CSE 491 HW 4

Ryan Casler and Phillip Prescher

Block size and the way the version was implemented had a big impact on how the code performed. First of all v1 was the worst performing at any block size. This was due to having to launch kernels for every loop that was written. This has a lot of overhead and brings down performance. Any kernel using an 8x8 thread block did terrible performing as well. Similar to the reason v1 did poorly, creating that many thread blocks has too high of an overhead to be efficient. The best running code turned out to be v4 with a 16x16 block size. Version 4 was the most optimized with all kernels fused, and utilization of shared memory. Cutting down on the number of global accesses of memory was enough to make it the most efficient implementation. Block size did not make too much of a difference between 16x16 and 32x32.

| Version number | Block size 8 | Block size 16 | Block size 32 |
| --- | --- | --- | --- |
| **1** | 49.7832 | 65.7629 | 68.12 |
| **2** | 78.7722 | 104.276 | 105.864 |
| **3** | 84.8302 | 104.416 | 107.916 |
| **4** | 82.5713 | 108.154 | 107.386 |

Extra Credit

Running times for serial version of the subset algorithm.

The running time increases by 2k as the number of elements in the bin increase.

For the parallel version, we designate a single thread to initiate the first call of the generateSubset() function. This ensures that the actual execution is ran once, and the tasks generated to each recursive call uses all of the available threads afterwards. In order for the tasks to work properly, each individual subset bin needs to be private and individual to the thread. The overall subset, S, can be shared to reduce memory communication overhead.

For the parallel version, the running times increased with the number of threads and bin size. We believe this is because the memory overhead needed to communicate between the tasks and threads. Sharing the S variable increases the running time, but the copying and private variables of subset might be causing the increased running time, since the tasks might be locking due to accessing it.