## Q1: Scan Algorithms

1. Show the trace the parallel executions of the Hillis-Steele scan algorithm on vector [1,2,3,4,5,6,7,8]

|  |  |
| --- | --- |
| Step | Vector |
| 0 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | |
| 1 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | **3** | **5** | **7** | **9** | **11** | **13** | **15** | |
| 2 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 3 | **6** | **10** | **14** | **18** | **22** | **26** | |
| 3 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 3 | 6 | 10 | **15** | **21** | **28** | **36** | |

1. Show the trace the parallel executions of the Blelloch scan algorithm on vector [1,2,3,4,5,6,7,8].

|  |  |  |
| --- | --- | --- |
| Step | Vector |  |
| 0 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | | Up |
| 1 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | **3** | 3 | **7** | 5 | **11** | 7 | **15** | | Up |
| 2 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 3 | 3 | **10** | 5 | 11 | 7 | **26** | | Up |
| 3 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 3 | 3 | 10 | 5 | 11 | 7 | **36** | | Up |
| 4 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 3 | 3 | 10 | 5 | 11 | 7 | **0** | | Clear |
| 5 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 3 | 3 | **0** | 5 | 11 | 7 | **10** | | Down |
| 6 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | **0** | 3 | **3** | 5 | **10** | 7 | 21 | | Down |
| 7 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | **0** | **1** | **3** | **6** | **10** | **15** | **21** | **28** | | Down |

1. Which is likely the faster scan to use if you have n elements and n processor device, and why?

The Hillis-Steele scan is likely to be faster because it is more step efficient. Work efficient isn’t as important in this case because there is enough work for the processors.

1. Which is likely the faster scan to use if you have n^3 elements and n processor device, and why?

The Blelloch scan is likely to be faster because it more work efficient. There is more than enough work to saturate the processors, so work efficiency is important.

## Q2: Transpose Coding Exercise

|  |  |  |
| --- | --- | --- |
| K value | Transpose w/ Tiling | Transpose w/Tiling + Shared Mem |
| 1 | 6.604 ms | 11.840 ms |
| 2 | 1.836 ms | 4.214 ms |
| 3 | 0.913 ms | 3.230 ms |
| 4 | 0.564 ms | 2.506 ms |
| 5 | 0.407 ms | 2.361 ms |
| 6 | 0.307 ms | 2.341 ms |
| 7 | 0.243 ms | 2.395 ms |
| 8 | 0.204 ms | 2.199 ms |
| 9 | 0.193 ms | **2.012 ms** |
| 10 | 0.184 ms | 2.135 ms |
| 11 | 0.178 ms | 2.235 ms |
| 12 | 0.175 ms | 1.881 ms |
| 13 | 0.178 ms | 2.205 ms |
| 14 | 0.178 ms | 2.314 ms |
| 15 | 0.18 ms | 2.600 ms |
| 16 | **0.148 ms** | 2.632 ms |
| 17 | 0.196 ms | 3.456 ms |
| 18 | 0.197 ms | 3.135 ms |
| 19 | 0.203 ms | 3.386 ms |
| 20 | 0.212 ms | 3.973 ms |
| 24 | 0.232 ms | 5.332 ms |
| 32\* | 0.285 ms | 8.709 ms |

\*Cannot go past 32 for tile because hardware is limited to 1024 threads per block and 32 \* 32 = 1024.

The tiling without shared memory increases in speed as K approaches 16. At exactly 16, it seems to have an extra speed up as the different from 15 to 16 or 16 to 15 is way more than 2 values in the same area. As K rises past 16, speed decreases. A K value of 16 maximizes the occupancy for tiling without shared memory.

The tiling with shared memory seems to increase in speed as the K value approaches nine. After nine, the tiling with shared memory increases again. It would appear that 9 maximizes occupancy in this case. A K value of 9 maximizes the occupancy for tiling with shared memory.