

운영체제 Assignment 5

이름 : 이준휘

학번 : 2018202046

교수 : 최상호 교수님

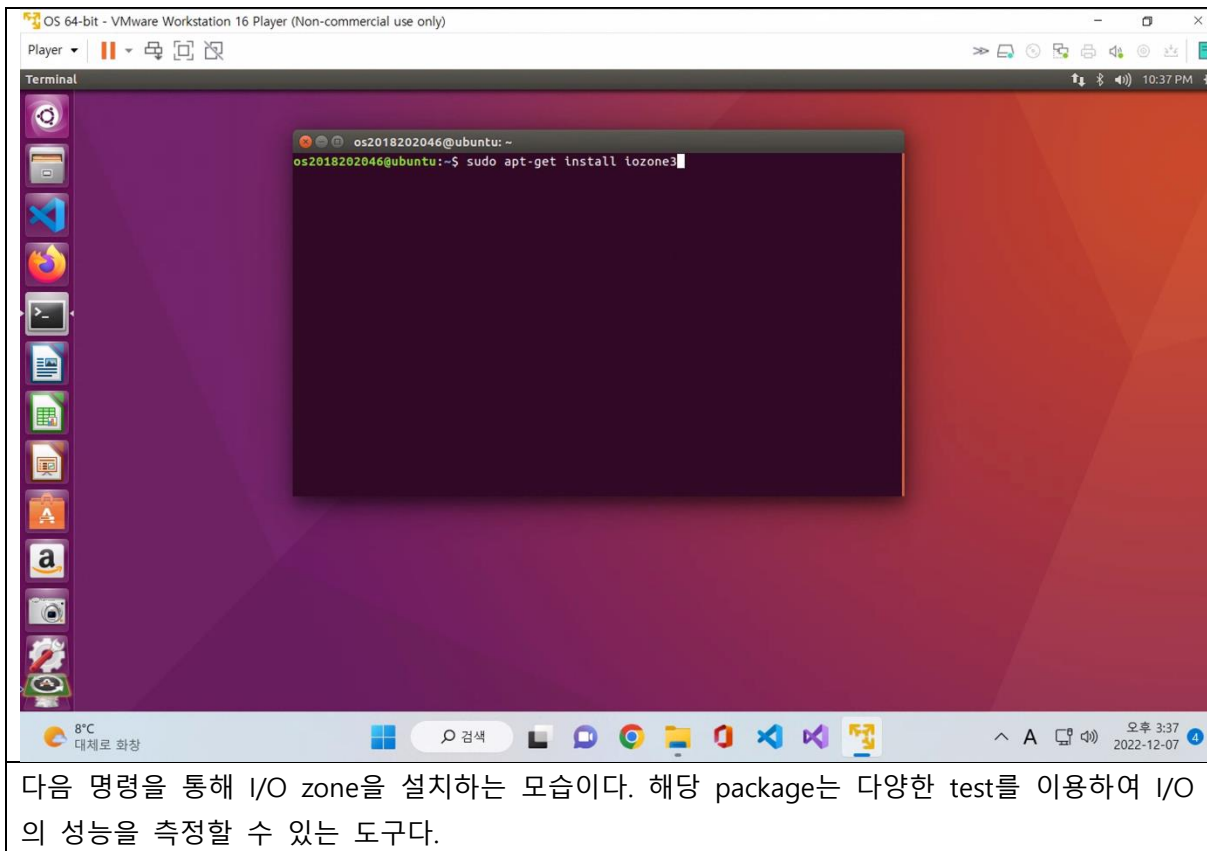
강의 시간 : 금 1, 2

1. Introduction

해당 과제는 I/O Zone을 이용하여 Linux I/O scheduler의 성능을 측정한다. 측정하는 scheduler는 noop, deadline, CFQ로 3가지이며, scheduler 별 성능 결과 표 및 그래프를 작성한다. iotop의 테스트 연산은 4가지로 write/re-write, read/re_read, random-read/write, 이외의 다른 연산 하나를 수행하여 검사한다. record size를 8KB ~ 16MB까지 변화시켜가며, thread는 1개, File size는 1GB, Buffer cache를 거치지 않고 연산을 수행하여 결과를 엑셀을 통해 확인한다.

2. Conclusion & Analysis

A. noop



```
OS 64-bit - VMware Workstation 16 Player (Non-commercial use only)
Player
Terminal
os_5_2018202046_B
os2018202046@ubuntu: ~/os_5_2018202046_B
sched = noop
record_size = 8k
file_size = 1g
default:
rm -rf ~/iozone_test
sync
echo 3 | sudo tee /proc/sys/vm/drop_caches
iozone -R -i 0 -i 1 -i 2 -i 8 -I -r $(record_size) -s $(file_size) -t 1 -F ~/iozone_test -b $(sched) $(record_size)_1.
rm -rf ~/iozone_test
sync
echo 3 | sudo tee /proc/sys/vm/drop_caches
iozone -R -i 0 -i 1 -i 2 -i 8 -I -r $(record_size) -s $(file_size) -t 1 -F ~/iozone_test -b $(sched) $(record_size)_2.
rm -rf ~/iozone_test
sync
echo 3 | sudo tee /proc/sys/vm/drop_caches
iozone -R -i 0 -i 1 -i 2 -i 8 -I -r $(record_size) -s $(file_size) -t 1 -F ~/iozone_test -b $(sched) $(record_size)_3.
rm -rf ~/iozone_test
sync
echo 3 | sudo tee /proc/sys/vm/drop_caches
iozone -R -i 0 -i 1 -i 2 -i 8 -I -r $(record_size) -s $(file_size) -t 1 -F ~/iozone_test -b $(sched) $(record_size)_4.
rm -rf ~/iozone_test
sync
echo 3 | sudo tee /proc/sys/vm/drop_caches
iozone -R -i 0 -i 1 -i 2 -i 8 -I -r $(record_size) -s $(file_size) -t 1 -F ~/iozone_test -b $(sched) $(record_size)_5.
2,16 All
```

해당 파일은 Makefile로 각 record size에 해당하는 iozone을 5번 test하기 위해 만들어졌다. sched 변수에 현재 수행하는 스케줄러명을 입력하며 record_size에 검사할 파일명을 입력한다. 또한 file_size에는 파일 크기를 입력한다.

명령으로 다음을 5회 반복한다.

