The Theory of Computation

Bernard M.E. Moret

Addison-Wesley-Longman, 1998

Solution Manual

Chapter 2, Problem 2.26

The key insight is to realize that there cannot be any advantage to not matching a leaf. Of course we may not be able to match all leaves—whenever two or more leaves share the same "parent" (neighbor), at most one of these leaves can be matched, but there is no gain in leaving them all unmatched. This gives rise to a simple recursive strategy: recursively traverse the tree (depth-first search); on backing out of a leaf, if its parent is not matched, then match it with the leaf; in general, in backing our of an unmatched node, if its parent is not matched, then match it with the node. This algorithm clearly runs in linear time. It is optimal because the principle of labeling leaves whenever possible (which is easily seen never to hurt) is applied recursively by the algorithm: having matched what leaves is could, it then removes from consideration all edges touching matched vertices and effectively starts again matching what newly created leaves it can.