CS 4720/5720 Design and Analysis of Algorithms
Homework #5a
Daniel Frey

Answers to homework problems:

1. ***Comparing Minimum Spanning Tree Algorithms***

   (a) *Prim's Algorithm* $\Theta(|V|^2)$*: weight matrix, priority queue as un-ordered array*
   Loop through the vertices not yet included in the minimum spanning tree on each iteration. Furthermore, it will again loop through each vertex connected to the current minimum and see if there is a better edge to choose, also on each iteration. **This gives a time complexity of** $\sum_{i=0}^{V-1}\sum_{j=0}^{V-1} = (V)(V) = V^2 \in \Theta(V^2)$.

   (b) *Prim's Algorithm* $\Theta(|E|\log|V|)$*: adj. list, priority queue as min-heap*
   Loop through each vertex, extracting it from the heap. Since each deletion is $\Theta(\log V)$, this gives $\Theta(V\log V)$. Then you must loop through adjacency lists, updating the heap if the new edge is a lower value. This gives $\Theta(2E\log V)$, where $\log V$ is the heap operation $\Rightarrow \Theta(E\log V)$ $\therefore$ **Time complexity is** $\Theta(E\log V + V\log V) = \Theta(E\log V)$.

   (c) *Sparse Graph* $|E| \in \Theta(|V|)$
   $\Theta(E\log E)$**: Kruskal with adjacency list and fast sort *OR*** $\Theta(E\log V)$**: Prim with adjacency list and min-heap** since E grows the same as V, these would end up being the same.

   (d) *Moderate Edge Graph* $|E| \in \Theta(|V|\log|V|)$
   $\Theta(E\log V)$**: Prim with adjacency list and min-heap** because E will eventually grow faster than V.

   (e) *Dense Graph* $|E| \in \Theta(|V|^2)$
   $\Theta(V^2)$**: Prim with adjacency matrix and un-ordered** since the ratio of edges to vertices is high, and edges are not accounted for in this time complexity.