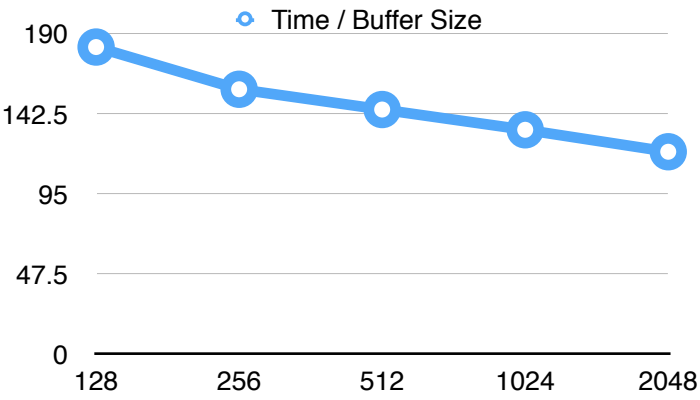
Part 1
View attached Java files.

Part 2
2) It's not a strait line as for each buffer size we are iterating different amount of times.



For the smallest buffer size, we iterate the most. That leads to bigger time usage as we us more system calls, while at the biggest buffer size, we iterate a significant less amount of times leading to less system calls and faster running times.

3) If we will add a printing command after each time the buffer is reloaded the running time will increase significantly as this is another operation in our flow and we are using another system call, this will occur a greater amount of times as the buffer size is reduced.
This will effect smaller on bigger buffer sizes as we iterate less over the file.

Part 3

1) **False**, To display user with any kind of output we need to use a system call.
2) **False**, an interrupt is created and not a system call.
3) **False**, Interrupts are 'signals' to the processor to stop executing current instructions and jump to some kernel (OS) code that handles interrupts.
4) **False**, They are run via user mode, but are using some system resources and therefore using system calls to operate correctly.
5) **False**, It does not matter when the software were installed, what matters is it's purpose. all user installed software are run in user mode (with minor exceptions) to protect the system.
6) **False**, The CPU is controlling the interrupts, not the OS.
7) **False**, Running a program in a VM only increases the way for each command of our program to relevant system calls as the VM only emulates system calls but actually calls them itself.
8) **False**, The program will run in user mode and use system calls to interact safely with the relevant hardware component.
9) **False**, Less system calls is better as they are expensive for us (running time wise).
10) **False**, External devices are using interrupts to invoke the OS and not system calls.