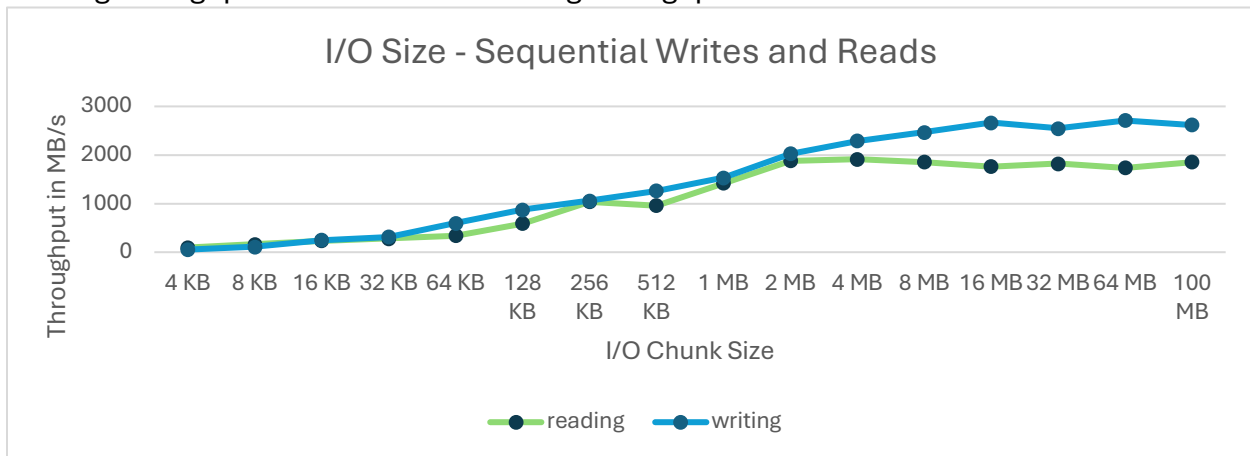


## Analyzing Solid State Drive Performance on Mac

For this lab, I was working on a Mac and didn't have access to testing on Adams, so I did all of the tests on my own device (contains a solid state drive, not a hard disk drive).

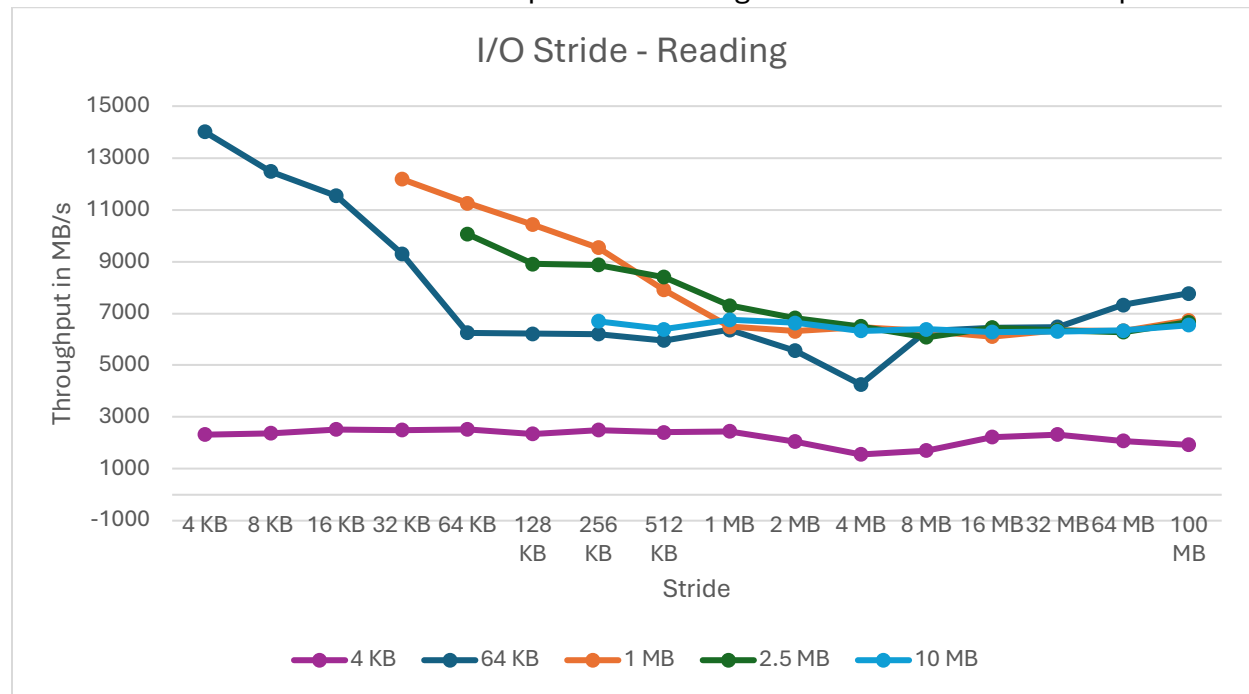
For the I/O Size microbenchmark, I noted an increasing throughput for writing until 4 MB and reading until 1 MB. After these data points, the throughput plateaued. After 4 MB, reading throughput was lower than writing throughput.



I recorded some very high throughput values during the I/O Stride tests which may be due to my using `fcntl(fd, F_NOCACHE, 1)` as a workaround for `O_DIRECT` as I wasn't able to use the `O_DIRECT` flag when opening a file on my Mac. Per the advice when using this workaround, I did `Sudo Purge` between runs but still ended up getting very high throughput. For the I/O Stride writing benchmark, if the 1 MB I/O size is discounted and the data before 1 MB stride is noted, there is a trend of smaller strides resulting in higher throughput. After an 8 MB stride, there is exponential growth in the throughput, likely due to my device. The most consistent stride value seems to be 8 MB, as the throughputs of I/O sizes are all the closest in value at that point.



For the I/O Stride reading benchmark, despite abnormally high throughput for smaller strides, the four largest I/O sizes quickly settled at around 7000 MB/s of throughput, with the 4 KB well below at about 2000 MB/s. It seems that a 1 MB stride was most consistent across I/O sizes with all I/O sizes except for 4 KB being the closest in value at this point.



For the Random I/O microbenchmark, I noted that the reading throughput increased until about 64 KB, plateauing in the range of 5000-6000 MB/s. The writing throughput steadily increased until about 64 MB where it plateaued in the 2500-3000 MB/s range. The best I/O size in this benchmark for reading and writing was in the range of 64 – 100 MB.



If I were designing a system, I would use random instead of sequential reads and writes. For writing, the throughput is comparable, but for reading there is a significant benefit for random over sequential.

