CSE 332
Summer 2018
Section 7

1 Parallel Prefix Sum

Given input array $[8, 9, 6, 3, 2, 5, 7, 4]$, output an array such that each output[i] = sum(array[0], array[1], ... array[i]), using the Parallel Prefix Sum algorithm from lecture. Show the intermediate steps. Draw the input and output arrays, and for each step, show the tree of the recursive task objects that would be created (where a node's child is for two problems of half the size) and the fields each node needs. Do not use a sequential cut-off.

2 Parallel Prefix FindMin

Given an input array $[8, 9, 6, 3, 2, 5, 7, 4]$, output an array such that each output[i] = min(array[0], array[1], ... array[i]). Show all steps, as above.

3 Recurrences

1. Show that quicksort with sequential partitioning, but parallel recursive sorting, has $\mathcal{O}(n)$, span by solving the recurrence relation shown in lecture:

$$T(n) = \begin{cases} T(n/2) + c_1 n & \text{if } n \geq 1 \\ c_2 & \text{othwerise} \end{cases}$$

2. Show that a completely parallel quicksort, (i.e. quicksort with parallel partition and recursion) has span $\mathcal{O}(log^2(n))$, by solving the recurrence relation shown in lecture:

$$T(n) = \begin{cases} T(n/2) + c_1 \log n & \text{if } n \geq 2 \\ c_2 & \text{otherwise} \end{cases}$$

.

3. Show that completely parallel mergesort (i.e. mergesort with parallel merging and recursion) has span $\mathcal{O}(\log^3(n))$ by solving the recurrence shown in lecture:

$$T(n) = \begin{cases} T(n/2) + c_1 \log^2(n) & \text{if } n \geq 2 \\ c_2 & \text{otherwise} \end{cases}$$