Algorithms Core 15-750 PROBLEM SET # 5 Spring 1998

DUE: 12Noon Friday 10 April 1998

Answer each question on separate sheets of paper for grading purposes. Make sure that each page has your name and email address.

Make sure that each algorithm you consider includes the following three bullets.

DESCRIPTION: A high level description of the algorithm.

CORRECTNESS: At least a high level outline of how the full proof would go. Possibly citing lemmas from class.

TIMING: A timing analysis.

The algorithms will be graded based on what you put in each bullet.

1 Making an edge separator into a vertex separator

Suppose you are given a very good edge separator $S \subseteq E$ for some graph $G = (V, E)$. You are ask to convert it to a vertex separator by picking a minimum number of vertices from the end points of S so as to "cut" all the edges of S by removing at least one endpoint of each edge in S.

Give an efficient algorithm to find such a vertex set.

Hint: Use flow or bipartite matching.

2 Scans and polynomial evaluation

Show how to evaluate a polynomial $f(x) = a_n X^n + \cdots a_0$ in constant time on an EREW PRAM augmented with unit time prefix operators $+$ and$\times$. Note that in unit time you can construct a vector of length n with each equal to the same value, say zero.

3 Ordered traversal of a tree

Suppose that $T = (V, E)$ is a rooted tree embedded in the plane as a planar graph. Thus, the tree T as planar graph has but one face. We shall call the cycle of arcs on that face enumerated in counter-clockwise order an **Euler Tour** of T

Show how using the Euler Tour to find the following orderings of a tree efficiently in parallel. Inorder, Postorder, Preorder. Use these orderings to construct a unit time algorithm which determines if one node is an ancestor of another.

You may assume that the list ranking problem is $O(\log n)$ time and $O(n)$ work. You may mark each arc as either pointing towards the root or away from the root. In the case of inorder tarversl you may assume the there is at most a left and a right child.