우선 `rm -rf ~/iozone_test`를 통해 process의 파일을 지운다. 그 후 `sync`를 통해 file system buffer를 flush한다. 그 후에는 `echo 3 | ~` 명령을 통해 pagecache, dentries, and inode를 수행한다. 그 후 `iozone` 명령을 수행한다. `-R` 옵션을 통해 Excel report를 생성하며 `-i 0 1 2 8` 옵션을 통해 실행할 test를 지정한다. 0은 write/re-write, 1은 read/re-read, 2은 random-read/write, 마지막으로 8은 필자가 사용한 **random-mix test**로 전체적인 성능을 평가하기 위하여 해당 test를 사용하였다. `-I` 옵션을 사용하여 cache를 거치지 않으며, `-r` 옵션을 통해 변수로 넣은 record size로 크기를 지정하고 `-s` 옵션은 파일 크기로 1GB로 설정한다. `-t 1` 옵션을 사용하여 thread를 1개 사용하며, `-F` 옵션을 통해 process가 생성할 파일의 이름을 지정한다. 마지막으로 `-b` 옵션을 사용하여 binary file의 path를 지정한다. 파일명은 각각 1~5까지의 숫자로 부여한다.

OS 64-bit - VMware Workstation 16 Player (Non-commercial use only)

Player ▾ | 1:45 AM

LibreOffice Calc

noop_8k_1.xls - LibreOffice Calc

Formula bar: `iozone -R -i 0 -i 1 -i 2 -i 8 -i 8k -s 1g -t 1 -F /home/os2018202046/iozone_test -b noop_8k_1.xls`

	A	B	C	D	E	F	G	H	I	J	L	M	N
1	iozone -R -i 0 -i 1 -i 2 -i 8 -i 8k -s 1g -t 1 -F /home/os2018202046/iozone_test -b noop_8k_1.xls												
2	Throughput report Y-axis is type of test X-axis is number of processes												
3	Record size = 8 kBytes												
4	Output is in kBytes/sec												
5	Initial write		99544.875										
6	Rewrite		119613.5313										
7	Read		143802.0469										
8	Re-read		114507.0781										
9	Random read		109203.0234										
10	Mixed workload		131181.4844										
11	Random write		92505.80469										
12													
13													
14													
15													
16													
17													
18													
19													
20													

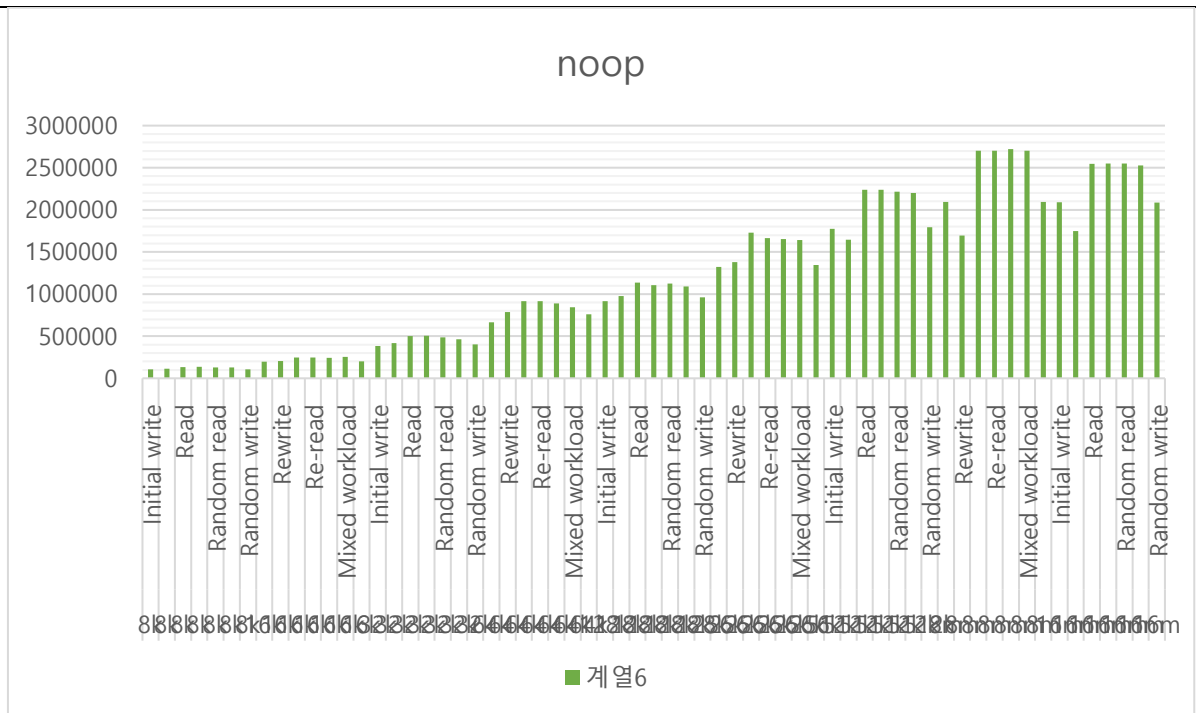
Sheet 1 of 1 | PageStyle:noop_8k_1 | Sum=0 | 100%

Taskbar: 5°C 맑음 | 검색 | 2022-12-07 오후 6:45

다음은 파일의 내용을 가져온 것이다. 각 파일에서 Initial write/Rewrite는 0번 test, Read/Re-read는 1번 test, Random read/write는 2번 test, 마지막으로 Mixed workload는 8번 test의 결과를 의미한다.

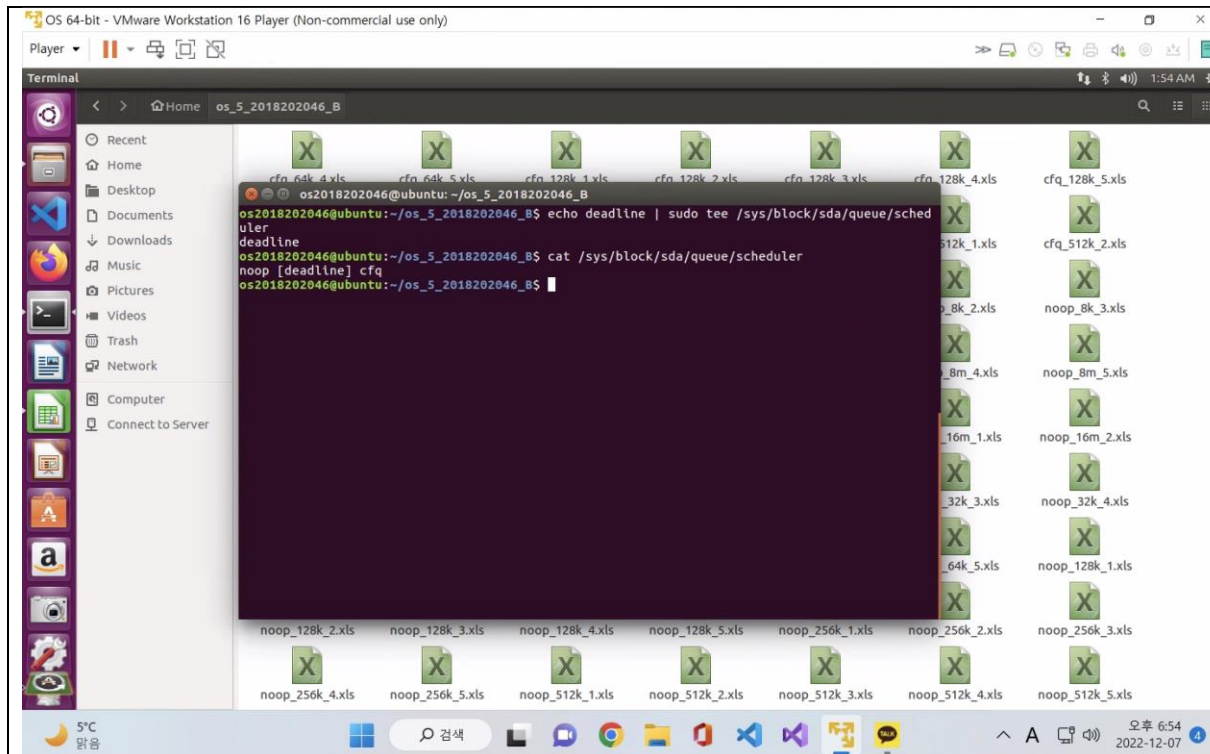
	record size		1	2	3	4	5	average(kbyte)
noop	8k	Initial write	115976.3438	120815.6	98052.03	97497.63	97928.73	106054.0668
	8k	Rewrite	127579.7656	101981.5	102479.6	102031	137757.3	114365.8331
	8k	Read	155257.0938	121669.3	121019	121165	151965.8	134215.2388
	8k	Re-read	125266.9688	150825.5	121967	123070.7	153030	134832.0338
	8k	Random	147217.9063	135126.4	118106.1	118386.9	117413.9	127250.2413
	8k	Mixed work	117791.9844	147883	116306.8	115737.2	143984	128340.5969
	8k	Random v	99181.39844	97477.15	100280.9	97934.47	128748.7	104724.5237
	16k	Initial write	186999	188389.141	235246	186300.6	187140.2	196814.9882
	16k	Rewrite	198864.5	199302.719	222926.6	198630.2	200547.3	204054.2638
	16k	Read	238092.7	238068.656	278017.3	237157.8	238121	245891.4912
	16k	Re-read	238443.8	290139.469	229387.4	237001.8	238850.4	246764.5738
	16k	Random	250697.8	275071.469	227519.6	226959.6	230937.7	242237.2338
	16k	Mixed work	227587.9	277272.75	265080.2	227135.4	282925.4	256000.33
	16k	Random v	223170.4	191715.922	204767.8	191037.1	190854.7	200309.1844
	32k	Initial write	348257.5	354565.6	429172.3	425614.1	358733.4	383268.58
	32k	Rewrite	370412.1	374985.6	483868.9	491688.6	374047.9	419000.62
	32k	Read	447642.3	448502.5	536547.3	535724.6	547985.3	503280.4
	32k	Re-read	538886.4	452630.5	539023	539164.9	448577.3	503656.42
	32k	Random	517715.9	443501.5	520341.8	511457.3	435425.3	485688.36
	32k	Mixed work	436016.4	428220.6	511351.1	506848.8	433411.8	463169.74
	32k	Random v	458394.5	360391.3	364771.8	461619.1	362977.8	401630.9
	64k	Initial write	641275.2	714232.8	648043.1	666077.1	652320.2	664389.68
	64k	Rewrite	691151.8	837851.9	872106.3	691122.6	842795.6	787005.64
	64k	Read	828579.9	960887.3	829738.2	985702	966970.4	914375.56
	64k	Re-read	977000	959389.3	827387.3	974927.8	842055.8	916152.04
	64k	Random	943241.5	935541.8	802821.8	945627.3	814294.6	888305.4
	64k	Mixed work	804161.6	928618.1	795406.8	932361	761881.4	844485.78
	64k	Random v	814283.7	801559.9	850222.6	669555.3	656607.9	758445.88
	128k	Initial write	946109.4375	931001.7	876494.6	861459.9	960957.8	915204.6875
	128k	Rewrite	1053456.625	961932.6	903954.3	1055352	902555.1	975450.125
	128k	Read	1190488.25	1103788	1094922	1192165	1098151	1135902.85
	128k	Re-read	1141926	1104775	1085835	1093934	1102105	1105715
	128k	Random	1145117.375	1059464	1203713	1146035	1061461	1123158.075
	128k	Mixed work	1118557.375	1057584	1077905	1142451	1060231	1091345.675
	128k	Random v	1016126.063	1040921	882760.8	987963.4	875437.1	960641.6725
	256k	Initial write	1385917	1293199	1317451	1292759	1315583	1320981.8
	256k	Rewrite	1454138	1488332	1346097	1301760	1298366	1377738.6
	256k	Read	1733630	1788656	1653165	1800650	1673766	1729973.4
	256k	Re-read	1723171	1668421	1620193	1640456	1670928	1664633.8
	256k	Random	1688925	1618815	1596332	1744086	1612782	1652188
	256k	Mixed work	1684811	1605463	1597251	1712624	1607981	1641500
	256k	Random v	1408016	1249190	1305944	1399295	1362762	1345041.4
	512k	Initial write	1792845	1796319	1730032	1741573	1816864	1775526.6
	512k	Rewrite	1800069	1827543	1602903	1210849	1779009	1644074.6
	512k	Read	2201014	2240782	2316348	2150723	2274347	2236642.8
	512k	Re-read	2202044	2312690	2317827	2151715	2217374	2240330
	512k	Random	2185599	2189405	2242090	2292747	2166117	2215191.6
	512k	Mixed work	2162231	2167769	2246201	2276910	2149558	2200533.8
	512k	Random v	1788704	1793695	1809874	1818648	1752431	1792670.4
	8m	Initial write	2143495	2143223	2141674	1988670	2054196	2094251.6
	8m	Rewrite	2099571	1963949	1824039	508010.3	2074269	1693967.66
	8m	Read	2666679	2707750	2676338	2760909	2705117	2703358.6
	8m	Re-read	2720931	2714268	2706239	2731364	2630773	2700715
	8m	Random	2731208	2723200	2734383	2711194	2699086	2719814.2
	8m	Mixed work	2719654	2692196	2704844	2731036	2669844	2703514.8
	8m	Random v	2113012	2071805	2104780	2131698	2052961	2094851.2
	16m	Initial write	2078141	2067480	2107365	2113771	2092478	2091847
	16m	Rewrite	1669003	1597231	1804058	1570954	2102197	1748688.6
	16m	Read	2537236	2554263	2553281	2544211	2544891	2546776.4
	16m	Re-read	2547989	2594317	2528779	2521125	2563242	2551090.4
	16m	Random	2582209	2547302	2534151	2582617	2512023	2551660.4
	16m	Mixed work	2538630	2543569	2522903	2502474	2522884	2526092
	16m	Random v	2084810	2104920	2082268	2098562	2067761	2087664.2

위의 Excel 파일을 하나로 정리하여 평균을 보았을 때 다음과 같은 결과를 보였다.

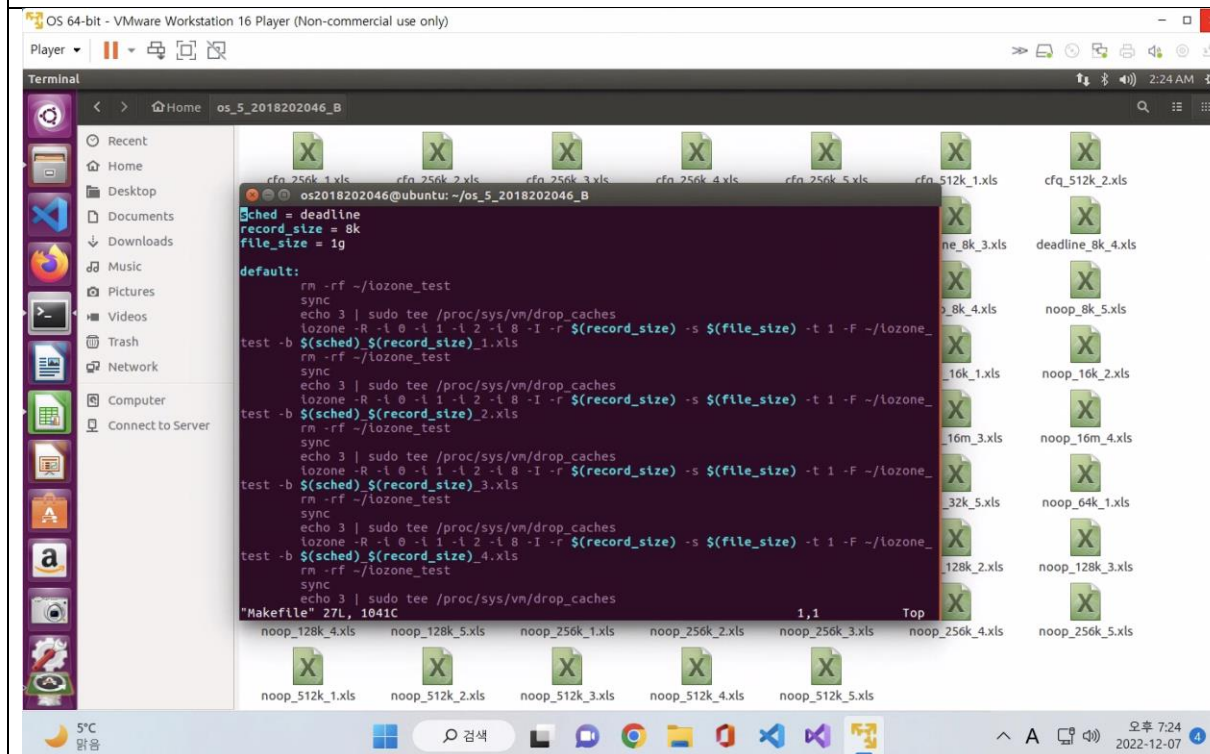


다음 그래프는 위의 표를 보기 쉽게 정리한 것이다. 일반적으로 write 동작이 read 동작에 비해 Kbytes/sec가 더 낮게 나오는 것을 확인할 수 있으며 이 차이는 record size가 커질수록 더 심해진다. 또한 record size가 커질수록 성능이 더 좋아지다 8MB를 기점으로 이보다 더 클 경우 읽기 성능의 경우 소폭 하락하는 경향을 확인할 수 있다. 또한 noop 스케줄러의 장점이기도 한 random read/write의 성능이 일반적인 read/write와 유사하게 나오거나 더 좋게 나오는 것을 확인할 수 있다.

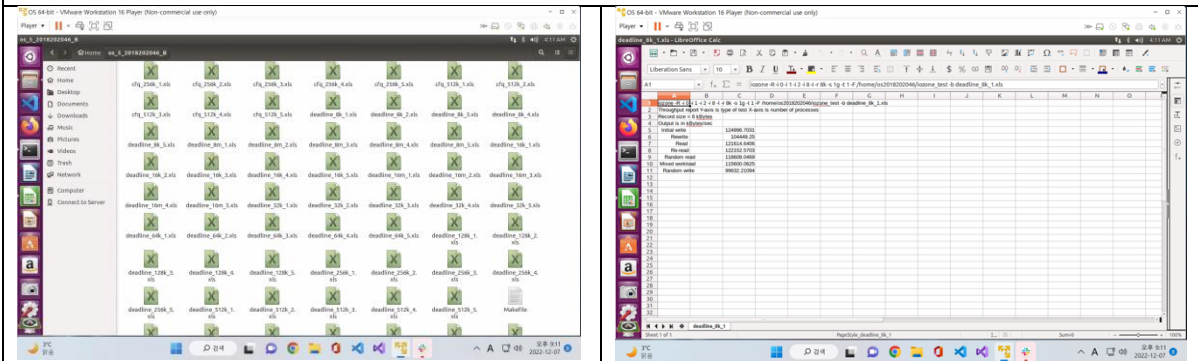
B. Deadline



이전과 같은 방법으로 이번에는 deadline으로 scheduler를 변경하였다. deadline scheduler는 4개의 Queue를 사용하는 scheduler다. 각각 elevator's read/write, FIFO read/write로 이루어져 있다. Deadline이 지난 요청이 없을 경우에는 정렬된(elevator) Queue에서 요청을 꺼내 처리한다. 즉 이는 모든 request가 마감 시간을 가짐을 의미하며, 마감시간은 읽기 요청의 경우 500ms, 쓰기 요청의 경우 5s의 값을 가진다. 이를 통해 starvation을 방지하며, 읽기 요청의 deadline이 짧기 때문에 이는 읽기 우선 정책으로 분류된다.



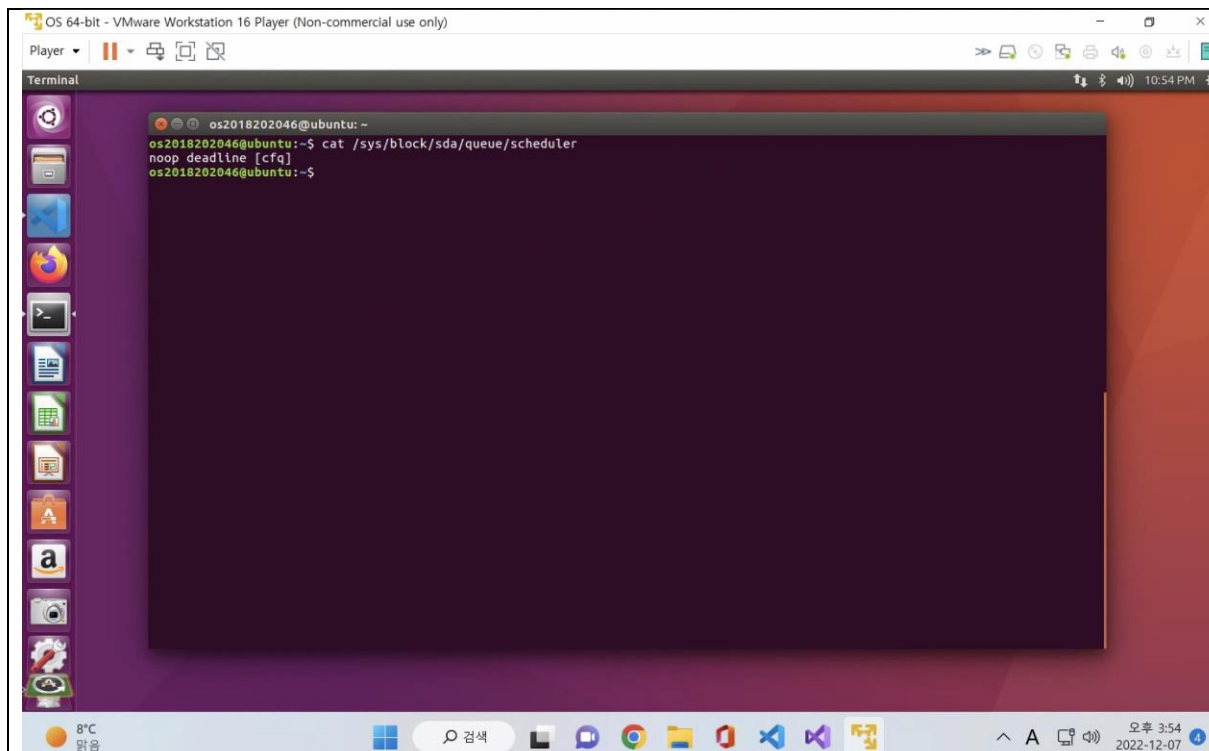
이후 Makefile에서 sched를 deadline로 하여 이름을 설정하고 record size를 8k에서 16m까지 바꾸어가며 5번씩 명령을 수행하는 메크로로 바꾸었다.



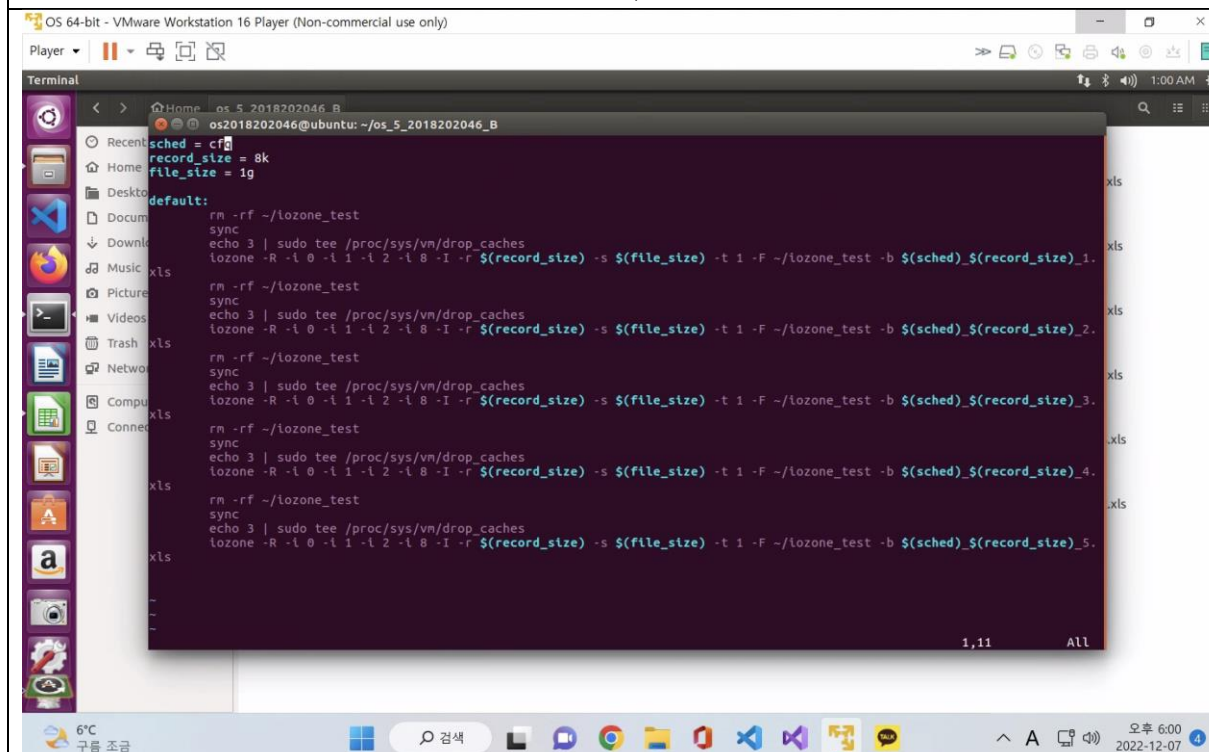
해당 make 명령을 각 record_size에 대하여 수행하였고 위와 같은 excel 파일을 얻었다. 또한 내용이 정상적으로 작성되어 있는 것을 확인할 수 있었다.

