CSE 6220 Homework 2

Karl Hiner, @khiner6

1

Determine if the parallel prefix algorithm can be used to compute prefix sums of a sequence of n numbers based on the binary operation $\bigoplus$ defined as:

(a) $a \bigoplus b = 2a + b$

(b) $a \bigoplus b = \sqrt{a^2 + b^2}$

(Hint: Verify if the operation is associative: $(a \bigoplus b) \bigoplus c? = a \bigoplus (b \bigoplus c)$.)

2

In the game of Photosynthesis, points are given for trees that receive sunlight. Consider n trees $T_0, T_1, ..., T_{n-1}$ planted along a single row of spaces, and sunlight is colinear with this row of trees. The tree placement is modeled by an array A of size n, where $A[i]$ denotes the height of the tree T_i. A tree tall enough to get sunlight exposure scores photosynthesis points according to its height, i.e., T_i is given $A[i]$ points. However, a tree can also be blocked from sunlight by an earlier tree *of equal height or taller*, in which case the blocked tree receives no points.

Design a parallel algorithm to compute the total number of points for a configuration given by A, and compute its runtime.

3

A sequence of nested parenthesis is said to be well-formed if 1) there are an equal number of left and right parenthesis, and 2) each right parenthesis is matched by a left parenthesis that occurs to its left in the sequence. For example, ((( ) ( )) ( )) is well-formed, but ( )) (is not.

There is a nested parenthesis sequence of length n distributed across p processors using block decomposition. Design a parallel algorithm to determine if it is well-formed and specify its run-time.

4

Let A be an array of n elements and L be a boolean array of the same size. We want to assign a unique rank in the range $1, 2, ..., n$ to each element of A such that for any $i < j$:

- If $L[i] = L[j]$, $A[i]$ has lower rank than $A[j]$.
- If $L[i] = 0$ and $L[j] = 1$, $A[i]$ has lower rank than $A[j]$.
- If $L[i] = 1$ and $L[j] = 0$, $A[j]$ has lower rank than $A[i]$.

Design a parallel algorithm to compute the ranks and specify its run-time. (Hint: Think of L as specifying labels. Then, all elements with 0 label receive lower ranks than any element with label 1. Within the same label, ranks are given in left to right order as per array A.)

5

Invent Segmented Parallel Prefix: Segmented parallel prefix is a generalization of the parallel prefix problem where the prefix sums need to be restarted at specified positions. Consider array X containing n numbers and a boolean array B of the same size. We wish to compute prefix sums on X but the sum resets at every position i where $B[i] = 1$. Formally, we wish to compute array S of size n such that

$$S[0] = X[0]$$

$$S[i] = \begin{cases} s[i-1] + X[i], & \text{if B[i] = 0} \\ X[i], & \text{if B[i] = 1} \end{cases}$$

Design parallel segmented prefix algorithm and specify its run-time.

(Hint: The problem can be transformed into a standard prefix sums problem.)

NOTE from Feb. 8 class: Can be solved using matrix raised to power of iteration - see class slides.