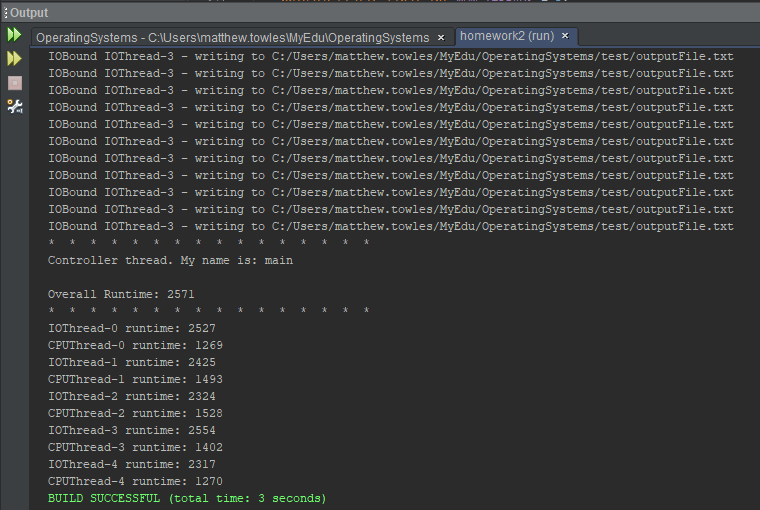
CMSC 412 – Homework 2

**First run:**



Overall Runtime: 2571

IOThread-0 runtime: 2527

CPUThread-0 runtime: 1269

IOThread-1 runtime: 2425

CPUThread-1 runtime: 1493

IOThread-2 runtime: 2324

CPUThread-2 runtime: 1528

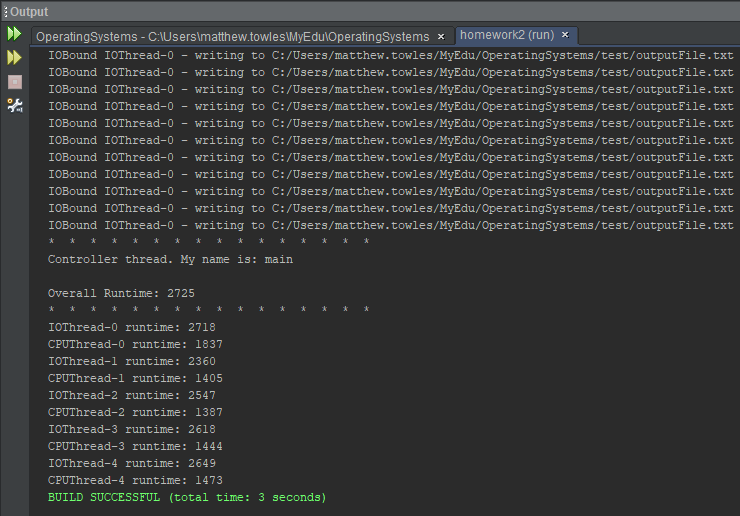
IOThread-3 runtime: 2554

CPUThread-3 runtime: 1402

IOThread-4 runtime: 2317

CPUThread-4 runtime: 1270

**Second run:**



Overall Runtime: 2725

IOThread-0 runtime: 2718

CPUThread-0 runtime: 1837

IOThread-1 runtime: 2360

CPUThread-1 runtime: 1405

IOThread-2 runtime: 2547

CPUThread-2 runtime: 1387

IOThread-3 runtime: 2618

CPUThread-3 runtime: 1444

IOThread-4 runtime: 2649

CPUThread-4 runtime: 1473

**Lessons Learned:**

I have never dealt with threads before so this was a great learning experience. When I first ran my program, all of the results for each thread were runtimes of zero since I was not allowing the child threads to finish prior to the main thread. I referenced the example provided to find a solution by setting the main thread to sleep. I worked my way down to putting it to sleep for 3500 milliseconds, but this was not as responsive as I wanted since either the user will have to wait extra time or the thread will not sleep long enough. Additionally, the runtime given for running all of the threads was inaccurate since it would always reflect the length of time given to the sleep method (plus a few extra milliseconds.) My final solution to this issue was to loop through all of the IO and CPU threads and call join on them since this would still allow for concurrency while also blocking the main thread.