Shaun Boerner

Textbook Questions and Solutions

Q: 1.5.1 Show the contents of the id[] array and the number of times the array is accessed for each input pair when you use quick-find for the sequence 9-0 3-4 5-8 7-2 2-1 5-7 0-3 4-2.

A:

0 1 2 3 4 5 6 7 8 9   
0 1 2 3 4 5 6 7 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 2 3 3 5 6 7 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 2 3 3 8 6 7 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 2 3 3 8 6 2 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 1 3 3 8 6 2 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 1 3 3 2 6 2 2 0 two access

0 1 2 3 4 5 6 7 8 9   
3 1 1 3 3 2 6 2 2 0 one access

0 1 2 3 4 5 6 7 8 9   
3 1 3 3 3 2 6 2 2 0 two access

Q: Do Exercise 1.5.1, but use quick-union (page 224). In addition, draw the forest of trees represented by the id[] array after each input pair is processed.

A:

0 1 2 3 4 5 6 7 8 9   
0 1 2 3 4 5 6 7 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 2 4 4 5 6 7 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 2 4 4 8 6 7 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 2 4 4 8 6 2 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 1 4 4 8 6 2 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 1 4 4 2 6 2 2 0 two access

0 1 2 3 4 5 6 7 8 9   
3 1 1 4 4 2 6 2 2 0 one access

0 1 2 3 4 5 6 7 8 9   
3 1 3 4 2 2 6 2 2 0 two access

Q: 1.5.3 Do Exercise 1.5.1, but use weighted quick-union (page 228).

A:

0 1 2 3 4 5 6 7 8 9   
0 1 2 3 4 5 6 7 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 2 3 3 5 6 7 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 2 3 3 8 6 7 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 2 3 3 8 6 2 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 1 3 3 8 6 2 8 0 one access

0 1 2 3 4 5 6 7 8 9   
0 1 1 3 3 2 6 2 2 0 two access

0 1 2 3 4 5 6 7 8 9   
3 1 1 3 3 2 6 2 2 0 one access

0 1 2 3 4 5 6 7 8 9   
3 1 3 3 3 2 6 2 2 0 two access

Q: 1.5.5 Estimate the minimum amount of time (in days) that would be required for quick-find to solve a dynamic connectivity problem with 109 sites and 106 input pairs, on a computer capable of executing 109 instructions per second. Assume that each iteration of the inner for loop requires 10 machine instructions.

A:

(106\*log^109/10^9)/60/60/24 = 3,135,232,474,272,650,764,400.83 days

Q: 1.4.38 Naive 3-sum implementation. Run experiments to evaluate the following implementation of the inner loop of ThreeSum: for (int i = 0; i < N; i++) for (int j = 0; j < N; j++) for (int k = 0; k < N; k++) if (i < j && j < k) if (a[i] + a[j] + a[k] == 0) cnt++; Do so by developing a version of DoublingTest that computes the ratio of the running times of this program and ThreeSum.

A: Original program 8000 in 20.7, new loop did it in 16.1. New loop is 33% faster.

Q: