COEN 241 HW1

1. Detailed information of experiment device

Figure 1 detailed information

CPU: Intel Core i5 2.4GHz, Quad-Core

Memory: 8GB

2. Steps for enabling QEMU container

a. Install QEMU on Mac using command:

```
brew install qemu
```

b. Create QEMU image using command:

```
sudo qemu-img create ubuntu.img 10G -f qcow2
```

```
(base) Haoyuan@Haoyuans-MacBook-Pro-5 ~ % sudo qemu-img create ubuntu.img 10G -f
 qcow2
Password:
Formatting 'ubuntu.img', fmt=qcow2 cluster_size=65536 extended_l2=off compressio
n_type=zlib size=10737418240 lazy_refcounts=off refcount_bits=16
```

Figure 2 execution output

c. Install the VM using command:

```
sudo qemu-system-x86_64 -hda ubuntu.img -boot d -cdrom
/Users/Haoyuan/Downloads/ubuntu-20.04.5-live-server-amd64.iso -m 2046 -boot strict=on
```

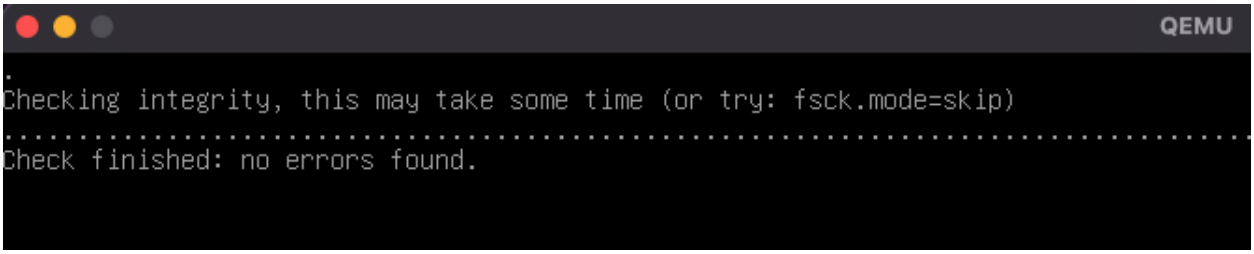


Figure 3 integrity check

d. Wait for the integrity check and then configure the VM:

