Problem Set #1
IE 495
Due September 11

Written Problems

1. Explain how you could simulate a shared memory model of computation using a distributed memory model with one additional processor. [Hint: What does the interconnection network look like?]

   How well do you think this scheme would work in practice? Hint: Consider the time required to perform a semi−group operation.

2. *Explain some of the potential pitfalls of asymptotic analysis. Do you think that the models of computation we've discussed in class are realistic and/or useful? Why or why not?

3. *In an effort to design more realistic models of computation, Parberry suggests that interconnection schemes be limited to those that satisfy the following three properties:

   – Each processor has a constant number of all−purpose registers.
   – The degree of the network is a constant.
   – The interconnection scheme can be computed in polynomial time.

   He calls such schemes *feasible networks*. Explain why register access in a feasible network can be limited to exclusive reads only without asymptotic time loss.

4. Consider an interconnection scheme based on a tree, but in which neighboring nodes on each level are connected. That is, each interior node has two additional links, one to each of its left and right neighbors. Nodes on the outer edge of the tree have one additional link. We will call such a scheme an *X−tree*.

   – What is the degree of an X−tree? Explain.
   – What is the communication diameter? Explain.
   – What is the bisection width? Explain.
   – Give a lower bound for sorting on an X−tree. Explain.

5. Prove the following properties of order relations:
   – $f(n) \in O(g(n)) \Leftrightarrow g(n) \in \Omega(f(n))$.
   – $f(n) \in \Theta(g(n)) \Leftrightarrow g(n) \in \Theta(f(n))$.
   – $f(n) \in \Theta(g(n)) \Leftrightarrow (f(n) \in O(g(n))$ AND $f(n) \in \Omega(g(n)))$.
   – $f(n) \in o(g(n)) \Leftrightarrow \lim_{n\to\infty} f(n)/g(n) = 0$.
   – $f(n) \in \omega(g(n)) \Leftrightarrow \lim_{n\to\infty} f(n)/g(n) = \infty$.
   – $f(n) \in o(g(n)) \Rightarrow f(n) \in O(g(n))$, but not the converse.
   – $f(n) \in \omega(g(n)) \Rightarrow f(n) \in \Omega(g(n))$, but not the converse.

6. *Prove or disprove: $O(f(n) + g(n)) = f(n) + O(g(n))$ if $f(n)$ and $g(n)$ are positive for all n.

7. Show that $\sum_{i=1}^{n} i \in O(n^2)$.

8. Prove that as $n \to \infty, (\ln n)^m / n \to 0 \; \forall m \in \mathbb{Z}$. Hint: Use the fact that $e^x = \sum_{i=0}^{\infty} \frac{x^i}{i!}$.

Programming Problems

9. *Give an efficient algorithm to sum *n* values on a hypercube. The algorithm can be described in words and/or with pseudo−code.

10. A matrix is said to have a *saddle point* if some position has the smallest entry in its row and the largest entry in its column. Design an efficient algorithm for determining if a matrix has a saddle point and determining its location if there is one. The algorithm can be described in words and/or given as pseudo−code.

11. An algorithm is called *optimal* or *asymptotically optimal* if it has the same order of magnitude as the fastest possible algorithm.

    Write a program that reads in a set of *n* integers in the range [1,...,100] from a file, sorts them, and writes the result to the standard output.

    Discuss the time and space, complexity of your algorithm, as well as its optimality.

Extra Credit

***What is the etymology of the word "algorithm"?

Problems denoted by a * are optional.