CS 1510 Midterm 4
Fall 1995

1. Give an EREW PRAM algorithm that has constant efficiency with $p = n/\log n$ processors for the problem of finding the maximum of n numbers.

2. Consider the knapsack problem where the input is n positive integers $x_1, \ldots, x_n$ and an integer L. The problem is to determine whether there is a subset of the x_i's that sums to L.

   (a) Is an $O(nL)$ time algorithm for this problem a polynomial time algorithm? Explain.

   (b) The knapsack problem is NP-hard (and NP-complete). Explain what it means for a problem to be NP-hard. That is, assume that you are seeking an algorithm for a problem and then you find out that the problem is NP-hard — what should this tell you?

3. Show that the following problem is NP-hard:

   INPUT: An undirected graph G and an integer k

   OUTPUT: yes, if each vertex of G is of even degree and G contains a vertex cover of size k or less

   Some definitions in case you have forgotten: The degree of a vertex is the number of edges incident to it. A vertex cover is a set U of vertices with the property that every edge in G is incident on a vertex in U.

4. Consider the following problem: The input is a set $\{x_1, \ldots, x_n\}$ of positive integers that sums to L. That is, $\sum_{i=1}^{n} x_i = L$. The problem is to determine if it is possible to add and subtract these numbers to get zero. That is, does there exist a collection of $s_i \in \{-1, 1\}$, $1 \leq i \leq n$, such that

$$\sum_{i=0}^{n} s_i x_i = 0$$

   So for example, if the input set was $\{1, 4, 6\}$ the eight possible sums would be $1 + 4 + 6$, $1 + 4 - 6$, $1 - 4 + 6$, $1 - 4 - 6$, $-1 + 4 + 6$, $-1 + 4 - 6$, $-1 - 4 + 6$, and $-1 - 4 - 6$. None of which happen to be zero in this case. Give an algorithm for this problem whose running time is polynomial in n and L.

5. Design an $O(n \log n)$ time algorithm that given a set $S = \{(x_i, y_i) | 1 \leq i \leq n\}$ of intervals as input, marks all intervals in S that are contained within another interval in S. (x_i, y_i) is contained in (x_j, y_j) if $x_j \leq x_i \leq y_i \leq y_j$. Assume that no pair of intervals share a common endpoint.

6. Consider the following variant of the Towers of Hanoi problem. There are three pegs, A, B, and C. Disks may be moved from peg A to peg B, from peg C to peg B, from peg B to peg A, and from peg B to peg C. Disks may **NOT** be moved from peg A to peg C, or from peg C to peg A. At no time may a larger disk sit atop a smaller disk. The n disks initially reside on peg A. Give an algorithm for moving the n disks to peg B that uses the fewest possible number of moves.