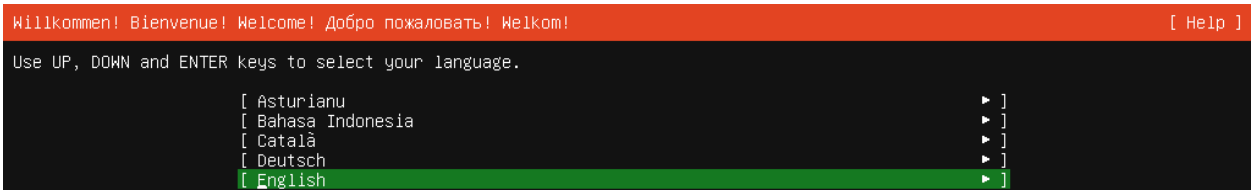


Figure 4 configuration

Figure 5 configuration

Figure 6 configuration

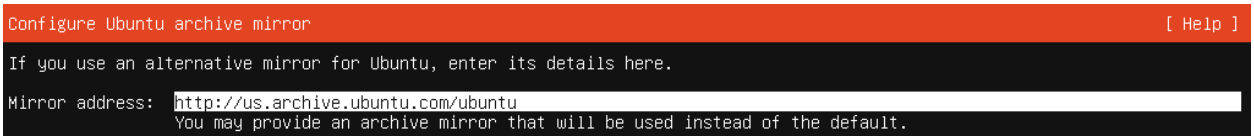


Figure 7 configuration

Figure 8 configuration

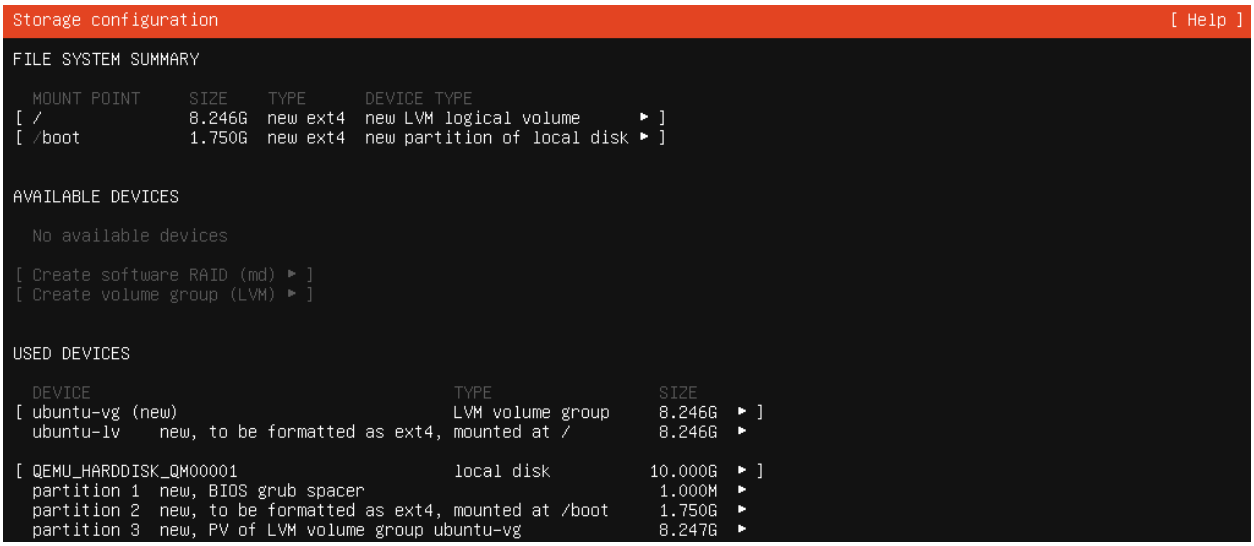


Figure 9 configuration

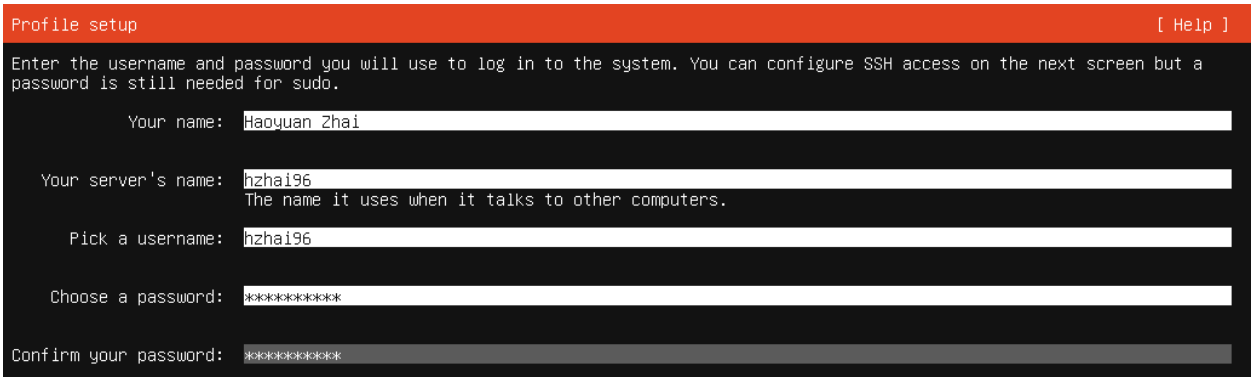


Figure 10 configuration

For other configuration, leave blank.

e. Start the VM in terminal using command:

```
sudo qemu-system-x86_64 -hda ubuntu.img -m 2046
```

