BU CLA CS 530: Analysis of Algorithms

Spring 1996

Problem Set 9

Due Thursday 96.03.28, at 11:00 am, in class. Late submissions not accepted.

---

1. Exercise 29.2-8, page 671, in [CLR].

2. Consider the search problem in which a given ordered list of $n \geqslant 1$ items (with possible duplicates) is to be searched for the index of an item equal to a given value X, if any such index exists. Binary search solves this problem in $\mathcal{O}(\log n)$ sequential time.

   (a) Design a time-efficient CREW parallel algorithm for this problem, using as many processors as you need.

   (b) What is the parallel running time of your algorithm? Is your algorithm work-efficient? Justify.

3. Design a CREW algorithm for finding the preorder numbering of a binary tree with $n \geqslant 1$ nodes, in $\mathcal{O}(\log n)$ parallel time using n processors. The algorithm should label each node of the tree with the number of nodes visited before it in a preorder traversal. Use the method of *pointer jumping*. For simplicity, assume the binary tree is *full*, i.e. every node which is not a leaf has exactly 2 children.
   (Exercise 30.1-6, page 701 in [CLR], asks the same question but does not restrict the number of processors to be n. Without this restriction, a solution based on the *Euler-tour technique* is the simplest — see Problem 1 in Homework Set 10 and its solution, from Spring 1995.)

4. Let $G = (V, E)$ be a connected undirected acyclic graph, i.e. G is just a tree. Design a linear-time (sequential) algorithm that finds an edge in G whose removal disconnects G into two disjoint subgraphs whose size difference is as small as possible, i.e. if the two subgraphs are $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ with $|V_1| \leqslant |V_2|$, then $|V_2| - |V_1|$ is minimized.
   Hint: The solution of Problem 5 in Homework Set 4 should inspire you.

Created: 96.03.20
Modified: 96.04.02