ECE 606, Fall 2021, Assignment 6
Due: Tuesday, October 26, 11:59pm

Submission: submit three things: (i) your solutions for the written problems to crowdmark, (ii) the page with your name and student ID for the **[python3]** problem to crowdmark, and, (iii) your **[python3]** file to the appropriate Dropbox on Learn.

1. Prove Claim 44 in Lecture 6 of your textbook.

   That is, the denominations of coins we allow are $D = \langle 1, 5, 10, 25 \rangle$. We assume we have infinitely many coins of each denomination. We are now given an amount $a \in \mathbb{Z}^+$. The claim is that the problem of handing out the fewest total coins whose values add up to a possesses the following greedy choice: hand out as many coins of the highest denomination we can.

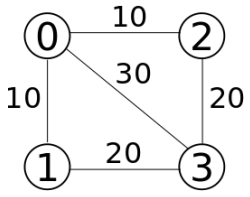
   (Hint: as a sanity check, you can double-check that your proof falls apart if $D = \langle 1, 3, 4 \rangle$.)

2. For the interval scheduling problem in Lecture 6 of your textbook, prove that the following is a valid greedy choice: pick a request with the latest start time.

3. Consider the proof for Theorem 24.6 (Correctness of Dijkstra's algorithm) in Lecture 6 of your textbook.

   In the graph that is input to Dijkstra's, assume that we allow negative edge-weights, but no negative weight cycles. Where does the proof fall apart?

4. **[python3]** You are given as input (i) a connected undirected graph $G = \langle V, E, w \rangle$, with positive integer edge-weights, i.e., the codomain of the function w is $\mathbb{Z}^+$, and, (ii) an edge $\langle u, v \rangle$ in that graph. Design and implement a polynomial-time Python 3 subroutine hasmst that returns True if G has a minimum spanning tree that contains the edge $\langle u, v \rangle$ and False otherwise.

   G is encoded as an adjacency list with weights incorporated. For example, the above graph would be encoded as:

   $$[[[2, 10], [1, 10], [3, 30]], [[0, 10], [3, 20]], [[3, 20], [0, 10]], [[1, 20], [2, 20], [0, 30]]]$$

   The input edge (ii) is encoded as a list of size two, e.g., $[0, 3]$. For the above graph, for the edge $[0, 3]$, the correct return value is False, and for the edge $[3, 2]$, the correct return value is True.