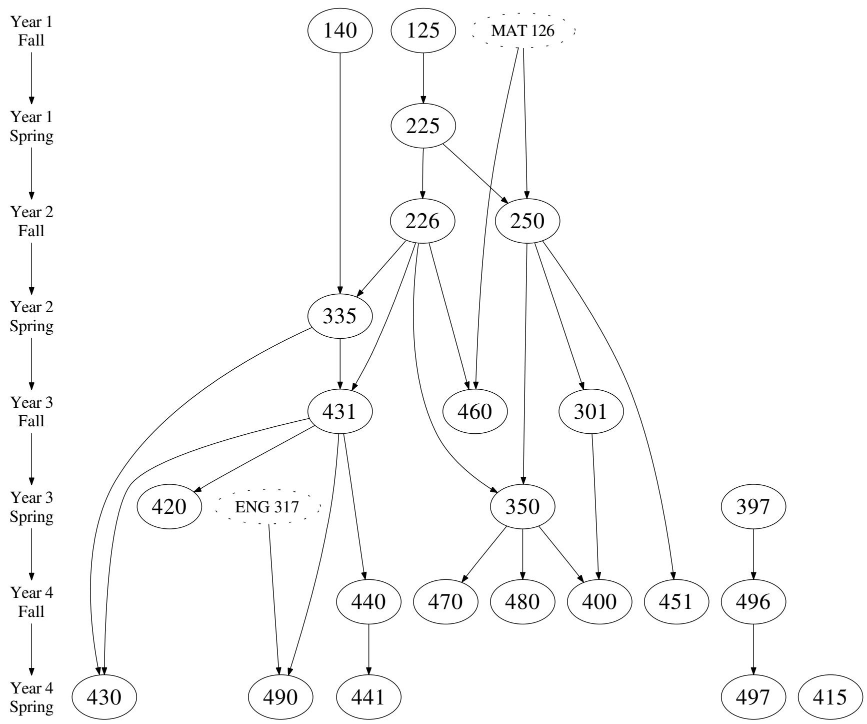
1. List the members of your group below. Underline your name.

2. The following graph encodes course prerequisites in computer science. (Ignore the timeline on the left.)



(a) The graph's *order* (number of vertices) is ______ .

(b) The graph's *size* (number of edges) is ______ .

(c) The number of *strongly connected components* is ______ .

(d) The number of *connected components* (undirected edges) is ______ .

(e) The number of *directed simple cycles* is ______ .

(f) The number of *undirected simple cycles* is ______ .

(g) The length of the *longest path* is ______ .

(h) The *in-degree* and *out-degree* of the '431' vertex are ______ and ______ .

(i) The number of *distinct simple paths* from '125' to '430' is ______ .

(j) The number of *edge-disjoint paths* from '125' to '430' is ______ .

(k) The number of *edge-disjoint paths* from '226' to '430' is ______ .

(l) The vertices *adjacent to* '226' (its *out-neighbors*) are ____________________ .

(m) The vertices *adjacent from* '226' (its *in-neighbors*) are ____________________ .

3. For the graph of Question 2, what is the number of *subgraphs* with vertex set $V_1 = \{225, 226, 250, 335, 350, 431\}$? Explain your answer.

4. For the graph of Question 2, (a) depict the *subgraph induced by* the vertex set V_1 of Question 3 and (b) augment the above induced subgraph by adding an edge from 431 to 225.

5. Depict the (a) adjacency matrix and (b) adjacency lists representations of the graph of Question 4 (augmented).

6. Let A be the adjacency matrix of Question5. Let A^0 denote the identity matrix I and define $A^k = A^{k-1} \times A$ (matrix multiplication) for $k > 0$. Compute A^k for $k = 0, 1, 2, 3, 4, 5$.

7. For the matrix A^k defined in Question 6, provide a method for computing A^k_{ij} (the entry in the i'th row and j'th column of A^k) directly, without using any explicit matrix operations. Explain your answer.

8. Provide a graph theoretic interpretation of $\sum_{i=0}^{k} A^i$ (matrix addition) where A^k is from Question 6. Explain your answer.

9. Provide a simple algorithm that computes the length of a shortest path between every pair of vertices of a directed graph of order n.

10. Modify the algorithm of Question 9 to compute the *transitive closure* of a graph and the *reflexive transitive closure* of a graph. Depict the results for the graph of Question 2.

[additional space for answering the earlier question]