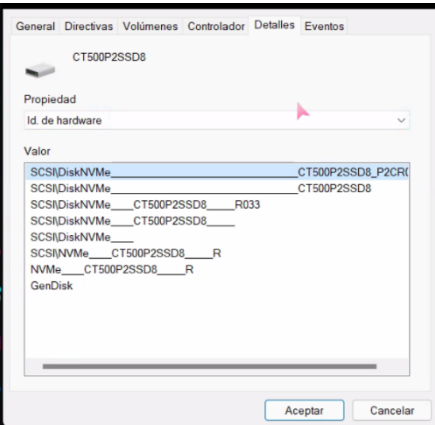


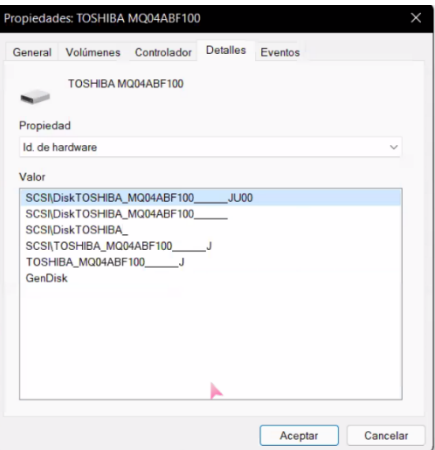
Homework 4

Specifies of the drives:

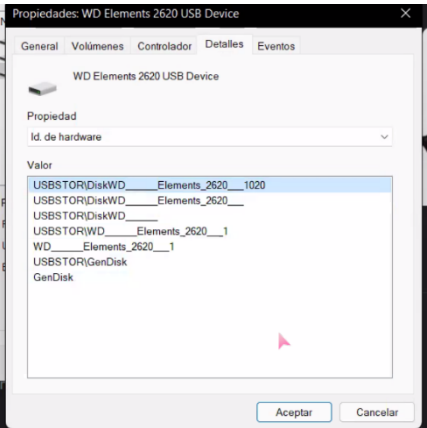
- SSD:



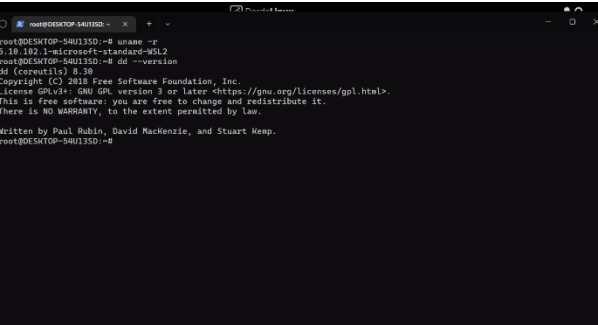
- HDD:



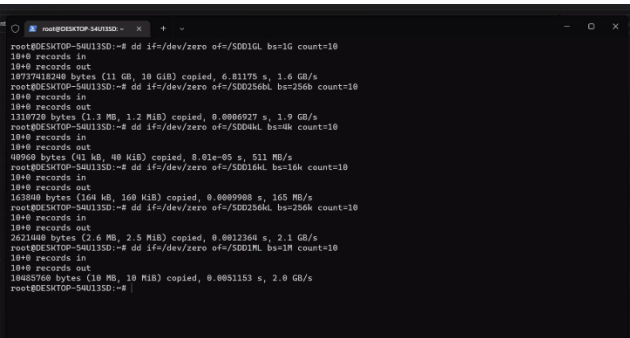
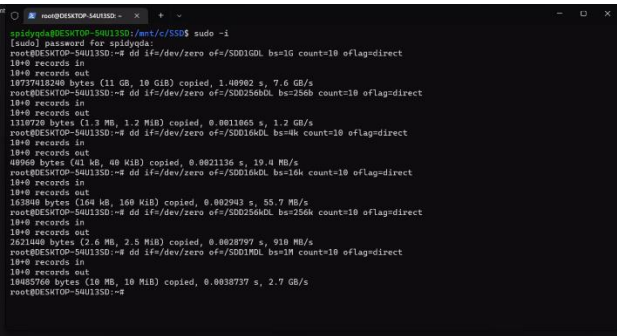
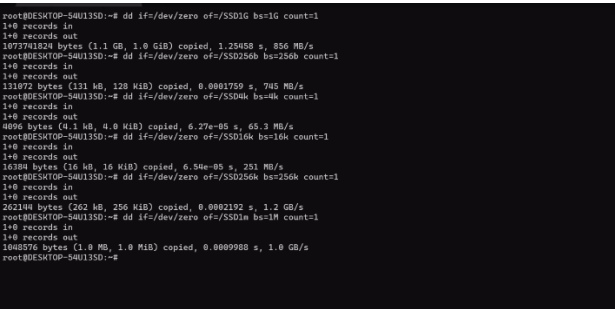
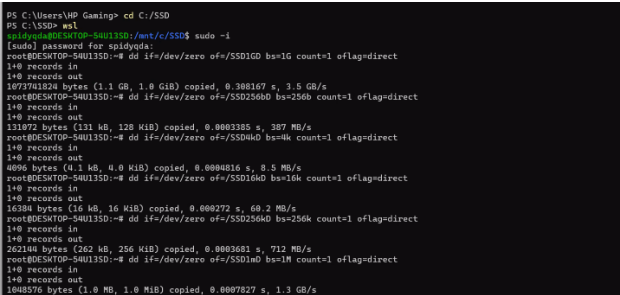
- EX(HDD):



