Parallel Algorithms	Spring 2022

Problem Set 6

Prof. Nodari Sitchinava — *Due: Wednesday, April 20, 2022 at 10:30am*

You may discuss the problems with your classmates, however **you must write up the solutions on your own** and **list the names** of every person with whom you discussed each problem.

Start **every** problem on a separate page. ***Any problem submitted by 11:59pm Friday April 15, 2022 will receive an additional 10% of the score you receive on that problem.***

1 Computing The Source Vertices of Edges (20 pts)

Given an adjacency array representation of a tree $T = (V, E)$, every entry $E[j]$ stores the destination vertex of the edge (u, v), i.e., $E[j] = v$. Let $n = |V|$ and $m = |E|$. Design an $O(\log n)$-time $O(n)$-work EREW PRAM algorithm that computes for every entry $E[j]$ the source vertex u of this edge (u, v). For every $E[j]$ you may store the computed vertex in either a separate array E' (as $E'[j] = u$) or you may store it within a field of $E[j]$ (as $E[j].from = u$) — the choice is yours. As always, don't forget to prove your algorithm's correctness and analyze its time and work complexities. *Hint: use segmented prefix sums.*

2 Tree Traversal (30 pts)

You are given a binary tree $T = (V, E)$ rooted at vertex r. Let $n = |V|$.

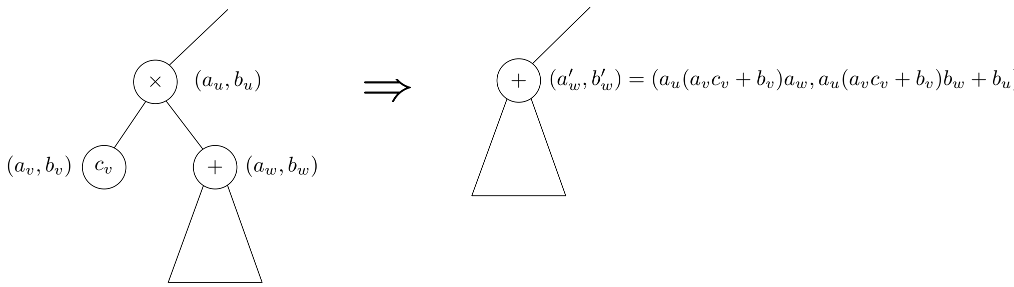
The *preorder traversal* of T consists of traversal of r, followed by the recursive preorder traversal of the left subtree of r, and followed by the recursive preorder traversal of the right subtree of r. Design an $O(\log n)$-time, $O(n)$-work CREW PRAM algorithm that computes for each vertex v its preorder number – the step in which v is traversed during the preorder traversal of T. As always, don't forget to prove your algorithm's correctness and analyze its time and work complexities.

3 Finding The Order of The Leaves (20 pts)

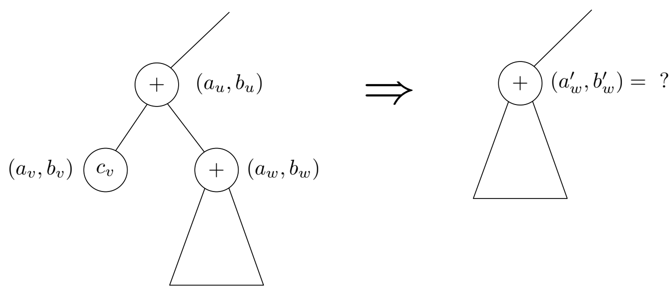
Given a binary tree T, design an $O(\log n)$-time and $O(n)$-work algorithm that labels all the leaves in T, except the left-most one and the right-most one, consecutively in order from left to right and places them into a contiguous array A. As always, don't forget to prove your algorithm's correctness and analyze its time and work complexities. *Hint: use Euler Tour technique.*

4 Expression Tree Evaluation (30 pts)

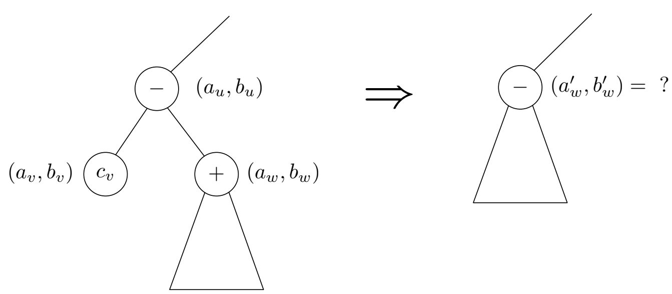
Let $T = (V, E)$ be an expression tree and let v be a leaf, with value c_v, parent u and sibling w. In lecture we saw that each tree vertex x is labeled with a pair of integers (a_x, b_x). We also saw that if the operation at vertex u is multiplication, then if we rake v, the new label at w is modified to (a'_w, b'_w), where $a'_w = a_u(a_v c_v + b_c)a_w$ and $b'_w = a_u(a_v c_v + b_v)b_w + b_u$.



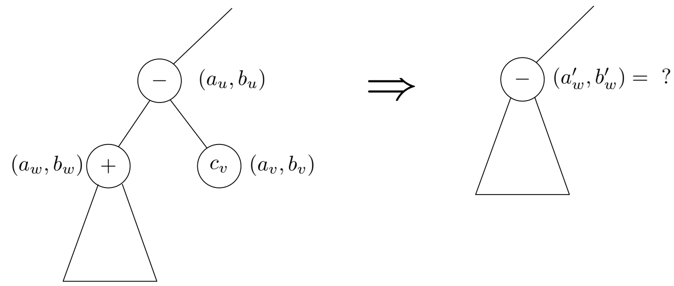
(a) **(10 pts)** What is the new label at w if the operation at u is addition? Show your work. Does it matter if v is the left child of u or the right child of u? Explain why or why not.



(b) **(10 pts)** What is the new label at w if the operation at vertex u is subtraction and v is the left child of u? Show your work.



(c) **(10 pts)** What is the new label at w if the operation at vertex u is subtraction and v is the right child of u? Show your work.


(a_u, b_u)
(a_w, b_w)
c_v
(a_v, b_v)
$\Rightarrow$
$(a'_w, b'_w) =\ ?$