1a. Worst case of Kruskal’s is O(n*m) because I have the algorithm iterate at max n-1 times which goes over the complete edge list which is length m;

1b. Worst case for my Prim’s algorithm is similar. It iterates at max n-1 times which goes over the complete Edge list looking for a particular edge which is length m thus O(n*m);

2. Kruskal’s algorithm will be best when having a large number of edges this is because it’ll just find the lowest cost edge and be done. Prim’s algorithm will be best when there are fewer than n-1 edges because otherwise the graph will be disconnected an not work.

3. Kruskal’s algorithm fails at any graph where the graph is not fully connected. Prim’s may fail if a greedy approach is not optimal. For example lets say it chooses Edge X to continue forward and then Edges Y and Z. But if Prims chose a different edge in the beginning it may be able to cut down using Edge Y which might be very large.