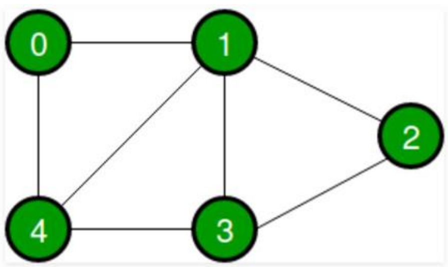
Sam Lee
6/2/18
CSC 130
Professor Chidella

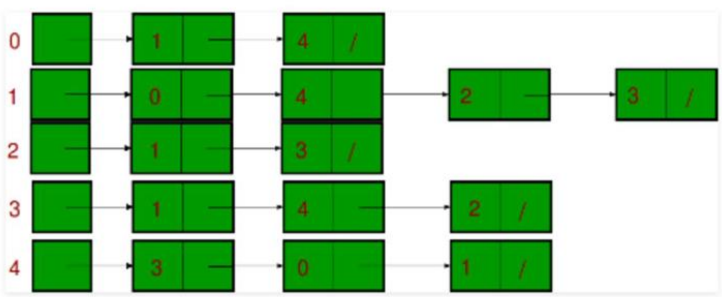
HW #3

1a. Since there are millions of nodes in the graph, this resources that the adjacency matrix will have millions X millions size of the matrix where because some of the nodes have especially few edges so here this will be an expenditure of memory and lead to inefficiency.

1b. In case of adjacency list illustration, the issue is with the nodes which have hundreds of thousands of edges related with it, this means that these nodes will have hundreds of thousands of nodes in the linked list depiction of the edges and the searching time of an edge will increase here which is an inadequacy.

1c. If the depiction is done in such a way that that only important information is displayed to show, and consumption can be avoided, so if we use incidence matrix sign then the inefficiencies can be avoided to some extent.





1d. Suppose a grid is there which is like a tree and one vertex is related to all the vertices and all the others are accessory vertices.

2a. The algorithm will be selecting the courses in such a way that the courses are selected with minimum time required for the term along with that the property of pre-requisites also needs to be satisfied.

2b. Every time a level in the graph is completed and a node or (class) is enqueued, each of the classes dequeued can be store in separate arrays. These arrays will effectively be all the classes required in a specific semester.

2c. The running time of the algorithm will be O(V+E) time.

3a. We find adjacent edges from A and update the distance and the previous vertices whose distance is less than the current value.

Names	s	A	B	C	D	E	F	G	H	I	T
Vertex	X	X	X	X	X	X	X	X		X	X
Previous	S	S	A	B	S	A	E	S	G	E	C
Cost	0	1	3	5	4	3	6	6	12	6	9

3b. The algorithm will not necessarily choose the path with the fewest edges, but the one with the least total coast. You could modify the algorithm to have a counter with each vertex visited. At the end, if there are multiple paths with the same cost, the algorithm with choose the path with the lower counter variable.

3c. An example where Dijkstra's algorithm gives the wrong answer is in a negative cycle. Each time we traverse the loop the weights decreases and therefore it can run infinite times and each time weight is different.

3d. It is not possible to add weights on each edge because shortest path might get changed.If there (a) - > (b) ->(c) and ab = 6 and bc =-3 by adding 3 we would get ab = 9 and bc = 0 which will mean the shortest path is ac but that is not true.