

**Rahul Bhiwande**  
**Program Structures & Algorithms**  
**Spring 2021**  
**Assignment No. 3**  
**Alternative**

- **Task –**

We mentioned two alternatives for implementing Union-Find:

1. For weighted quick union, store the depth rather than the size.
2. For weighted quick union with path compression, do two loops, so that all intermediate nodes point to the root, not just the alternates.

For both of these, code the alternative and benchmark it against the implementation in the repository. You have all of that available from a previous assignment.

If you can explain why alternative #1 is unnecessary to be benchmarked, you may skip benchmarking that one.

- **Evidence to support the conclusion:**

Benchmark of alternative 2 –

Above 5 values are for UF with path compression where alternates are pointing to the root. Below 5 values are for UF with path compression where every intermediate node is pointing to the root

```
/Library/Java/JavaVirtualMachines/jdk-15.0.2.jdk/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=52707:/Applications/I
2021-02-28 13:44:02 INFO Benchmark_Timer - Begin run: Number of nodes 2000 with 50 runs
0.44
2021-02-28 13:44:02 INFO Benchmark_Timer - Begin run: Number of nodes 4000 with 50 runs
0.7
2021-02-28 13:44:02 INFO Benchmark_Timer - Begin run: Number of nodes 8000 with 50 runs
0.78
2021-02-28 13:44:02 INFO Benchmark_Timer - Begin run: Number of nodes 16000 with 50 runs
1.46
2021-02-28 13:44:02 INFO Benchmark_Timer - Begin run: Number of nodes 32000 with 50 runs
3.28
-----
2021-02-28 13:44:03 INFO Benchmark_Timer - Begin run: Number of nodes 32000 with 50 runs
0.42
2021-02-28 13:44:03 INFO Benchmark_Timer - Begin run: Number of nodes 32000 with 50 runs
0.42
2021-02-28 13:44:03 INFO Benchmark_Timer - Begin run: Number of nodes 32000 with 50 runs
0.64
2021-02-28 13:44:03 INFO Benchmark_Timer - Begin run: Number of nodes 32000 with 50 runs
1.36
2021-02-28 13:44:03 INFO Benchmark_Timer - Begin run: Number of nodes 32000 with 50 runs
3.2

Process finished with exit code 0
```

Proof why benchmarking alternative 1 isn't necessary –  
 As the upperbound is same for both regardless of depth or size.

$$\rightarrow \text{Invariant: Height} = h$$

$$\text{Nodes} = 2^{h-1}$$

weighted by size:  $h = 1$

$$\text{Nodes} = 2^{1-1} = 2^0 = 1$$

$$h = 2$$

$$\text{Nodes} = 2^{2-1} = 2^1 = 2$$

$$h = 3$$

$$N \geq 2^{3-1} = 2^2 = 4$$

$$h = 4$$

$$N \geq 2^{4-1} = 2^3 = 8$$

$$\therefore h = \lg N + 1$$

weighted by height,

$$h = 2, N \geq 2$$

$$h = 3, N \geq 4$$

$$h = 4, N \geq 8$$

$$h = \lg N + 1$$