	record size		1	2	3	4	5	average(kb/yr)
deadline	8k	Initial write	124996.7031	96268.69531	96537.00781	96714.92188	96660.42969	102235.5516
	8k	Rewrit	104449.25	136843.9375	101429.5313	100702.2266	101797.1719	109044.4235
	8k	Rea	121614.6406	141807.9375	122300.0547	121451.0859	120332.3438	125501.2125
	8k	Re-res	122152.5703	123655.2969	121709.2891	120933.2266	151405.1406	127971.1047
	8k	Random	118608.0469	117660.3281	144776.5	115240.5078	133261.5781	125909.3922
	8k	Mixed work	115600.0625	128389.3359	117454.2188	116345.3906	143436.2344	124245.0484
	8k	Random v	99632.21094	115420.2266	96632.28906	97766.03125	121252.9688	106140.7453
	16k	Initial write	237058.8906	184153.0313	234218.0156	185414.6406	184045	204977.9156
	16k	Rewrit	257749.9375	259483.4688	260856.6094	196438.375	197438.7031	234393.4188
	16k	Rea	291365.1563	235774.9844	287046.3438	236588.5313	235374.2969	257229.8625
	16k	Re-res	289129.5313	235630.2969	289154.8125	236970.3594	227021.7969	255581.3594
	16k	Random	275794.8125	225283.2813	273674.9375	225989.9063	274530.75	255054.7375
	16k	Mixed work	277480.7188	223791.7344	272642.7813	226031.1875	223826.6563	244754.6157
	16k	Random v	247992.5625	188780.7188	204314.9844	188492.7656	244875.5156	214891.3094
	32k	Initial write	343822.0313	353069.0938	351199.7188	352578.875	357427.8125	351619.5063
	32k	Rewrit	369464.9688	453738.2813	375222.875	372662.0625	373355.875	388888.8125
	32k	Rea	539962.875	447152.2188	447143.25	448932.4375	447559.5	466150.0563
	32k	Re-res	446030.375	444624.375	446421	544741.0625	448010.5938	465965.4813
	32k	Random	520606.0313	516051.0313	432682.5938	430841.125	432623.1563	466560.7875
	32k	Mixed work	515162.4063	512494.5313	427769.5313	521028.9375	432665.0625	481824.0938
	32k	Random v	475605.2813	422894.0625	360549.875	448046.5938	464997.75	434418.7125
	64k	Initial write	765786.5	650533.8125	756118.5	745911.75	769899.25	737649.9625
	64k	Rewrit	857685.125	687652.1875	692128.875	845041.6875	851494.1875	786800.4125
	64k	Rea	958825.125	831768.1875	834165.5	954754.1875	822414.5	880385.5
	64k	Re-res	955569.4375	826785.625	986754.875	951658.1875	819083.625	907970.35
	64k	Random	803534.8125	802891.25	931467.25	917364.375	918604.4375	874772.425
	64k	Mixed work	933065.375	787913.625	933252.0625	909879.375	796600.0625	872142.1
	64k	Random v	784142.0625	661732.4375	661099.1875	661467.375	664734.3125	686635.075
	128k	Initial write	984956.75	980700.6875	978110.9375	856369.8125	979690.125	955965.6625
	128k	Rewrit	1059932	1064091.5	992699.875	1056840.75	1058795.5	1046471.925
	128k	Rea	1185173	1194822	1075705.25	1197254.625	1178976.75	1166386.325
	128k	Re-res	1094974.625	1090578.5	1073905	1126912	1177755.625	1112825.15
	128k	Random	1056437	1059773.625	1057510.75	1031965.375	1060126.125	1053162.575
	128k	Mixed work	1051380	1150395.125	1048376.813	1014626.438	1050405.75	1063036.825
	128k	Random v	871406.75	877559	879225.625	1040372.688	1019411.375	937595.0876
	256k	Initial write	1411319	1419882.625	1397250	1305381.125	1366066	1379979.75
	256k	Rewrit	1329372.5	1461348.625	1454101.625	1454755.875	1360454.75	1412006.675
	256k	Rea	1631672.25	1744460.625	1725110.625	1763497.75	1656073.25	1704162.9
	256k	Re-res	1623841.125	1613325.75	1608045.125	1743642.5	1628479.25	1643466.75
	256k	Random	1599004.25	1697258	1584179	1706949.875	1594183.75	1636314.975
	256k	Mixed work	1591553.625	1679382.25	1579865.5	1692952.5	1596871.875	1628125.15
	256k	Random v	1316871.125	1406559.375	1299547.125	1313700.75	1311356	1329606.875
	512k	Initial write	1824988.5	1763575.5	1795439.25	1773588.375	1798269.25	1791172.175
	512k	Rewrit	1598053.75	1845896.125	1656311.5	1696176	1796666.375	1718620.75
	512k	Rea	2258498.75	2341434.25	2216356.5	2221432.25	2273104	2262165.15
	512k	Re-res	2200680.75	2242007.25	2226010.5	2347793.75	2273400	2257978.45
	512k	Random	2231696.75	2197010	2185448.5	2323045	2235365	2234513.05
	512k	Mixed work	2178044.5	2270835.25	2170043.75	2263237.25	2184405.5	2213313.25
	512k	Random v	1841655.75	1746168.625	1736257.125	1744393.5	1754293.375	1764553.675
	8m	Initial write	2153779	2093788.625	2067879.625	2089263	2120836.75	2105109.4
	8m	Rewrit	2100930	2028510.875	2059754.25	2046549.5	2057806.5	2058710.225
	8m	Rea	2709800.75	2616299.75	2675572.5	2683524.5	2614009.25	2659841.35
	8m	Re-res	2664871	2622515.25	2638218.25	2651146	2581797	2631709.5
	8m	Random	2638020.5	2681217.75	2663449.25	2595730.25	2521184.25	2619920.4
	8m	Mixed work	2720663.25	2620418.25	2658297.75	2613821.25	2581034.75	2638847.05
	8m	Random v	1133050.5	2067402.375	2015764.375	2036959.875	2031462.375	1856927.9
	16m	Initial write	2058385.375	2058885.5	2042101.25	2072485.875	2070492.5	2060470.1
	16m	Rewrit	2022485.75	1995262.375	2022591.875	1830301.875	2001752.75	1974478.925
	16m	Rea	2520427.25	2526293.5	2512071.75	2552726.75	2441757.5	2510655.35
	16m	Re-res	2475467.75	2529787	2529223.75	2510133.25	2206534.5	2450229.25
	16m	Random	2466379	2541984.25	2489539	2581898.5	2572839	2530527.95
	16m	Mixed work	2471396.25	2532040	2522095.25	2513629.5	2503268.25	2508485.85
	16m	Random v	2056370	2050799.875	2007420.25	2035797.375	2070120.25	2044101.55

