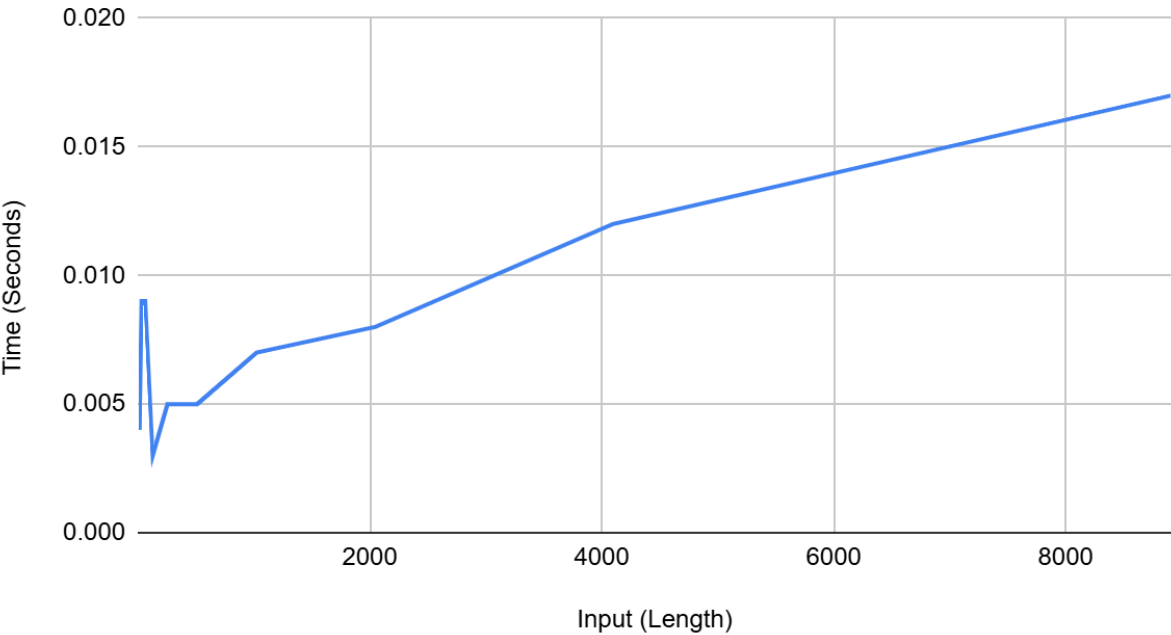


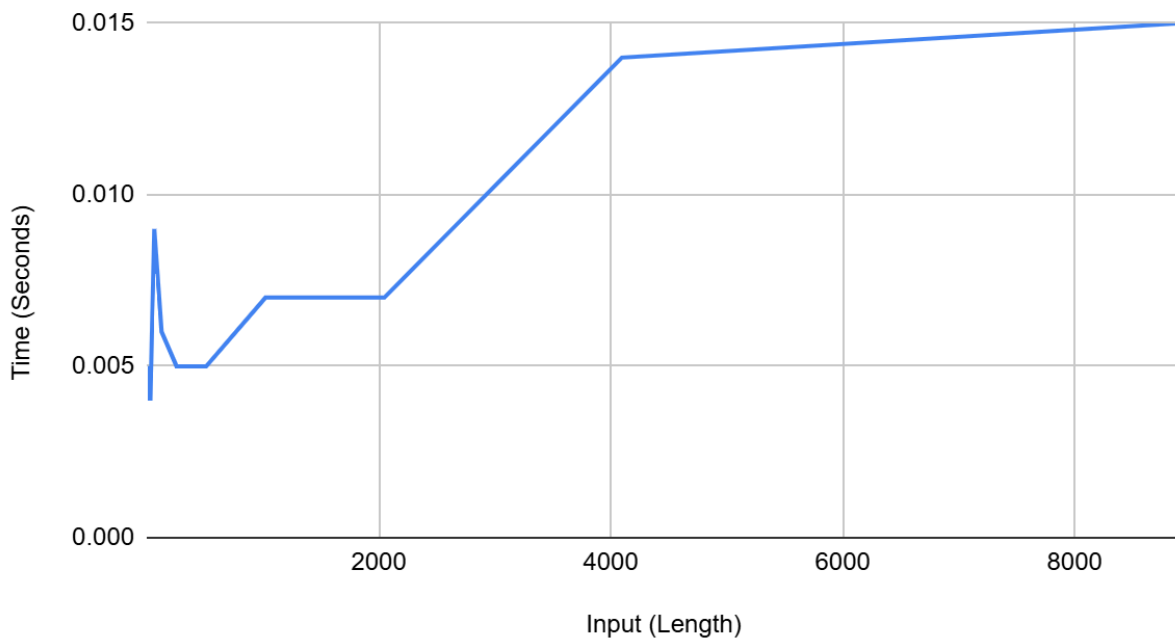
TASK 1

Write				
SysCalls			C Functions	
Input (Length)	Time (Seconds)		Input (Length)	Time (Seconds)
1	0.004		1	0.005
2	0.006		2	0.004
16	0.004		16	0.005
32	0.009		32	0.004
64	0.009		64	0.009
128	0.003		128	0.006
256	0.005		256	0.005
512	0.005		512	0.005
1024	0.007		1024	0.007
2048	0.008		2048	0.007
4096	0.012		4096	0.014
8912	0.017		8912	0.015

