C++ QUESTIONS

ROMIL PUNETHA·THURSDAY, MAY 5, 2016

Question Profiles

- http://www.spoj.com/users/milind11/
  Ques worth doing out of the 116 mentioned (still updating):
  1. ABCDEF (Type- Ad hoc, STL, BS) (Good ques. A little initial thought. Then use of STL worked. Static hash mapping instead of dynamic.)
  2. ABCPATH
  3. AGGRCOWS (Type- BS) (V.Good BS question)
  4. ACODE
  5. MCOINS

Binary Search

1. http://www.spoj.com/problems/AGGRCO... (DONE)
2. https://www.interviewbit.com/proble... (DONE)
3. https://www.interviewbit.com/proble... (DONE)

DP Essentials

1. https://leetcode.com/problems/maximum-product-subarray/ (DONE)
2. https://leetcode.com/problems/dunge... (DONE) (Simple though, just a slight thought on how to start... rest is normal DP)

Sqrt Decomposition

1. http://www.spoj.com/problems/DQUERY... (DONE)
2. https://www.hackerrank.com/contests... (DONE)
3. http://codeforces.com/contest/617/p... (MO's Algorithm)
4. http://codeforces.com/contest/86/pr...

5. https://www.codechef.com/MARCH14/pr...
6. http://codeforces.com/problemset/pr...
7. https://www.codechef.com/problems/I...

Fast matrix exponentiation

1. https://www.hackerearth.com/problem... (DONE)
2. https://www.hackerrank.com/contests... (DONE)
3. https://www.hackerearth.com/april-e...
4. https://www.hackerearth.com/april-e...

Prefix-Suffix Array

1. http://codeforces.com/contest/660/p...
2. https://www.codechef.com/SNCKQL16/p...

Sliding window

1. https://www.codechef.com/SNCKQL16/p...
2. http://www.codeforces.com/contest/7...

Graph Coloring

1. https://www.hackerearth.com/problem/algorithm/containers-of-choclates-1/description/

Segment Tree

1. https://www.hackerearth.com/practic...

2. https://www.hackerearth.com/code-mo…

Modular Arithmetic

1. **http://codeforces.com/problemset/problem/490/C**

Hashing

1. **http://codeforces.com/contest/463/submission/18291034**

Binary Indexed Tree

1. **https://www.hackerearth.com/practice/data-structures/advanced-data-structures/fenwick-binary-indexed-trees/practice-problems/1/?sort_by=partially%20solved&p_level=**

Maths

1. **https://www.hackerearth.com/practice/math/number-theory/totient-function/practice-problems/algorithm/phi-phi-phi/ (Pollard-rho algo, Miller-rabin Primality Test, Euler's Totient function)**

Tutorials

1. LCA reduction to RMS : https://www.topcoder.com/community/…

Though there's an easier way if you define the node structure properly. Initialize a 'parent' variable as well in the node structure. Then when performing DFS, give a ‘timer’ or visiting order to each node. Then traverse to one of the nodes whose LCA is to be found. Then using the parent link, traverse up the link until the timer of the parent < timer of the other node. When this condition becomes false, you’ll be on the node that probably will be the LCA. From this node, traverse the other child (suppose you reached this node by traveling up the right child, so now traverse the left child downwards) and search for the other node. If found, then that parent node is the LCA of the given two nodes. Complexity= O(N)

The LCA algo runs in O(log N) for a BST.

1. Binary Indexed Trees : https://www.topcoder.com/community/...