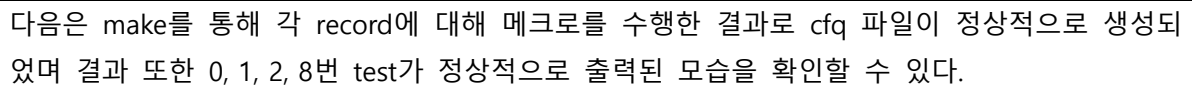
Operation	Access Count (게열6)
Initial write	50,000
Read	50,000
Random read	50,000
Random write	50,000
Rewrite	100,000
Re-read	100,000
Mixed workload	100,000
Initial write	200,000
Read	200,000
Random read	200,000
Random write	200,000
Rewrite	400,000
Re-read	400,000
Mixed workload	400,000
Initial write	600,000
Read	600,000
Random read	600,000
Random write	600,000
Rewrite	800,000
Re-read	800,000
Mixed workload	800,000
Initial write	1,000,000
Read	1,000,000
Random read	1,000,000
Random write	1,000,000
Rewrite	1,200,000
Re-read	1,200,000
Mixed workload	1,200,000
Initial write	1,400,000
Read	1,400,000
Random read	1,400,000
Random write	1,400,000
Rewrite	1,600,000
Re-read	1,600,000
Mixed workload	1,600,000
Initial write	1,800,000
Read	1,800,000
Random read	1,800,000
Random write	1,800,000
Rewrite	2,000,000
Re-read	2,000,000
Mixed workload	2,000,000
Initial write	2,200,000
Read	2,200,000
Random read	2,200,000
Random write	2,200,000
Rewrite	2,400,000
Re-read	2,400,000
Mixed workload	2,400,000
Initial write	2,600,000
Read	2,600,000
Random read	2,600,000
Random write	2,600,000
Rewrite	2,800,000
Re-read	2,800,000
Mixed workload	2,800,000
Initial write	3,000,000
Read	3,000,000
Random read	3,000,000
Random write	3,000,000



현재 I/O zone은 CFQ로 설정되어 있다. 이는 linux의 기본 scheduler로 입출력을 요청하는 모든 프로세스들에 대해 디스크 I/O 대역폭을 공평하게 할당하는 것을 보장하는 기법이다. 이는 I/O를 요청한 프로세스 별로 Queue를 할당하며 각 프로세스에 관한 큐는 섹터 순으로 정렬되어 있다. 내부에서 큐는 RR 방식으로 4개씩 처리하며, 처리 크기는 설정이 가능하다.



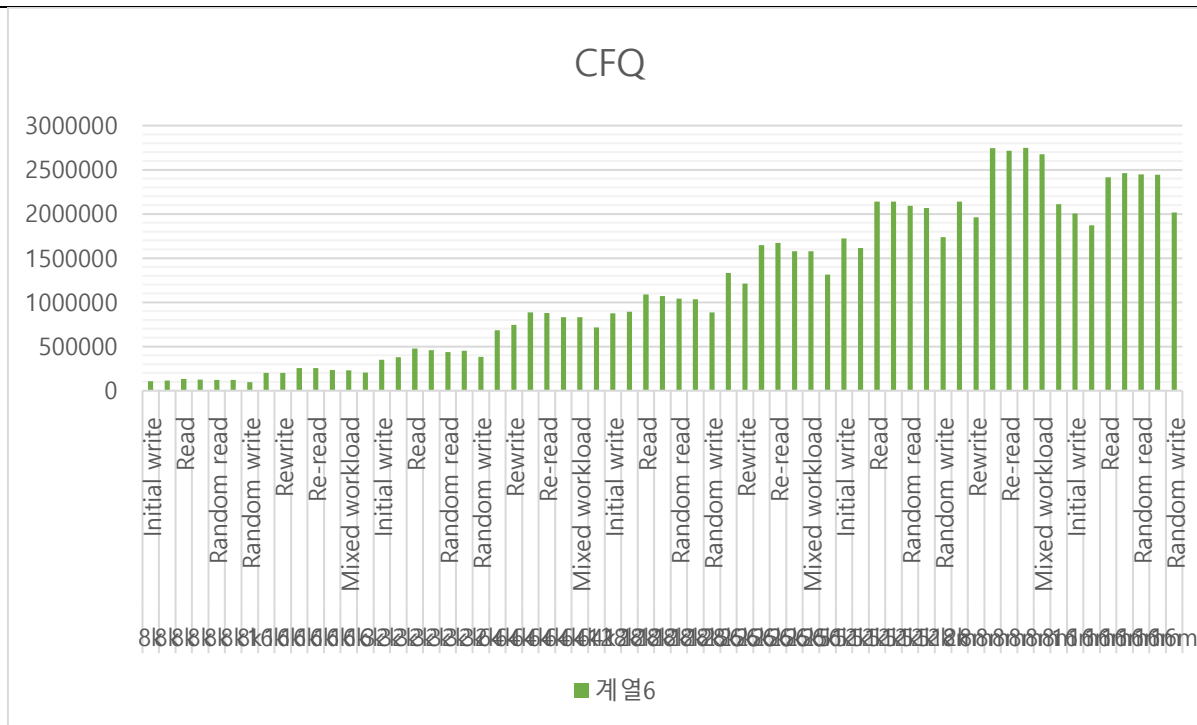
이후 Makefile에서 sched를 cfq로 하여 이름을 설정하고 record size를 8k에서 16m까지 바꾸어 가며 5번씩 명령을 수행하는 메크로로 바꾸었다.



다음은 make를 통해 각 record에 대해 매크로를 수행한 결과로 cfq 파일이 정상적으로 생성되었으며 결과 또한 0, 1, 2, 8번 test가 정상적으로 출력된 모습을 확인할 수 있다.