Time (Seconds) vs. Input (Length) Writing with SysCall Functcs



## Time (Seconds) vs. Input (Length) Writing with C Functions



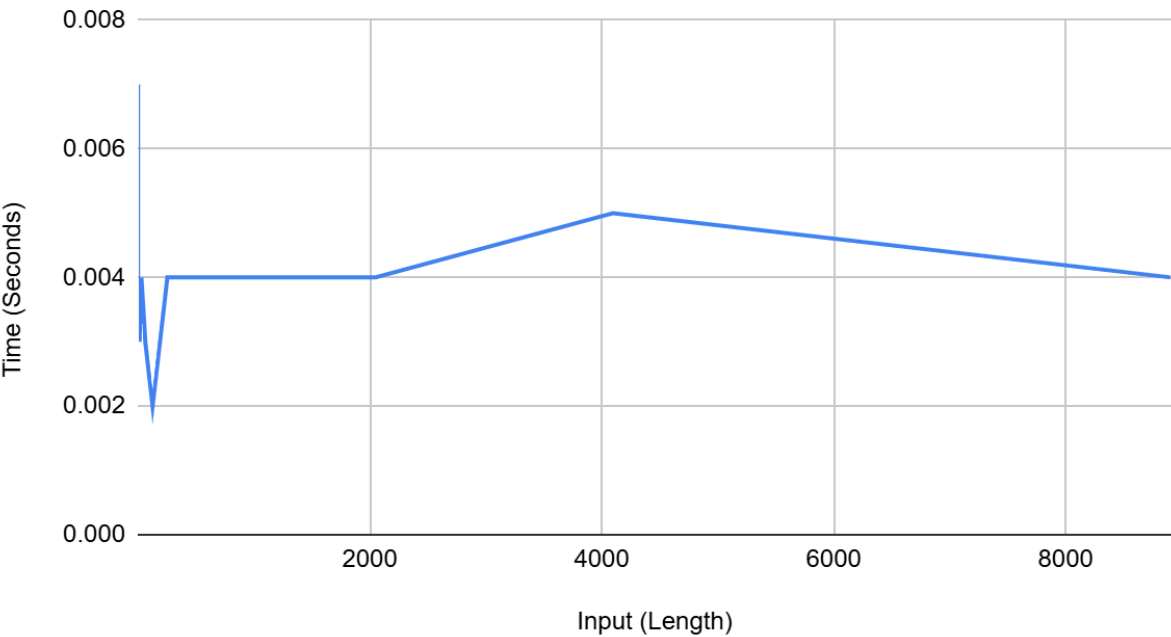
### **Analysis:**

The C functions seem to be taking less time to run than the system calls. The system calls have to take the time to communicate with the Kernel, which is slowing down the program. However, the C functions remain in the user space as they are all buffered by the I/O, so they don't take as long to run.

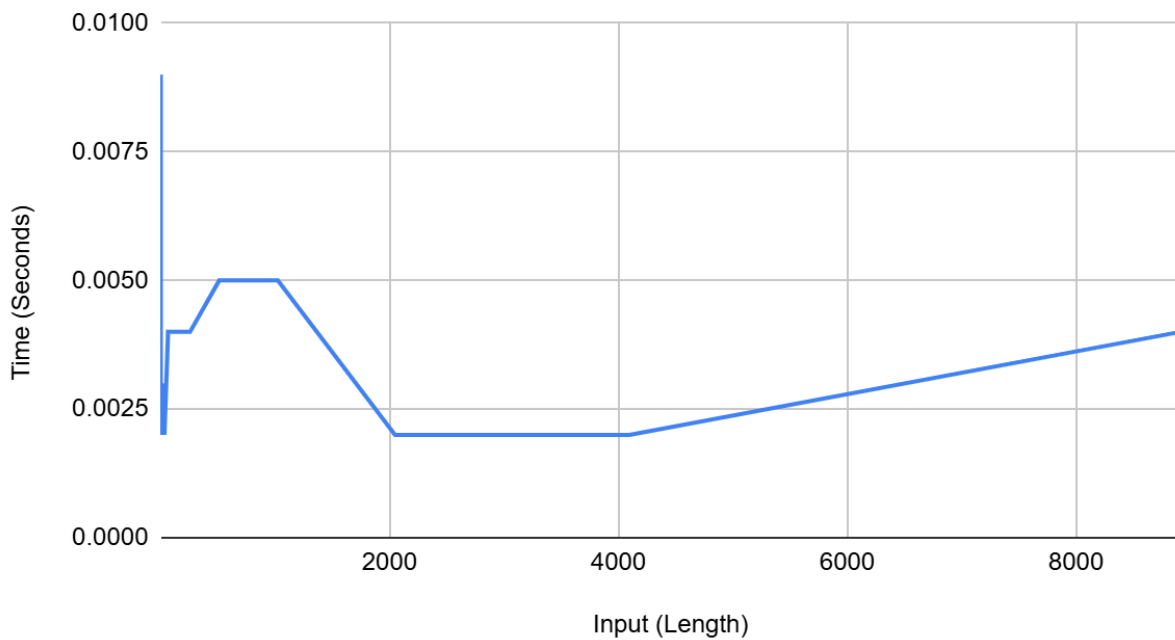
TASK 2

Read				
SysCalls			C Functions	
Input (Length)	Time (Seconds)		Input (Length)	Time (Seconds)
1	0.007		1	0.009
2	0.004		2	0.002
16	0.003		16	0.003
32	0.004		32	0.002
64	0.003		64	0.004
128	0.002		128	0.004
256	0.004		256	0.004
512	0.004		512	0.005
1024	0.004		1024	0.005
2048	0.004		2048	0.002
4096	0.005		4096	0.002
8912	0.004		8912	0.004

Time (Seconds) vs. Input (Length) Reading with SysCall Functcs



## Time (Seconds) vs. Input (Length) Reading with C Functions



### **Analysis:**

Again, the System Calls tended to take longer to run than the C functions, which could be resulting from the system calls needing to interact with the Kernel. The C functions are faster because they are buffered and do not need to interact with the Kernel.