Interview Questions: Union-Find

Help Center

Warning: The hard deadline has passed. You can attempt it, but **you will not get credit for it**. You are welcome to try it as a learning exercise.

These interview questions are for your own enrichment and are not assessed. If you click the *Submit Answers* button, you will get a hint.

☐ **In accordance with the Coursera Honor Code, I (David Resnick) certify that the answers here are my own work.**

Question 1

Social network connectivity. Given a social network containing N members and a log file containing M timestamps at which times pairs of members formed friendships, design an algorithm to determine the earliest time at which all members are connected (i.e., every member is a friend of a friend of a friend ... of a friend). Assume that the log file is sorted by timestamp and that friendship is an equivalence relation. The running time of your algorithm should be $M \log N$ or better and use extra space proportional to N.

Question 2

Union-find with specific canonical element. Add a method `find()` to the union-find data type so that `find(i)` returns the largest element in the connected component containing `i`. The operations, `union()`, `connected()`, and `find()` should all take logarithmic time or better.

For example, if one of the connected components is $\{1, 2, 6, 9\}$, then the `find()` method should return 9 for each of the four elements in the connected components because 9 is larger 1, 2, and 6.

Question 3

Successor with delete. Given a set of N integers $S = \{0, 1, \ldots, N-1\}$ and a sequence of requests of the following form:

- Remove x from S
- Find the *successor* of x: the smallest y in S such that $y \geq x$.

design a data type so that all operations (except construction) should take logarithmic time or better.

Question 4

Union-by-size. Develop a union-find implementation that uses the same basic strategy as weighted quick-union but keeps track of tree height and always links the shorter tree to the taller one. Prove a $\lg N$ upper bound on the height of the trees for N sites with your algorithm.

☐ **In accordance with the Coursera Honor Code, I (David Resnick) certify that the answers here are my own work.**

Submit Answers Save Answers

You cannot submit your work until you agree to the Honor Code. Thanks!