	record size		1	2	3	4	5	average(kb/s)
CFQ	8k	Initial write	107495.1172	101053.3906	118539.5781	118532.0938	99544.875	109033.0109
	8k	Rewrite	131031.3438	130358.5781	96474.73438	95941.625	119613.5313	114683.9625
	8k	Read	115250.5078	141945.625	113765.0547	146190.25	143802.0469	132190.6969
	8k	Re-read	143634.3906	141463.3438	114257.3516	115689.5859	114507.0781	125910.35
	8k	Random	108489.6875	108841.7578	137033.8125	139363.3594	109203.0234	120586.3281
	8k	Mixed work	129837.1328	109230.5547	138220.8906	109714.3984	131181.4844	123636.8922
	8k	Random v	83358.04688	91825.90625	126485.1641	92306.49219	92505.80469	97296.28282
	16k	Initial write	177027.8594	218061.5625	213402.0625	225931.0469	177391.2813	202362.7625
	16k	Rewrite	189762.9219	188904.4375	249518.3906	188912.5	190532.6406	201526.1781
	16k	Read	224914	227265.4063	276240.4063	270332.8438	280159.4375	255782.4188
	16k	Re-read	224755.7656	228061.0313	271217.7813	268691.8125	281216.5313	254788.5844
	16k	Random	214880.3125	216463.3125	263202.875	256143.5938	218151.125	233768.2438
	16k	Mixed work	215661.5938	215565.4219	219634.9219	244449.5938	264457.2813	231953.7625
	16k	Random v	181990.8906	181370.9531	183258.4688	235058.5469	240764.4219	204488.6563
	32k	Initial write	336601.8438	340873.625	410132.5	334695.5313	337399.1875	351940.5375
	32k	Rewrite	359447.2813	358883.4375	468173.0313	358155.5625	359154.7188	380762.8063
	32k	Read	513001.6875	424055.5	508267.4063	429894.5625	511324.6563	477308.7625
	32k	Re-read	507777.875	426810.4375	431200.25	414403.1875	512643.8125	458567.1125
	32k	Random	481847.5938	412517.4063	412288.9063	465889.4688	415235.6875	437555.8125
	32k	Mixed work	472365.375	409785.5313	484194.9688	411005.125	476388.3125	450747.8625
	32k	Random v	441712.3125	345173.0313	437684.4063	343651.0625	344288.9063	382501.9438
	64k	Initial write	721281.5	733305.375	615392.25	615143.75	729108.8125	682846.3375
	64k	Rewrite	817554.75	658658.1875	769138	830596	659176.6875	747024.725
	64k	Read	904773.1875	901031.3125	792982.0625	911291.625	922055.1875	886426.675
	64k	Re-read	902343.125	901505.4375	790599.9375	907657.5625	890826.5625	878586.525
	64k	Random	876472.5625	872199.8125	765197.8125	881043.125	772892.875	833561.2375
	64k	Mixed work	863501.6875	756407.0625	795004.4375	863648.875	880569.5	831826.3125
	64k	Random v	761735.4375	767493.125	635408.9375	769746.875	638790.5	714634.975
	128k	Initial write	815799.75	951550.125	838770.875	823026.375	945933.625	875016.15
	128k	Rewrite	854159.9375	1008393.438	874571.9375	878982.4375	855160.125	894253.575
	128k	Read	1051439.125	1151176.75	1050152.125	1050558.25	1150553.125	1090775.875
	128k	Re-read	1051602.5	1061393.375	1165017.125	1052013	1028487.625	1071702.725
	128k	Random	1008176.375	1056784.125	1022630.75	1018460.063	1110929.25	1043396.113
	128k	Mixed work	1022779.375	996670.1875	1018912.375	1018592.813	1116593.375	1034709.625
	128k	Random v	829040.1875	965271.125	816337.125	843668.375	969633.375	884790.0375
	256k	Initial write	1370005.625	1272477.375	1344859	1348125.625	1333966.25	1333886.775
	256k	Rewrite	1290418.875	1415909.875	636045.6875	1411294.125	1301825.125	1211098.738
	256k	Read	1593532.25	1694892.875	1704926.375	1676147.5	1563794.5	1646658.7
	256k	Re-read	1580249.25	1690537.625	1823769.25	1690047.125	1573819.125	1671684.475
	256k	Random	1544715.875	1580722.875	1639904.5	1647914.625	1482629	1579177.375
	256k	Mixed work	1541277.75	1632934.75	1553271.125	1632901.375	1525325.875	1577142.175
	256k	Random v	1270659.375	1378912	1281249.625	1366394.125	1270330.125	1313509.05
	512k	Initial write	1740768	1667010.375	1711798.5	1770531.75	1720336.375	1722089
	512k	Rewrite	1595144.125	1633010.625	1771123.625	1507042.375	1576038.375	1616471.825
	512k	Read	2177777.25	2127680.25	2082202.625	2085133.5	2225331.5	2139625.025
	512k	Re-read	2180845	2118605.5	2054882.75	2132357.5	2208290.5	2138996.25
	512k	Random	2110945.5	2095740.25	2142627.5	2034513.75	2082736	2093312.6
	512k	Mixed work	2045990.625	2053490.625	2125458.25	2066147.5	2045599.5	2067337.3
	512k	Random v	1752184.75	1736015.875	1754847.75	1670013.375	1771064.375	1736825.225
	8m	Initial write	2129412	2148950.75	2129153.25	2124592.75	2160970	2138615.75
	8m	Rewrite	2101157	2063971.625	1829854.875	2062599.125	1763696.5	1964255.825
	8m	Read	2714852.25	2748295	2764838	2753715.5	2736789	2743697.95
	8m	Re-read	2699294.75	2761751	2719998.5	2708609.25	2694626	2716855.9
	8m	Random	2757733.5	2751655	2749260.5	2752312.75	2729494	2748091.15
	8m	Mixed work	2508801.5	2713073.75	2705052.25	2714562.5	2733735.5	2675045.1
	8m	Random v	2116407.5	2114076.75	2106136.5	2107359.75	2116633.5	2112122.8
	16m	Initial write	1970766.375	1999644.875	2039665.5	2065064.875	1947430.125	2004514.35
	16m	Rewrite	1985724.875	1806787.125	1844960.75	1945489.25	1779684.5	1872529.3
	16m	Read	2452833	2402787	2408559.75	2388512.5	2430478	2416634.05
	16m	Re-read	2505874	2438225.75	2461013.25	2472700.75	2429485.75	2461459.9
	16m	Random	2466895	2423842.25	2479337.5	2407742.25	2461804	2447924.2
	16m	Mixed work	2432976	2454199	2452511.5	2469958.25	2420608.5	2446050.65
	16m	Random v	1993585.5	2004538.25	2037747	2046737.125	1996545.75	2015830.725

위의 Excel 파일을 하나로 정리하여 평균을 보았을 때 다음과 같은 결과를 보였다.



다음 그래프는 위의 표를 보기 쉽게 정리한 것이다. 일반적으로 write 동작이 read 동작에 비해 Kbytes/sec가 더 낮게 나오는 것을 확인할 수 있으며 이 차이는 record size가 커질수록 더 심해진다. 또한 record size가 커질수록 성능이 더 좋아지다 8MB를 기점으로 이보다 더 클 경우 읽기 성능이 소폭 하락하는 경향을 확인할 수 있다. 또한 대부분의 경우에서 random read/write의 성능이 일반적인 random read/write 성능과 유사하게 나오는 것을 확인할 수 있었다. 해당 알고리즘은 내부적으로 각 프로세스의 queue에서 RR 알고리즘에 따라 수행하기에 프로세스가 많아질 때 해당 알고리즘이 어떻게 변하는지 더욱 세밀하게 확인이 가능할 것으로 예상된다.

3. Consideration

해당 과제를 통해 I/O scheduler가 동작하는 원리를 이해하고, 어떠한 scheduler가 존재하는지 알 수 있었다. 또한 이러한 각각의 scheduler에서 어떠한 장단점이 있는지를 알 수 있었다. 특히 linux에서 기본적으로 제공하는 scheduler에서 noop, deadline, CFQ의 성능을 직접 비교해볼 수 있었다. 또한 record size가 커짐에 따라 성능이 증가하지만, 일정 크기 이상이 될 경우 read 성능이 보통 감소한다는 것을 확인할 수 있었다. 마지막으로 기본적으로 write 동작이 read 동작보다 느리다는 것을 볼 수 있었던 과제였다.

4. Reference

- 강의자료만을 참고