Exercise 1: MST Algorithm (4 points)

You are given a graph with average degree $\Theta\left(\sqrt{\log(n)}\right)$ and all edge weights are integers between 1 and 10. Does Kruskal’s or Prim’s algorithm have the better runtime? Explain your solution.

Exercise 2: Union-Find (3+3 points)

(a) In the lecture the union-by-size and the union-by-rank heuristics were introduced to implement a union-find data structure. The rank is basically the height of the tree. This is not true if we use path compression as the height of trees changes; but the rank is still an upper bound on the actual height of the tree. Give pseudo code for `union(x,y)`.

Show that when implementing a union-find data structure by using disjoint-set forests with the union-by-rank heuristic, the height of each tree is at most $O(\log n)$.

(b) Demonstrate that the above analysis is tight by giving an example execution (of merging $n$ elements in that data structure) that creates a tree of height $\Theta(\log n)$. Can you even get a tree of height $\lceil \log_2 n \rceil$?