

**NAME:** Please put your name here!

1. A frequent error in written text is the transposition of adjacent letters, as in "witner" instead of "winter". Using two replaces, or one delete and an insert, assigns a cost of 2 to this error (if we make the cost of insertion, deletion, and replacing characters all equal to 1). To account for its frequency, we want to assign a cost of 1 to this error. We do this by allowing an additional edit operation:

Transpose two adjacent letters.

Rewrite the recurrence relation for computing edit distance to account for this new operation.

Hint: Look two cells back.

<b>Solution:</b>
------------------

2. Create a specific instance of a coin change problem for which the greedy algorithm does not produce an optimal solution. In other words, specify the values of your coins and the sum of money for which you want to produce change, and then show that the greedy algorithm does not provide the optimal change for that sum of money using those coins.

<b>Solution:</b>
------------------

3. This is a version of Exercise 16.1-3 from page 422 of our textbook.

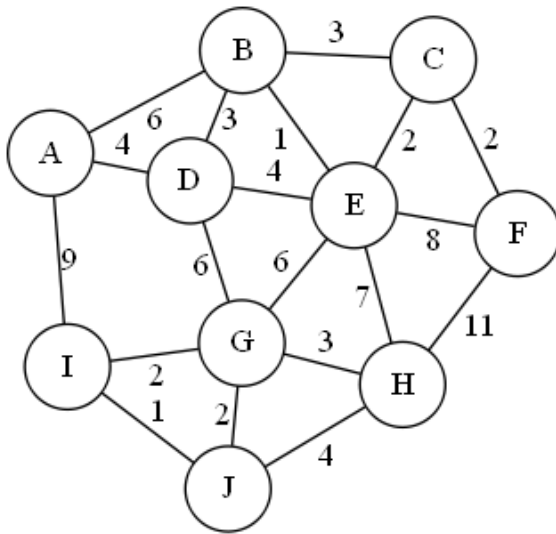
- (a) For the activity-selection problem, show that the greedy strategy of always choosing the compatible remaining activity with the earliest start time need not lead to an optimal solution. (That is, come up with a specific instance of the activity-selection problem for which this greedy strategy fails to find an optimal solution.)

**Solution:**

- (b) For the activity-selection problem, show that the greedy strategy of always choosing the compatible remaining activity with the least duration need not lead to an optimal solution. (That is, come up with a specific instance of the activity-selection problem for which this greedy strategy fails to find an optimal solution.)

**Solution:**

4. Consider the following graph  $G$ .



- (a) Report the order of the vertices encountered on a breadth first search of  $G$  starting from vertex  $A$ . Iterate through adjacent vertices using alphabetical order. Draw the BFS spanning tree.

**Solution:**

- (b) Report the order of the vertices encountered on a depth first search of  $G$  starting from vertex  $A$ . Iterate through adjacent vertices using alphabetical order. Draw the DFS spanning tree.

**Solution:**

- (c) Find and draw the minimum spanning tree of  $G$ . Compare the weight of the minimum spanning tree with the weights of the BFS and DFS spanning trees from Parts (a) and (b).

**Solution:**