Specifies of Linux and dd:

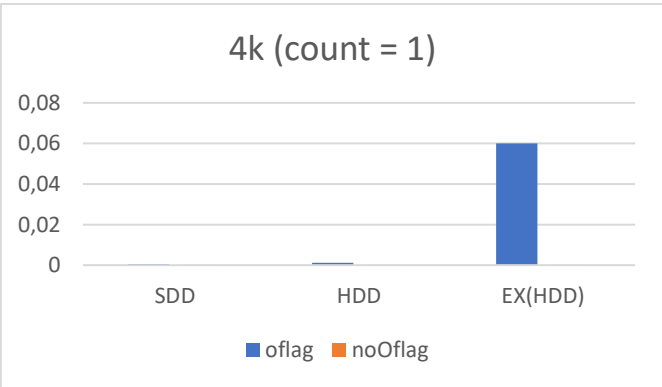
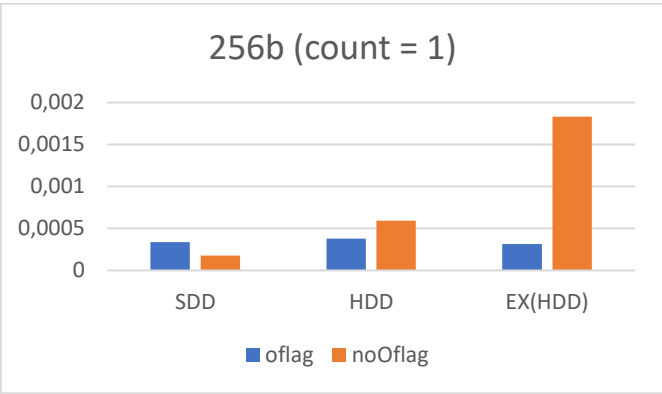
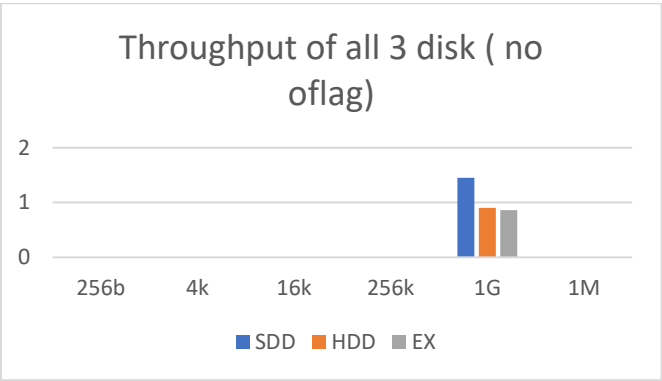
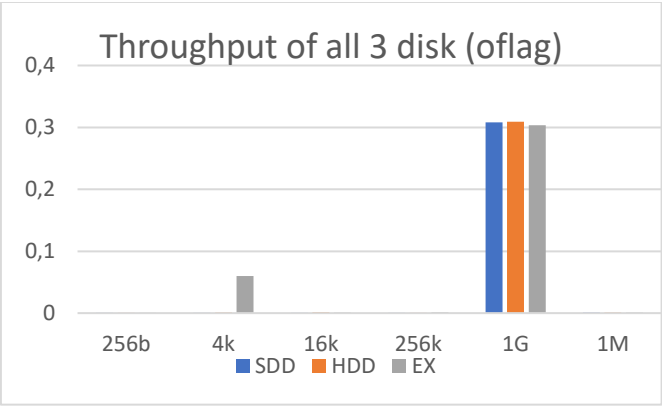


