This homework is super challenging. Implementing single thread and naïve algorithm is not that hard, but for recursive doubling, we cannot adopt array size greater than 6000 since shared memory won’t be able to handle much data, which challenges me since every recursive doubling function I’ve learned online or on textbook used shared memory to perform the task. I tried not to use shared memory but use the iteration stride instead, and it works perfectly for array size = 100 and 1000, but failed for 10000. Then I debugged with Ashley, she told me to have multiple kernels running through the stride, and do the addition in the device, but iteration in host, which runs perfectly. My naïve function cannot handle size=10M, so I only did size up to 1M for 3 algo, which doesn’t tell much differences in time between naïve and recursive doubling. I guess it’s because that the size is not large enough. However, Michael told me that it was because that I didn’t synchronize kernel, thus, I cannot tell the difference between naïve and recursive doubling.