This homework is due on gradescope Friday February 5th at 11:59pm pacific time. Remember to justify your work even if the problem does not explicitly say so. Writing your solutions in \LaTeX is recommended though not required.

**Question 1** (Public Transit on a Budget, 40 points). Lars is trying to get around town. He has various options for transportation with the possible routes represented by edges of a directed graph $G$. Each edge $e$ has a positive integer cost $\text{cost}(e)$ dollars and a time it takes to traverse $\text{time}(e)$. Lars has a limited number $N$ of dollars and would like to get between two locations ($s$ and $t$) in as little total time as possible.

(a) Give an algorithm that given $G, s, t$, the functions $\text{cost}$ and $\text{time}$ and the total budget $N$, determines the shortest time to get from $s$ to $t$ under the budget. For full credit your algorithm should run in time $O(N(|V| + |E|))$ or better. [20 points]

(b) Suppose that some routes are allowed to have a cost of 0. Provide an algorithm to solve the new version of this problem with runtime $O(N(|V| \log(|V|) + |E|))$ or better. [20 points]

**Question 2** (Negative Cycle Finding, 35 points). We know how to use Bellman-Ford to determine whether or not a weighted, directed graph $G$ has a negative weight cycle. Give an $O(|V||E|)$ time algorithm to find such a cycle if it exists. Hint: If there is such a cycle use Bellman-Ford to find a vertex $v$ with $\text{dist}_{|V|-1}(v) > \text{dist}_{|V|}(v)$ and compute the paths involved. From this you should be able to find a cycle. You may also need to modify your graph some to deal with the possibility of a negative weight cycle not reachable from your chosen starting vertex $s$.

**Question 3** (Divide and Conquer Recursions, 25 points). Give the asymptotic runtimes of the following divide and conquer algorithms.

(a) An algorithm that splits the input into two inputs of a two-thirds the size and then does $\Theta(n)$ extra work. [2 points]

(b) An algorithm that splits the input into five inputs of half the size and then does $\Theta(n^{5/2})$ extra work. [2 points]

(c) An algorithm that splits the input into four inputs of half the size and then does $\Theta(n^2)$ extra work. [2 points]

(d) An algorithm that splits the input into six inputs of a third the size and then does $\Theta(n^{3/2})$ extra work. [2 points]

(e) An algorithm that splits the input into two inputs of a third the size and then does $\Theta(n)$ extra work. [2 points]

(f) An algorithm that splits the input into two inputs of half the size and then does $\Theta(n \log(n))$ extra work. Note: you cannot use the Master Theorem in this case. You may have to do some work to derive the answer. [15 points]

**Question 4** (Extra credit, 1 point). Approximately how much time did you spend working on this homework?