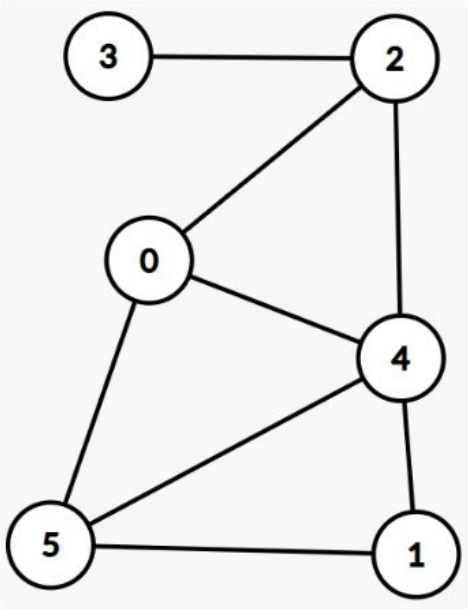
1. Time complexity in terms of V(# of vertices) and E(# of edges):
    a. Undirected Graph:
        i. Matrix Representation: Since in a matrix model there are a total of V^2 values, the time complexity of this representation can be best described as O(n^2) where n = V
        ii. List Representation: The adjacency list representation requires list of size m = E for each node or in this case vertex. THe resulatance time complexity can be best summarized as O(n+m) where n = V and m = E.
    b. Directed Graph:
        i. Matrix Representation: Similar to the undirected graph matrix model, the directed model has a time complexity of O(n^2)
        ii. List Representation: O(n+m)
2. Given:



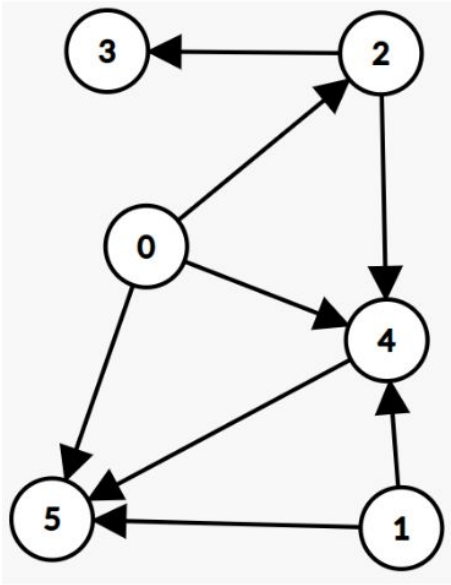
    a. Depth First Search Algorithm:
        i.

Step	1	2	3	4	5	6	7
Stack	0	0,2	0,2,3	0,2,4	0,2,4,1	0,2,4,1,5	-
Output	0	0,2	0,2,3	0,2,3,4	0,2,3,4,1	0,2,3,4,1,5	**0,2,3,4,1,5**

    b. Breadth First Search:

i. Breadth First Search utilizes a queueing system to determine what to output next where once a node is completed being searched, the search refers to a queue.

Step	1	2	3	4	5	6	7	8	9
Queue	-	2	2,4	2,4,5	4,5,3	5,3,1	3,1	1	-
Output	0	0,2	0,2,4	0,2,4,5	0,2,4,5,3	0,2,4,5,3,1	0,2,4,5,3,1	0,2,4,5,3,1	**0,2,4,5,3,1**



3. Given:
   a. Depth First Search Algorithm:
      i.

Step	1	2	3	4	5	6	7
Stack	0	0,2	0,2,3	0,2,4	0,2,4,5	1	-
Output	0	0,2	0,2,3	0,2,3,4	0,2,3,4,5	0,2,3,4,5,1	**0,2,3,4,5,1**

   b. Breadth First Search Algorithm:
      i.

Step	1	2	3	4	5	6	7	8	9
Queue	-	2	2,4	2,4,5	4,5,3	5,3	3	1	-
Output	0	0,2	0,2,4	0,2,4	0,2,4	0,2,4	0,2,4	0,2,4	**0,2,4**

				,5	,5,3	,5,3	,5,3	,5,3,1	**,5,3,1**