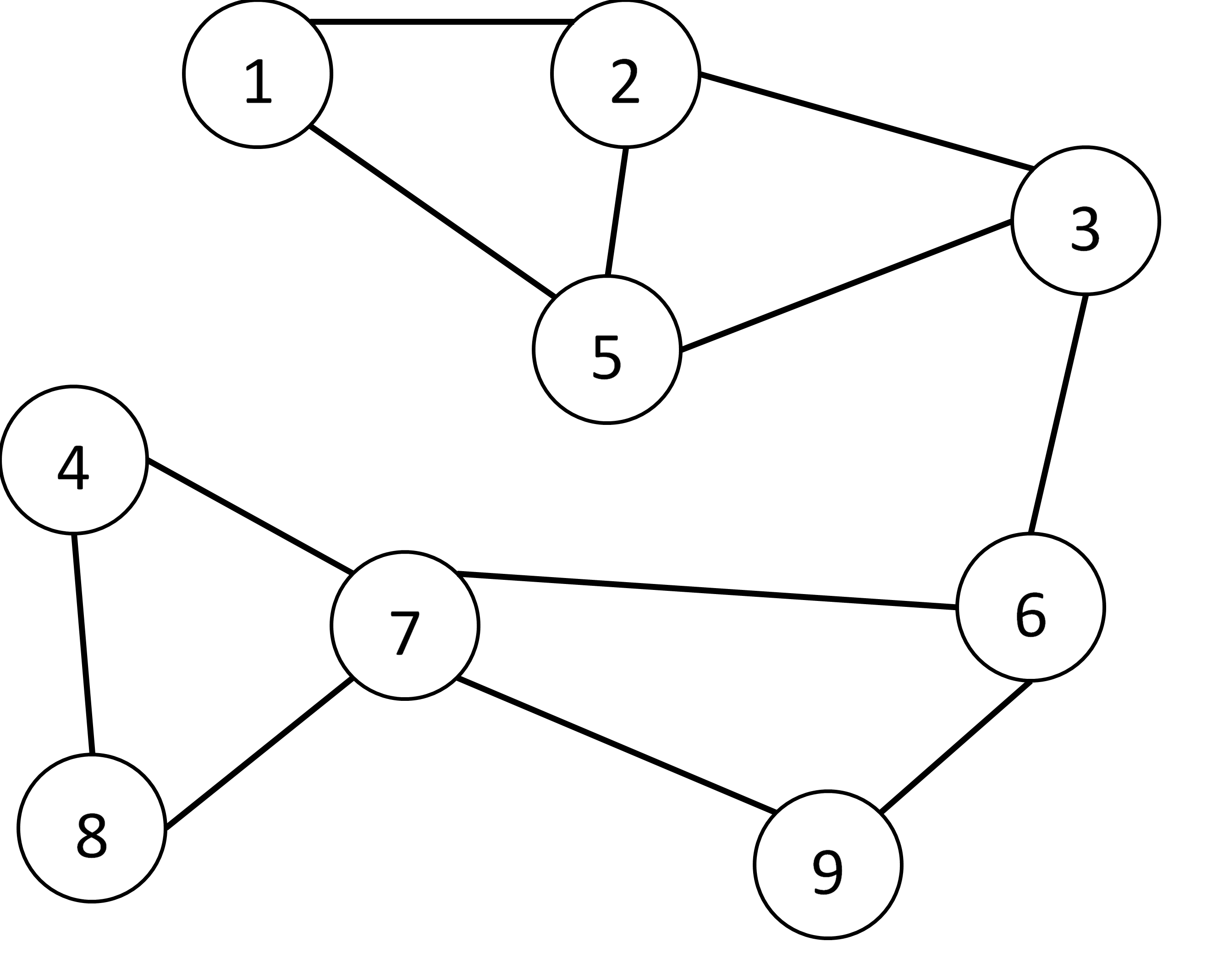
101, 98, 27, 15, 333, 234, 531, 503, 122, 432, 199, 200, 155, 188, 79, 631

|  |  |  |  |
| --- | --- | --- | --- |
| 0 |  |  |  |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |

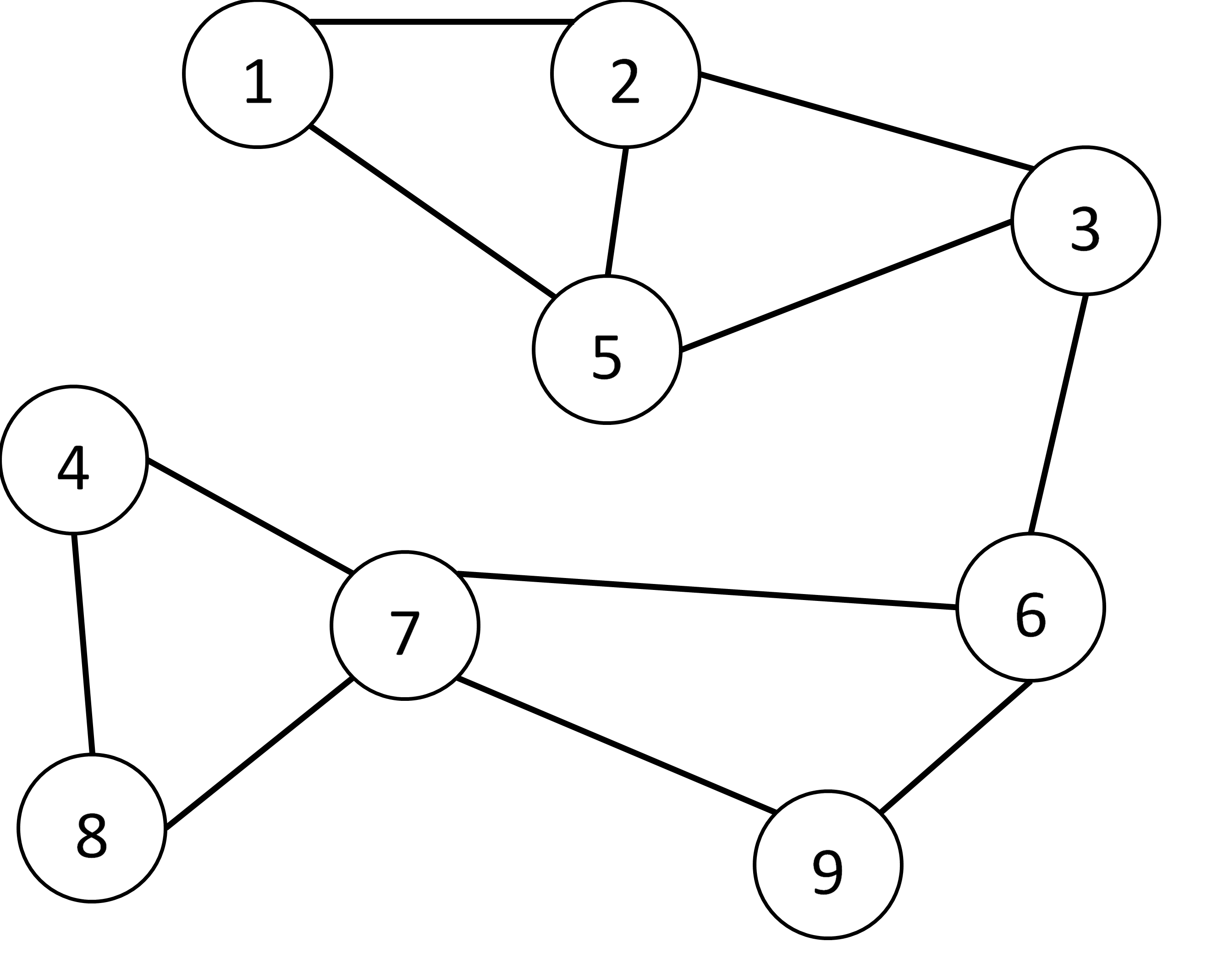
3. [3] Order the following sorting algorithms from worst to best case algorithmic complexity:

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

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



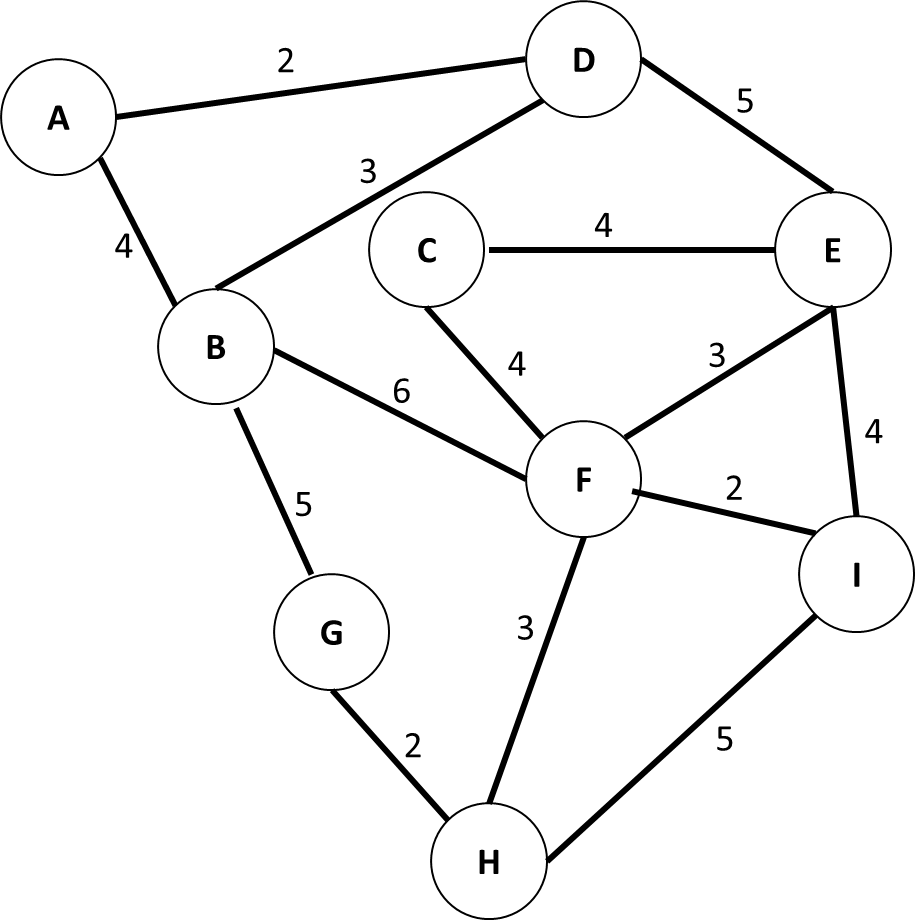
5. [3] Articulation Points. Draw the DFS articulation tree for the following graph starting at vertex 7. Circle any articulation points in your tree.



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

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

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?