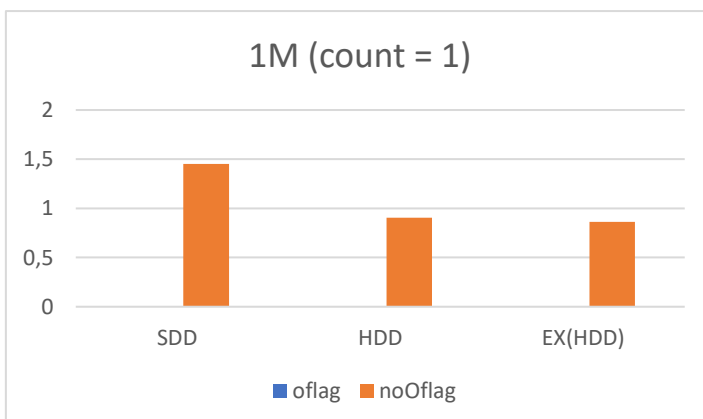
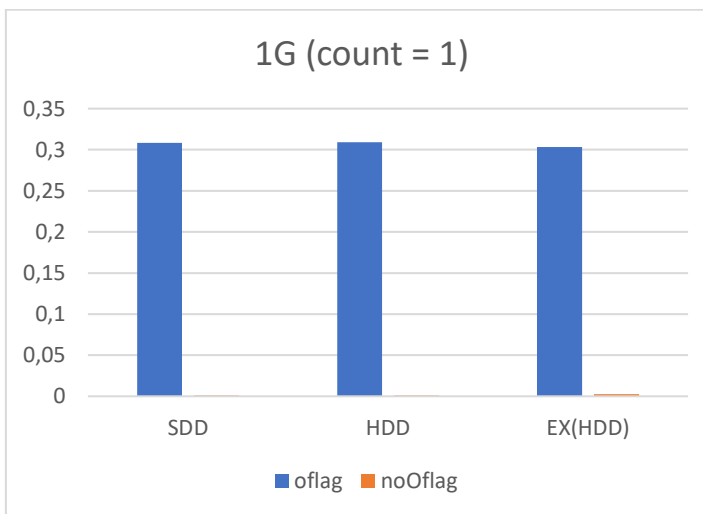
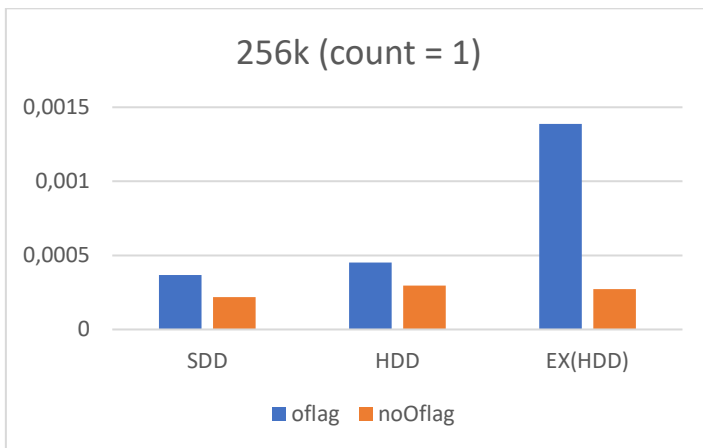
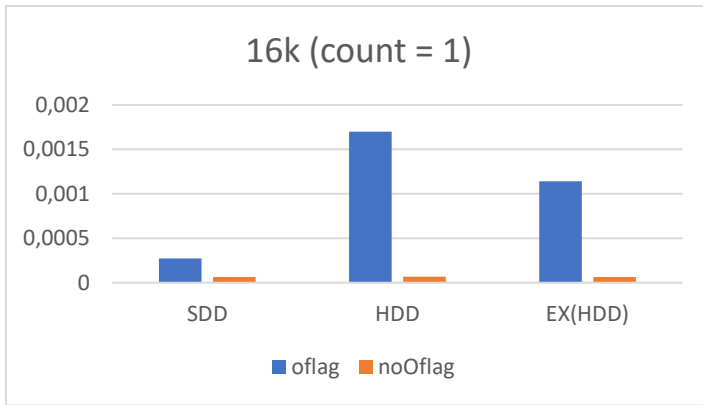
Commands used:

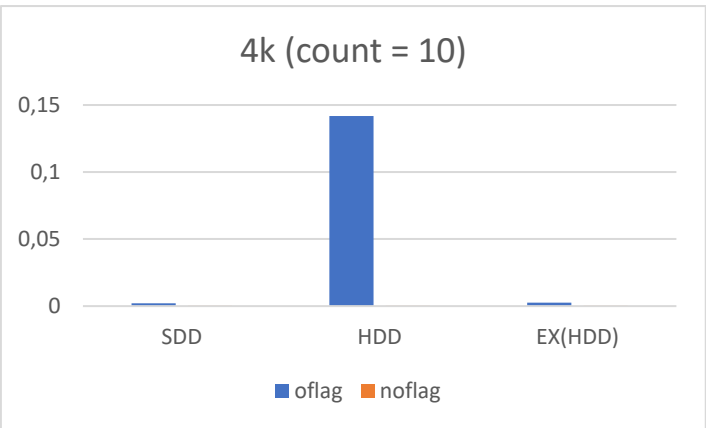
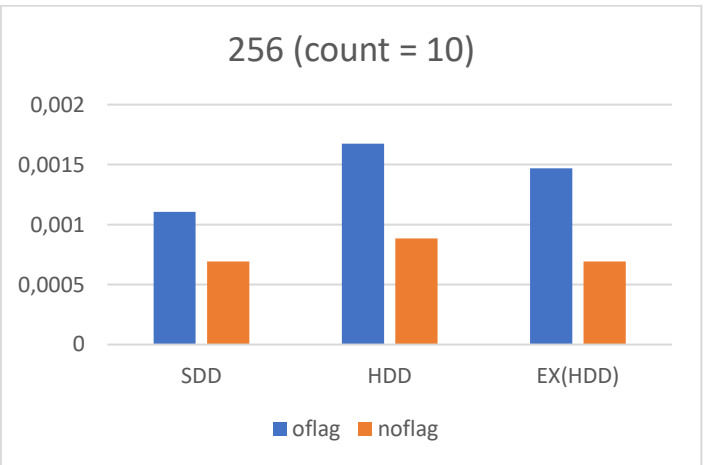
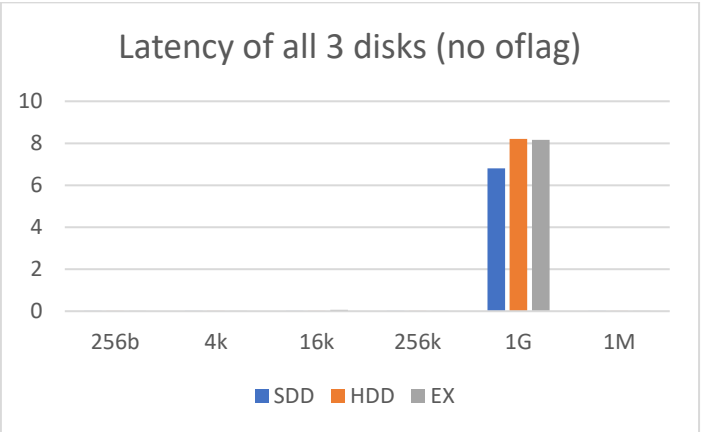
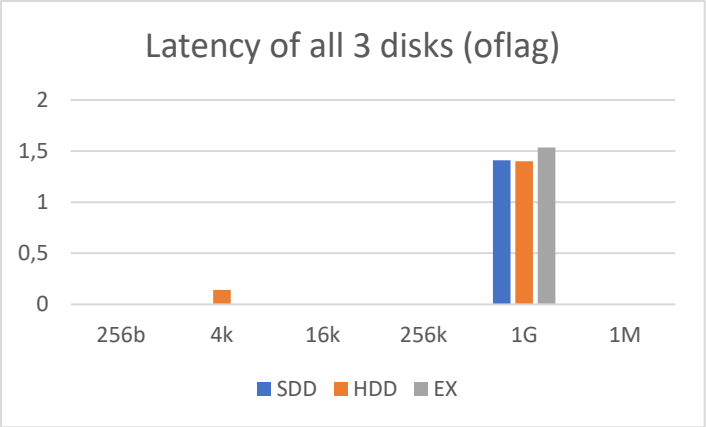


