Name Removed – Assignment 5 – Q1, All parts

1. A. Hillis-Steele scan [1,2,3,4,5,6,7,8]

Underlined = Intermediate/Changed results

Vector: 1 2 3 4 5 6 7 8 Start

Step 1 Vector: 1 3 5 7 9 11 13 15 1 step

Step 2 Vector: 1 3 6 10 14 18 22 26 2 steps

Step 3 Vector: 1 3 6 10 15 21 28 36 4 steps

B. Blelloch Scan [1,2,3,4,5,6,7,8]

Underlined = intermediate/Changed results

Vector: 1 2 3 4 5 6 7 8 Start

Step 1 Vector: 1 3 3 7 5 11 7 15 1 step (‘upsweep’)

Step 2 Vector: 1 3 3 10 5 11 7 26 2 steps (‘upsweep’)

Step 3 Vector: 1 3 3 10 5 11 7 36 4 steps (‘upsweep’)

Step 4 Vector: 1 3 3 10 5 11 7 0 Put 0 @ rightmost index

Step 5 Vector: 1 3 3 0 5 11 7 10 4 steps (‘downsweep’)

Step 6 Vector: 1 0 3 3 5 10 7 21 2 steps (‘downsweep’)

Step 7 Vector: 0 1 3 6 10 15 21 28 1 step (‘downsweep’)

C. Faster scan w/ n elements & n processor device?

Hillis-Steele would be faster in this case because it does at most n-1 work in a step and at the end of the algorithm and each step divides the work in half (so it does log(n) steps, meaning the algorithm is O(nlog(n))). In the same case, the Blelloch scan would do about 2x + 1 steps as Hillis-Steele (so 2log(n)). So, it essentially does 2x as many steps, meaning Hillis-Steele would be approximately 2x faster as an algorithm.

D. Faster scan w/ n^3 elements & n processor device?

In this case, the Blelloch scan would be faster than the Hillis-Steele Scan. The Hillis-Steele has to do nlog(n) with n elements, but with n^3 elements, it would have to do approx n^3log(n) work in total because it will do n^3 – 1 work at most (approx n^3 work). The Blelloch scan will at most do n^3 work, but will usually be doing less because the steps in between require far less work than the Hillis-Steele algorithm. This will leave the Blelloch Scan at about n^3/n = O(n^2) and the Hillis-Steele algorithm at about 2n^3log(n)/n = O(2n^2log(n)), making the Blelloch Scan faster in this case (more than 2x faster).