18.435/6.848 Theory of Parallel and VLSI Computation	Fall 1997

Problem Set #6

Due: Wednesday, December 3, 1997 — *Prof. Leighton*

Reading Lecture Notes

Problems:

1. Show that there is a permutation of $2N$ items for which the BBN routing algorithm on a $\log N$-dimensional butterfly (with edge capacity 1) will only deliver 2 messages if the worst-case tie-breaking choices are made at every node. If the best-case tie-breaking decisions are made, show that the throughput is $\Theta(\frac{1}{\sqrt{N}})$ for the worst-case permutation. (Note that you should proved both the upper and lower bounds.)

2. Show that the congestion bound proved in class for the butterfly (based on a random destination for each packet) holds for the CM-5 automatic load-balancer algorithm for any p-relation.

3. Consider a distributed-memory optical model of computing in which any processor can communicate with any other processor in a single step, *except* if more than one processor tries to communicate with another processor, then none of them get through. Collisions are not detected at the receiving end (i.e., a processor can read or write to a location if and only if it is the only processor that tries to access that location). In addition any processor can broadcast a single value to any subset of the other processors. Show how to compute a global OR of N-bits in $O(1)$ steps using a N-processor optical machine. (Hint: Try to identify the first bit that is 1, if it exists.) Then show how to compute the minimum of N numbers in $O(\log\log N)$ steps with an N-processor optical machine.

4. Consider an ordered list of N elements that has each element marked with a bit x that is either 0 or 1. Describe an efficient EREW PRAM algorithm to partition the list into two sublists, one consisting of all elements with $x = O$, and one of those with $x = 1$. The order in the sublists should be consistent with the order in the original list. (Your algorithm should take $O(logN)$ steps using $N/logN$ processors.)

5. Devise an $O(log^*N)$-step algorithm for k-coloring the nodes of an N-node binary tree (stored arbitrarily in memory by pointers) on an N-processor EREW PRAM, where $k = O(1)$.

6. The *diameter* of a tree is the length of the longest path of the tree. A *center* is a vertex v such that the longest path from v to a leaf is minimal over all vertices in the three. Show how to use treefix computations to find the diameter and a center of a binary tree. (Hint: First find the farthest leaf from the root.) (5.15)

*7. Show that evaluating straightline code containing only sentences of the form $x_i = T$, $x_i = F$, $x_i = x_j \vee x_k$ and $x_i = x_j \wedge x_k$ is P-complete. Such codes are called *monotone* since negation is not allowed.

8. Show that $BT \geq \Omega(N)$ for the problem of deciding if inputs $x_1, \ldots, x_N$ are a cyclic shift of inputs $y_1, \ldots, y_N$. There is just one output bit z which is 1 if and only if the x bits are a shift of the y bits. The shift is not specified and you may not make any restrictive assumptions about the location of the input bits.

*9. Find operations $\oplus$ and $\otimes$ for which $BT = o(N)$, but for which the composition has complexity $BT = \Omega(N)$.

10. Show that $AT^2 \geq \Omega(N^2k^2)$ for the problem of computing the convolution of N, k-bit numbers.

11. Design an $O(logN)$-time cyclic shifter that achieves optimal AT^2 performance up to a constant factor.

*12. Show that any layout of the N-node complete binary tree with all its leaves on a line at the bottom of the layout must have $\Omega(NlogN)$ area. Can you figure out how long the longest wire must be in any such layout?

13. Let A denote the minimum layout area of some N-node network, and let W denote its active (or wire) area. Show that $A \leq O(Wlog^2(N/\sqrt{W}))$, and thus that an optimal chip must be fairly dense with processors and/or wire.

*14. Extend the result of the problem 5 to show that $A \leq O(Wlog\frac{N}{\sqrt{W}})$. Can you improve this bound further?

$$ 15. **Extra Money** – Consider a $\log N$-dimensional butterfly with $\log N$ packets/input, with destinations chosen randomly. Prove or disprove that for a greedy algorithm in which a packet advances whenever the next edge is available and the *constant size* queue at the other end of the edge is not full, (with a tie-breaking scheme of your choice) the expected completion time is $O(\log N)$.