Figure 11 successfully booted

3. Steps for enabling Docker container

a. Pull the latest version of Ubuntu image using command:

docker pull ubuntu

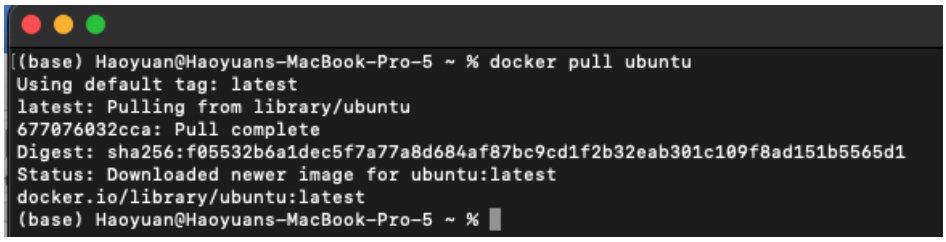


Figure 12 pull latest ubuntu image

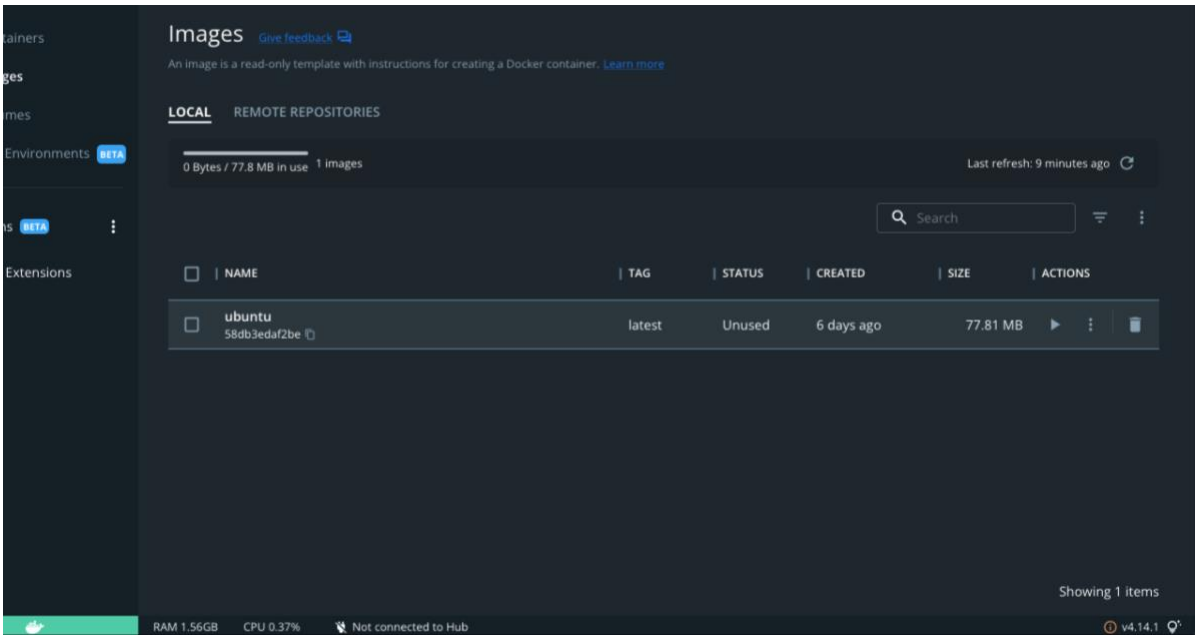


Figure 13 ubuntu image in Docker

b. Create a new container “myUbuntu” based on the Ubuntu image using command:

