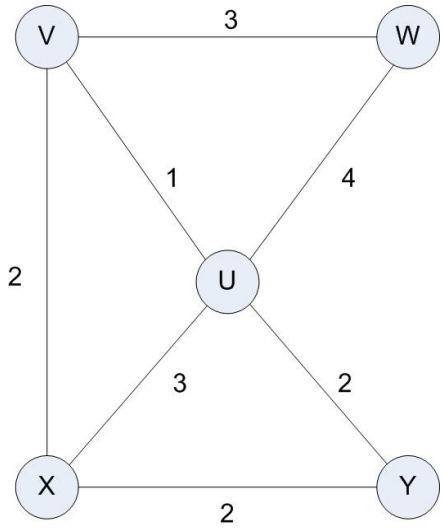
Lab 11 A

1. Must every dense graph be connected? Prove your answer.

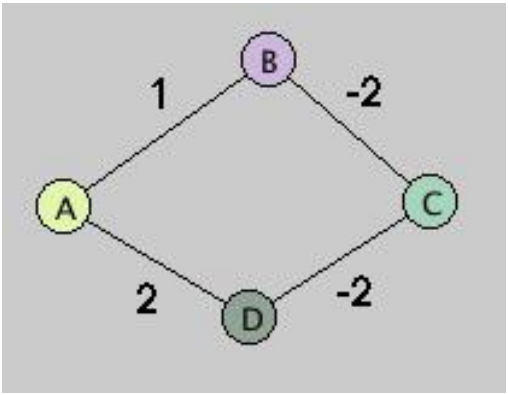
2. Carry out the steps of the fast Dijkstra's algorithm to compute the length of the shortest path between vertex V and vertex Y in the graph below. Your final answer should consist of two elements:

    a) The length of the shortest path from V to Y
    b) The list A[] which shows shortest distances between V and every other vertex

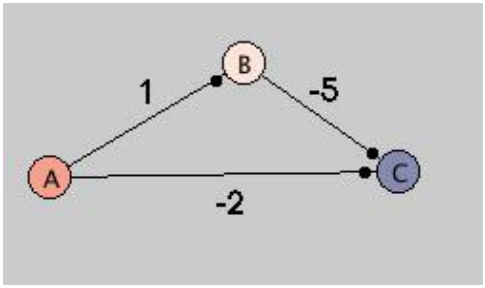


3. *Points about Dijkstra's Algorithm*

    a. What is the shortest path from A to C in the graph below (using any algorithm you like)?



b. As discussed in class, Dijkstra's algorithms also works when we start with a directed graph, but it still requires edges to have non-negative weights.
   i. What goes wrong when you apply Dijkstra's algorithm to the following directed graph, where starting vertex is A? (Arrowheads look like big dots in this diagram.)



   ii. Now apply the Dynamic Programming algorithm for DAGs, discussed in class, to this graph. Show the computation of each D[v], as in the slides. Does it produce a correct result?

4. Describe an algorithm for deleting a key from a heap-based priority queue that runs in O(log n) time, where n is the number of nodes. (Hint: You may use auxiliary storage as the priority queue is built and maintained.) This technique is needed for the optimized Dijkstra algorithm discussed in the slides.