This homework is due on gradescope Friday November 2nd at 11:59pm. 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** (Verifying Shortest Path Lengths, 30 points). Let $G$ be a directed, weighted graph with potentially negative edge weights, but no negative weight cycles. Let $s$ be a vertex of $G$. Using Bellman-Ford to compute the shortest path lengths from $s$ to the other vertices is somewhat slow, however verifying that the answer you have is correct is somewhat faster. Provide an algorithm that given values $d(v)$ for each vertex $v \in V$, determines whether or not $d(v)$ is the length of the shortest path from $s$ to $v$ for all $v$ (your algorithm should return a single TRUE/FALSE value, returning FALSE if there is any vertex $v$ so that $d(v)$ is not the correct shortest path length from $s$). For full credit your algorithm should run in linear time or better.

**Question 2** (The Skyline Problem\footnote{\texttt{sdctf[N1ce_d0rKiNG_C@pt41N]}} 30 points). A city’s skyline is traced out by $n$ rectangular buildings. In particular, each building has a height $h_i$ and a base given by an interval $[x_i, y_i]$. For our purposes we will assume that the $x_i$ and $y_i$ used by the $n$ buildings are exactly the integers from 1 to $2n$ without any repeats. The height of the skyline at location $\ell_i$ at a position $i$ is given by $\max_{x_i \leq \ell \leq y_i} h_i$. Give an algorithm to compute $\ell_1, \ell_2, \ldots, \ell_{2n}$. For full credit, your algorithm should run in time $O(n \log(n))$ or better. Hint: use divide and conquer. Your subproblems may need to look slightly more complicated than your original problem.

**Question 3** (Densest Interval, 30 points). Given a set $S$ of $n$ real numbers and a positive real number $L$, give an algorithm to compute an interval of length at most $L$ that contains as many elements of $S$ as possible. For full credit, your algorithm should run in time $O(n \log(n))$ or better.

**Question 4** (Divide an Conquer Runtimes, 10 points). For each of the following, give the asymptotic runtime of the following divide and conquer algorithms:

(a) An algorithm that splits a problem of size $n$ into five problems of size $n/3$ and does $O(n^{3/2})$ work combining the answers.

(b) An algorithm that splits a problem of size $n$ into eight problems of size $n/2$ and does $O(n^3)$ work combining the answers.

(c) An algorithm that splits a problem of size $n$ into three problems of size $2n/3$ and does $O(n^3)$ work combining the answers.

(d) An algorithm that splits a problem of size $n$ into one problem of size $99n/100$ and does $O(1)$ work combining the answers.

(e) An algorithm that splits a problem of size $n$ into two problems of size $2n$ and does $O(n)$ work combining the answers.

**Question 5** (Extra credit, 1 point). Approximately how much time did you spend on this homework?