Parallel Computing

COMP 633 Fall 2021

Written Assignment #1

<u>Assigned:</u> Thu Sep 2
<u>Due:</u> Tue Sep 16

I. **[7]** The Work-Time (W-T) presentation of EREW sequence reduction (Algorithm 2 in PRAM handout) has work complexity $W(n) = O(n)$ and step complexity $S(n) = O(\lg n)$. Following the strategy of Brent's theorem, the translation of this algorithm will yield a p processor EREW PRAM program with running time

$$T_C(n, p) = O(n/p + \lg n)$$

(a) Construct an alternate sequence reduction algorithm directly for the bare bones EREW PRAM with running time $T_C(n, p) = O(n/p + \lg p)$.

(b) Explain why your solution to (a) cannot be expressed in the W-T model.

II. **[10]** Given a sequence $s[1..n]$, the *maximum contiguous subsequence sum* (mcss)of s is the largest sum that can be formed from any contiguous subsequence of s (including the empty subsequence, with sum zero), i.e.

$$\max_{1 \le i \le j \le n} \left(\operatorname{sum}_{k \in i:j} s_k \right)$$

When all elements of s are positive the *mcss* is the sum of all elements in s. When all elements are negative the *mcss* is zero, corresponding to the sum of an empty subsequence. Here is an optimal sequential algorithm for this problem:

```
integer MCSS(sequence<integer> s)

MaxSoFar, MaxEndingHere ← 0, 0
for i = 1 to n do
   MaxEndingHere ← max(MaxEndingHere + s[i], 0)
   MaxSoFar ← max(MaxSoFar, MaxEndingHere)
enddo
return MaxSoFar
```

Design a work-efficient EREW algorithm in the Work-Time framework with step complexity $\Theta(\lg n)$ for this problem.

III. **[10]** Let A and B be sets of integers with $|A| = m \le n = |B|$. The elements of the sets are stored in increasing order in arrays $A[1..m]$ and $B[1..n]$, respectively (since A and B are sets, there are no duplicate elements in either of these arrays). Using this representation, construct a CREW W-T algorithm that determines whether $A \subseteq B$ in $O(\lg n)$ steps and $O(n)$ work.