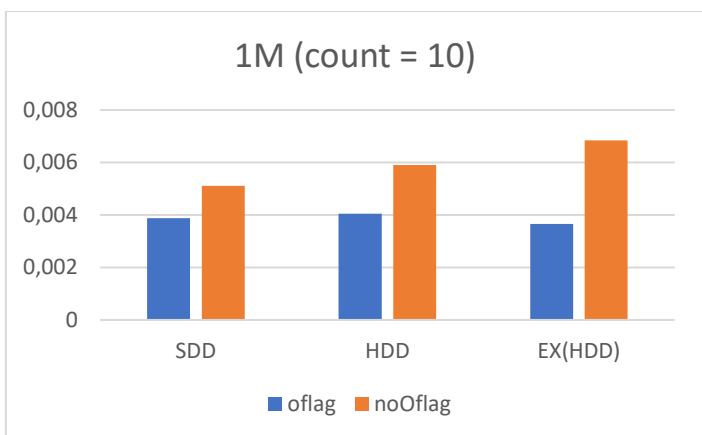
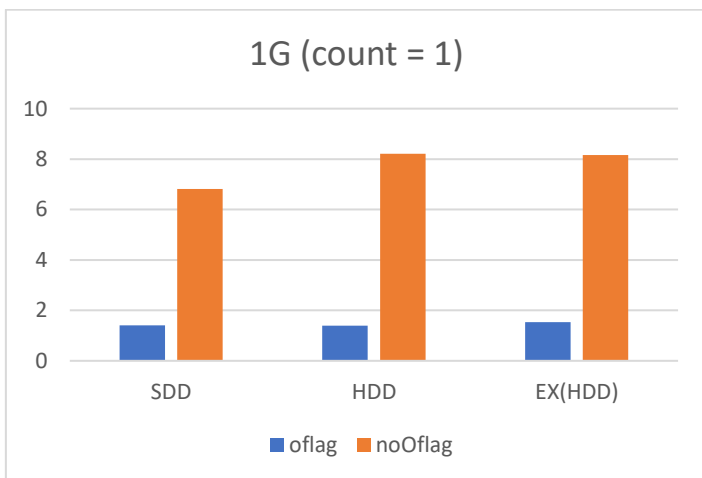
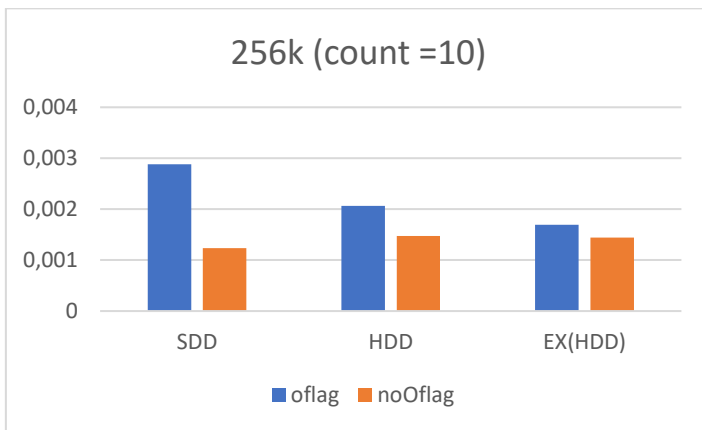
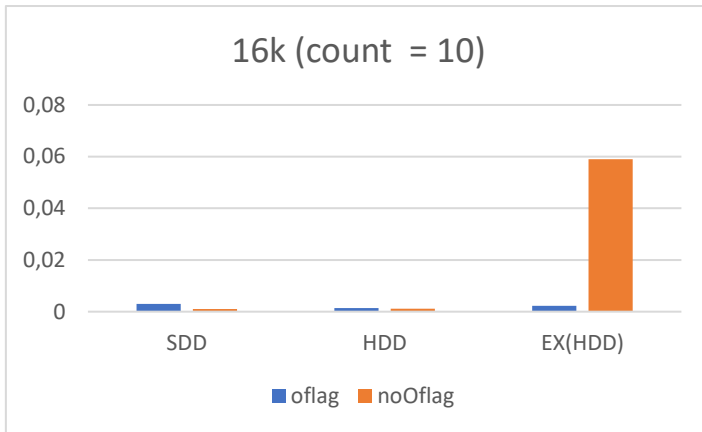
Note: Se utilizo los comandos presentados para el resto del trabajo, pero modificando tipo de disco.

