

Homework 3

Question 1. Without path compression, what is the runtime of the union and find operation? (10 points)

Without path compression, the runtime of the union and find operation is $O(n)$

Question 2. Draw out the tree structure and array values (for that type of implementation) of disjoint-set/union-find without path compression after the following operations. (10 points)

`makeset(A..L)` (shorthand for running on each node A to L)

`Union(C,K)`

`Union(F,E)`

`Union(A,J)`

`Union(A,B)`

`Union(C,D)`

`Union(D,I)`

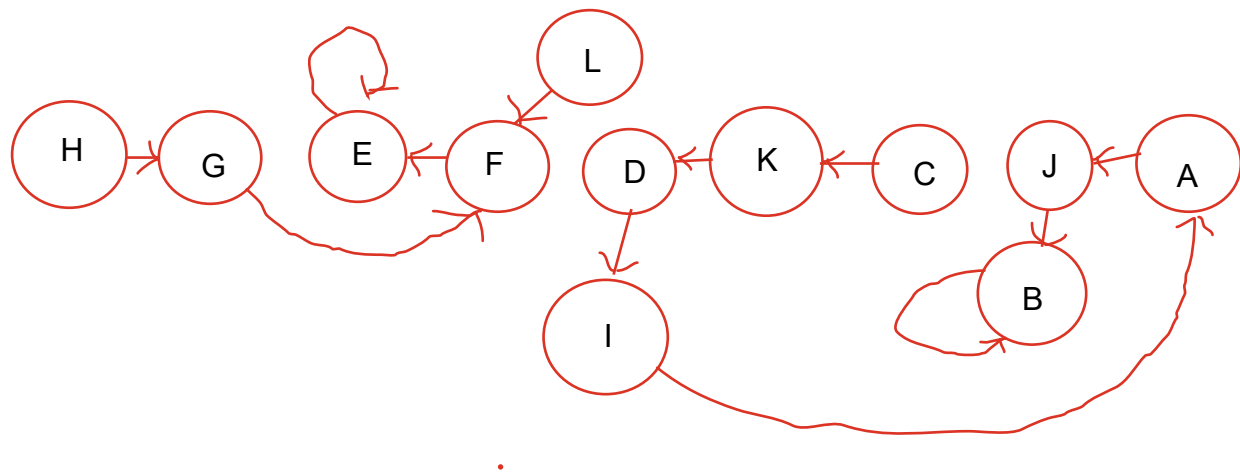
`Union(L,F)`

`Union(C,A)`

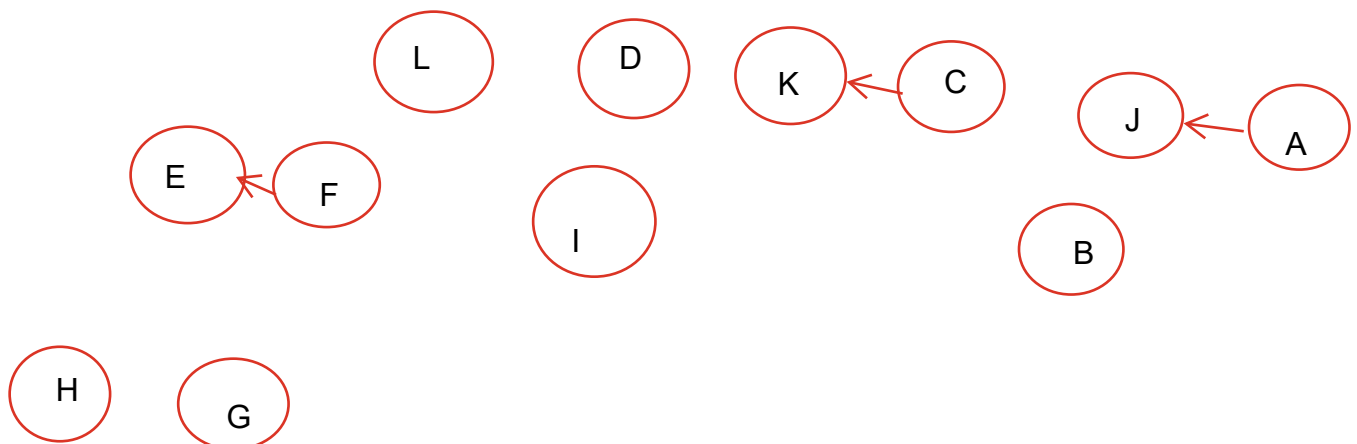
`Union(A,B)`

`Union(H,G)`

`Union(H,F)`



Question 3. Do the same with path compression. (10 points)



Question 4. What abstract data structure does Prim's algorithm use and name at least two implementations of that abstract data structure. (10 points)

Prim's algorithm uses Priority Queue abstract data structure. Two implementations are binary heap and Fibonacci heap

Question 5. Implement the DisjointSet class in homework2.py with path compression. (30 points)

Question 6. Implement Kruskal's algorithm using your union-find/disjoint set functions and draw the minimum spanning tree using networkx. (30 points)