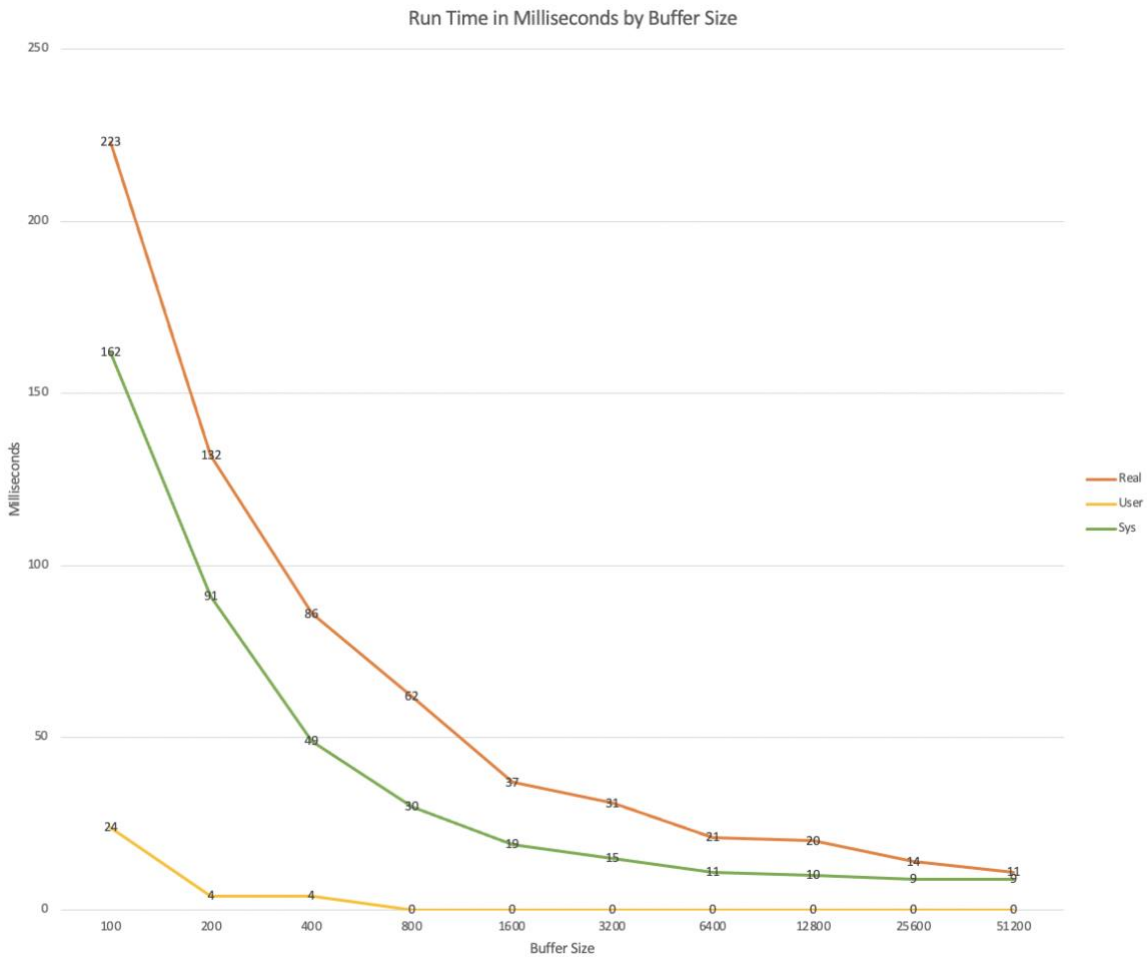


Ilana Sivan
205634272
Exercise 1
Operating Systems

3.



For reference the table:

Buf size	Real	User	Sys
100	223	24	162
200	132	4	91
400	86	4	49
800	62	0	30
1600	37	0	19
3200	31	0	15
6400	21	0	11
12800	20	0	10
25600	14	0	9
51200	11	0	9

4.

- What is the meaning of real, user and sys?

Real is the time on the clock which represents the time from start to finish of the call and includes time used by other processes and the time the process spends waiting or completing. User is the CPU time used in execution and does not include the time of other processes or how long the process is waiting. Sys is the amount of CPU time spent in the kernel within our process, and executes CPU time spent in system calls *within the kernel*, as opposed to library code, which is still running in user-space, like 'user', this is only CPU time used by the process,

- Why don't sys and user sum up to real?

Real time includes the time of other processes and blocked time, and it includes I/O and can run longer if the CPU lacks.

- Why isn't real a straight line parallel to the X axis?

The smaller the buffer is the more system calls will be required when we read from the file (and write to the file) and system calls take more time which makes the program run slower.

5. Hypothetically, if we change the **append_file** in part 1 to print out to the screen a message each time we read **buffer_size** bytes, will the running time change significantly?

- **IMPORTANT:** Do not make this change. Submission with this change will result in 0 points for this question!

The running time will change significantly and we will run slower since we will print a lot of messages and printing is expensive (time-wise) as an operation which requires the use of a system call.