docker run --name myUbuntu -it ubuntu:latest bash

Figure 14 execution output

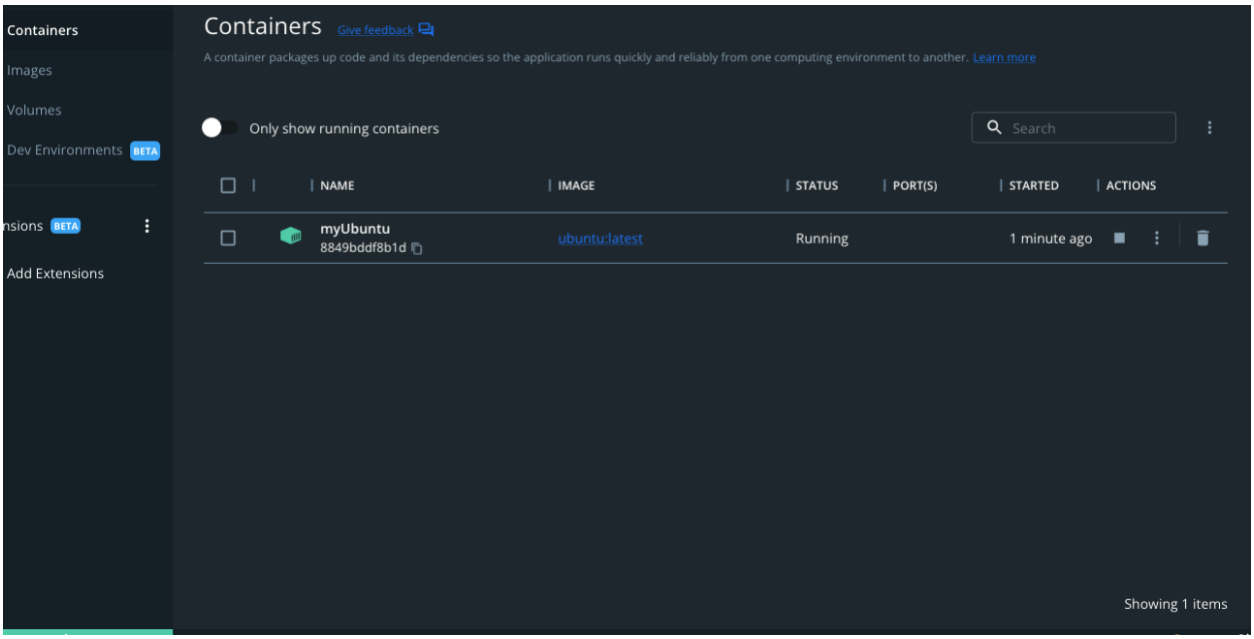


Figure 15 container in Docker

c. In the Ubuntu command line, update the package sources list with the latest versions of the packages in the repositories using command:

apt update

```
root@8849bddf8b1d:/# apt update
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy InRelease [270 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [753 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [114 kB]
Get:5 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [99.8 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy/restricted amd64 Packages [164 kB]
Get:7 http://archive.ubuntu.com/ubuntu jammy/main amd64 Packages [1792 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [681 kB]
Get:9 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [4732 B]
Get:10 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [798 kB]
Get:11 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [266 kB]
Get:12 http://archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [17.5 MB]
Get:13 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [730 kB]
Get:14 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [8978 B]
Get:15 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1007 kB]
Get:16 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1078 kB]
Get:17 http://archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [7284 B]
Get:18 http://archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [3520 B]
Fetched 25.4 MB in 4s (6196 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
root@8849bddf8b1d:/#
```

Figure 16 execution output

d. Install sysbench using command:

apt install sysbench

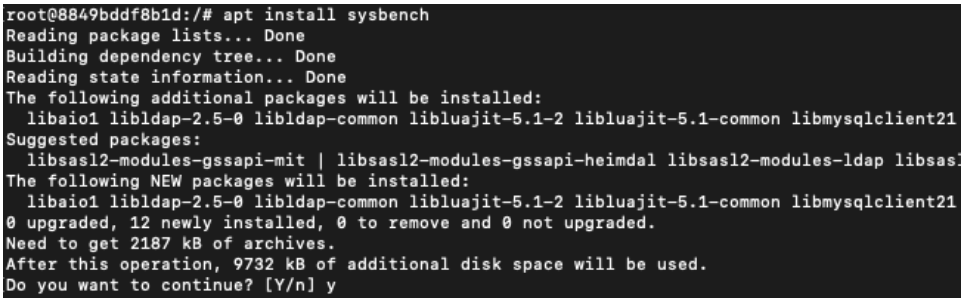


Figure 17 execution output

e. Test if sysbench has been correctly installed using command:

sysbench –version

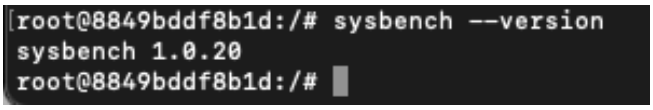


Figure 18 execution output

f. Install necessary tools to collect performance data using command:

apt-get install sysstat

apt-get install net-tools

4. Test different arguments of QEMU

I have tested two different arguments of qemu.

