CS 5970/6970: Graph Algorithms

September 1, 2023

**Homework 2**

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

**Without path compression the runtime of the operations union and find are both O(n).**

**Question 2.** Draw out the tree structure and array values (for that type of implementation) of disjointset/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)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| nodes | A | B | C | D | E | F | G | H | I | J | K | L |
| index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| value | 2 | 0 | 2 | 2 | 5 | 11 | 7 | 7 | 2 | 0 | 2 | 7 |

A diagram of a diagram

Description automatically generated with medium confidenceUnion(H,F)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| nodes | A | B | C | D | E | F | G | H | I | J | K | L |
| index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| value | 2 | 2 | 2 | 2 | 5 | 7 | 7 | 7 | 2 | 0 | 2 | 7 |

A black and white diagram

Description automatically generated**Question 3.** Do the same with path compression. (10 points)

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Push** | **Extract-Min** | **Decrease Key** |
| **Linked List** | **O(1)** | **O(n)** | **O(1)** |
| **Binary Heap** | **O(log n)** | **O(log n)** | **O(log n)** |
| **Binomial Heap** | **O(1) amortized** | **O(log n)** | **O(log n)** |
| **Fibonacci Heap** | **O(1)** | **O(log n) amortized** | **O(1)** |

**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 a priority queue which can be implemented with a linked list, binary heap, binomial heap, and Fibonacci heap. Below is a table of the different asymptotic notations for the functions push, extract-min, and decrease key.**

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

**See homework2.py**

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

**See homework2.py**