MET CS 566 - Analysis of Algorithms
Assignment 4- 20 Points

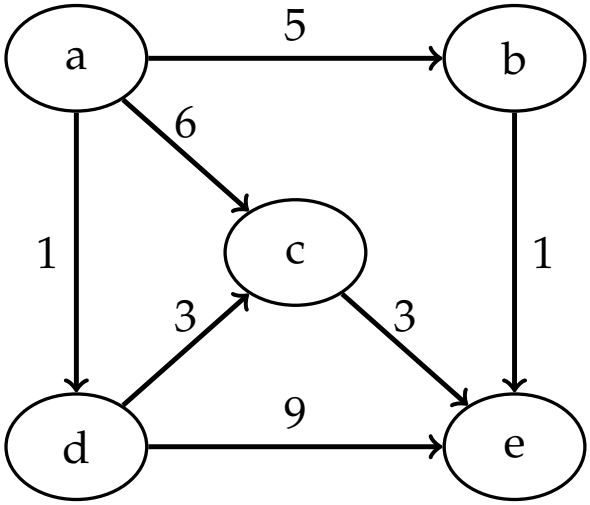
Please note the honor code policy regarding this Assignment. You may not discuss particular questions or discuss or transmit answers from the assignment with other people, except for the MET CS 566 teaching team.

Please submit to the Gradescope!

Tasks

1. **Graph Representation, Dijkstra's Algorithm (3 points)**:

   Consider the following directed Graph.



   - Represent the above graph in Adjacent Matrix format. (1 point)
   - Run the Dijkstra's algorithm on the above directed graph, using **vertex "a" as the start source**. Write down your steps and describe it briefly. (2 points)

2. **Depth-First Algorithm (4 points)**:

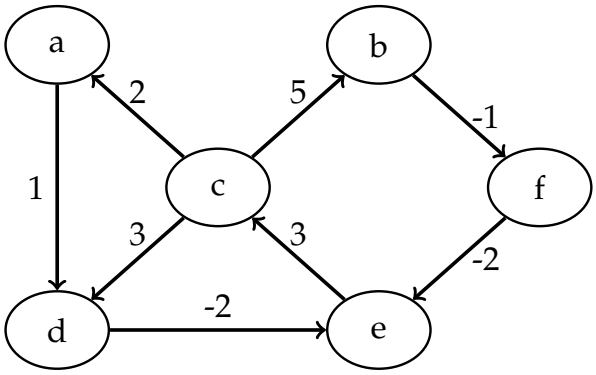
   Consider the directed Graph from task 1 with the weights.

   - Start from the **vertex "a"** and apply the Depth-First Algorithm. Write your steps and describe them briefly. (2 points)
   - Do you have any Back, Forward or Cross Edges? (2 points)

3. **Edge Classification (3 points)**: As we have learned in the lecture, when we apply Depth-First algorithm on graphs, it is possible to have Back, Forward or Cross Edges.

   - Which kind of edges (Back, Forward or Cross Edges) is possible on **undirected graphs**? Describe your answer.

4. **Design an Algorithm (4 points)**: Design an optimal algorithm that can detect cycles in a given directed graph $G = (V, E)$.
   - Provide Pseudocode for your algorithm
   - Describe the running time of your algorithm.
5. **Bellman-Ford Algorithm (4 points)**: Consider the following Graph.



   - Run the Bellman-Ford algorithm on the above directed graph, using **vertex "a" as the start source**. Write down your steps and describe it briefly.
6. **Bellman-Ford algorithm (2 points)**:
   - Describe why Bellman-Ford algorithm does not work when the given graph includes negative cycles.
   - Describe how the Bellman-Ford algorithm detects the negative cycles. Provide an example graph with negative cycles and show how it can be detected.