15-451/651 Algorithm Design & Analysis, Spring 2026

Quiz 3 Solutions

Problem: The input is a graph with n vertices (and initially no edges) whose vertices are colored either red or blue. Give an algorithm that supports the following operations.

1. AddEdge(u, v): Add an undirected edge between vertices u and v in the graph,
2. Query(): Return the number of pairs of vertices $u \neq v$ in the graph that are the same color and are in the same connected component (i.e., have a path between them).

Desired runtime: Over m operations, total time $O((m+n)\log n)$, equivalently amortized $O(\log n)$ per operation.

We will utilize a union-find data structure. Each vertex maintains its parent in the tree. Additionally, the roots of each subtree contain the following information:

Data at each root Size of the subtree, as well as the number of red/blue vertices in its subtree. Call these $\mathsf{red}[v]$ and $\mathsf{blue}[v]$ respectively.

AddEdge Let's say the edge insertion is from $u' \to v'$, where u' is the root of u's component and v' is the root of v's component. Then the parent of u' is set to v'. We can update

$$\mathsf{red}[v'] \leftarrow \mathsf{red}[v'] + \mathsf{red}[u'] \quad \text{and} \quad \mathsf{blue}[v'] \leftarrow \mathsf{blue}[v'] + \mathsf{blue}[u'].$$

Query. We maintain a global counter of the number of pairs.

When the edge $u' \to v'$ is added, we this global counter by:

$$\binom{\mathsf{red}[v'] + \mathsf{red}[u']}{2} + \binom{\mathsf{blue}[v'] + \mathsf{blue}[u']}{2} - \left(\binom{\mathsf{red}[v']}{2} + \binom{\mathsf{blue}[v']}{2}\right) - \left(\binom{\mathsf{red}[u']}{2} + \binom{\mathsf{blue}[u']}{2}\right).$$

There are many other ways to do this, e.g., counting the number of "newly created pairs", which is:

$$\mathsf{red}[v']\mathsf{red}[u'] + \mathsf{blue}[v']\mathsf{blue}[u'].$$

You can check that this is the same as above.

Runtime Updating the additional data take $O(1)$ time, so the cost per operation is $O(\log n)$ by many variants of union find.