대용량 파일 복사

Mmio를 사용해서 파일을 만들면 좀더 빠른 시간 안에 파일을 만들 수 있다.

file\_copy\_using\_read\_write와 file\_copy\_using\_memory\_map 두 가지 방법으로 파일 복사를 했을 때 처음에는 file\_copy\_using\_read\_write 가 더 빨리 복사되는 것을 볼 수 있다. 미리 버퍼 크기를 정해두고 한번에 복사하기 때문에 작은 용량의 파일 복사에 유리하다. 버퍼크기를 1로 지정해주면 file\_copy\_using\_memory\_map 가 더 빠른 것을 확인 할 수 있다. 결론적으로는 memory mapped IO 를 사용하는 것이 더 빠르다.

(실제 수업시간에는 버퍼크기 때문에 read write 를 쓰는 것이 빠르게 보였을 지라도)

그러나 32비트 운영체제에서 용량이 큰 파일은 복사가 불가능 한 문제가 있었다. 32비트 운영체제에서는 400 메가 짜리 파일까지는 복사가 가능했고 64비트 운영체제에서 700 메가 정도의 파일까지 복사가 가능했다.

따라서 32비트 운영체제에서도 복사가 가능하게 하려면 400메가 단위로 잘라서 복사하는 방법이 있다. 오프셋으로 복사한 부분까지 파악하고 400 메가 단위로 메모리에 올리고 내리고 올리고 내리는 방식으로 5기가 이상 용량의 파일을 복사 할 수 있다. 오프셋을 사용해서 만드려고 했으나 system 함수를 사용하여 copy를 사용해도 파일이 복사되는 것을 확인 하였다. 멘토님의 의도는 오프셋을 활용한 memory mapped IO 복사 였지만 copy 명령어를 활용하여 대용량 파일 복사가 가능하다.