Graphics:









Notes:

- Los gráficos con sus correspondientes tablas se encuentran en el Excel adjuntado, y se pueden observar mejor.
- Si bien se realizó un gráfico comparando los 3 discos, tanto en throughput como en latency, los resultados no se pudieron analizar correctamente. Esto se debió a que había una gran diferencia en algunos tiempos, por ende, la escala del gráfico no permitió visualizar los resultados adecuadamente.

Questions:

What do you observe?

A partir de los gráficos se puede observar que, dependiendo del disco, de los valores de bs y de si tiene el oflag o no, las diferencias en el performance varían bastante, y no hay una tendencia muy clara.

En términos del throughput, en teoría los discos SSD deberían ser mas rápidos que los HDD, pero al utilizar 256b y 1G como valores del bs y usando el oflag, se muestra lo contrario. Además, en teoría al usar el oflag el proceso debería ser más rápido, pero para los mismos valores de bs, previamente mencionados, ocurre lo contrario.

En términos de la latency, se puede observar un comportamiento similar, en términos de peor performance por parte de los SSD, al utilizar 16k y 256k como valores del bs y usando el oflag. Igualmente, para los valores de 1M y 1G el performance es mejor sin el oflag, contradiciendo de la teoría.

What are the main differences between the three drives?

- Cuando se trabaja con valores de bs más grandes el SSD tiene mejor performance que los HDD
- El espacio de memoria que tenía cada disco, el SSD tenía 500GB, el HDD tenía 1T, y el HDD externo tenía 2T.
- El SSD accede a los datos de forma más rápida sin importar en donde se encuentre. En cambio, el HDD accede más rápido a los datos que se encuentran más cerca del exterior, en comparación a los datos que se encuentran en el centro.

What are the differences in performance when we change the value of bs?

Se puede observar que en general el performance no cambia mucho al aumentar el valor de bs, excepto cuando este es 1G. En este valor de bs se puede observar que el performance se relentiza en grandes cantidades, a comparación de los otros valores. Cabe mencionar que esto ocurre independientemente de si está o no con oflag. También cuando el valor de bs es 4k, se puede observar que en el HDD en el throughput y en el

EX (también es un HDD) hay un performance más lento, pero sigue siendo más rápido que en 1G. Hay que destacar que esto solo ocurre cuando esta con oflag.

Why we see differences in performance for different values of bs?

Como sabemos, el dd permite realizar distintas funciones en el sistema, en este caso, sabemos que la función “bs” se encarga de especificar el numero de octetos que deberán ser leídos y escritos durante el proceso. Esto significa que, a más cantidad de bits pidamos, más información guardara en la copia. Y por tanto se demorará más tiempo al cumplir el proceso Por este motivo podemos ver una gran diferencia entre los valores definidos en la función bs.

How do we explain the difference in performance of these two options?

Como sabemos, oflag se encargar de acceder a los archivos de salida usando Flags especificas por cada argumento FRAG. En este caso podemos observar que mientras mas valores bs tenemos en nuestro proceso, este es más rápido si se usa la función “oflag=direct” a comparación de no usarla, el motivo por el cual se da este tipo de comportamiento es porque el oflag le está dando al kernel la capacidad de escribir datos de forma inmediata en lugar de almacenarlos en el buffer a esperar a que los datos entren y salgan del buffer.