a. “-m” argument

I used the argument -m to set the amount of memory. The amount of the memory is set to 20. And after installing the VM, I used “free” command to check the memory of the VM.

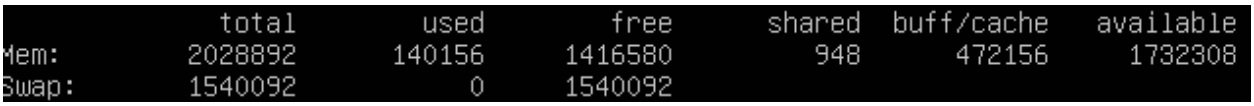


Figure 19 execution output

b. “-accel tcg” argument

I used the argument -accel to test out the function for this argument. The accelerating method I chose is TCG, which is Tiny Code Generator.

After installing, I used “top” command to monitor the CPU utilization. By testing, the VM shows that it has a better performance in CPU utilization.

5. System performance

Scenario 1: CPU utilization

In both QEMU and Docker Ubuntu, we can use the command “top” to see the CPU utilization.

a. Collect CPU utilization in QEMU Ubuntu

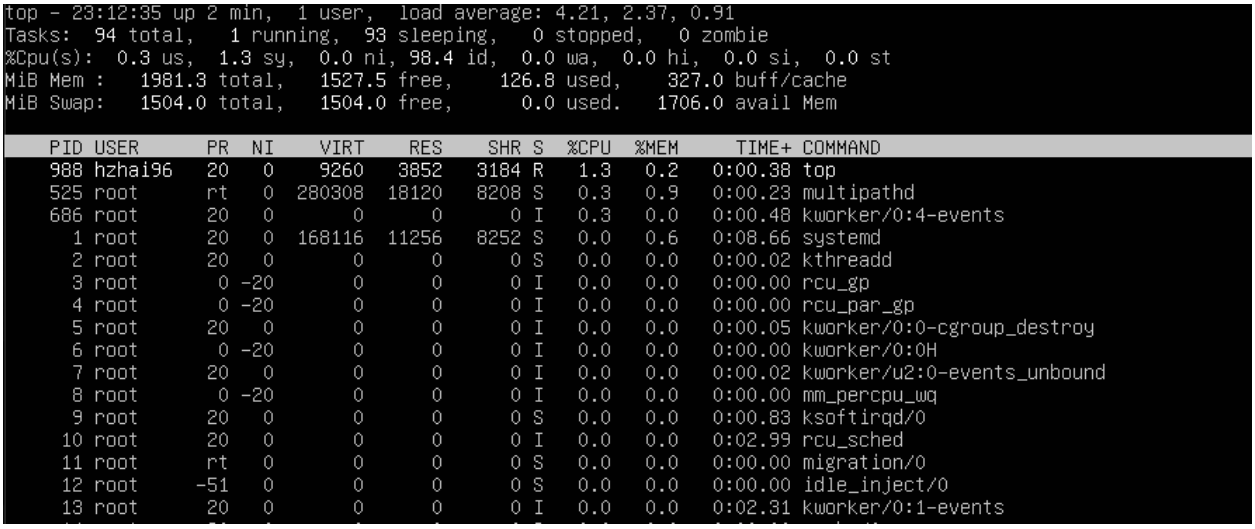


Figure 20 execution output

We can see from the figure that there are totally 94 tasks currently. One is running, the other 93 tasks are sleeping. In the %CPUs part, 0.3% of CPU is used for user-level, and 1.3% is used for kernel-level.

b. Collect CPU utilization in Docker Ubuntu

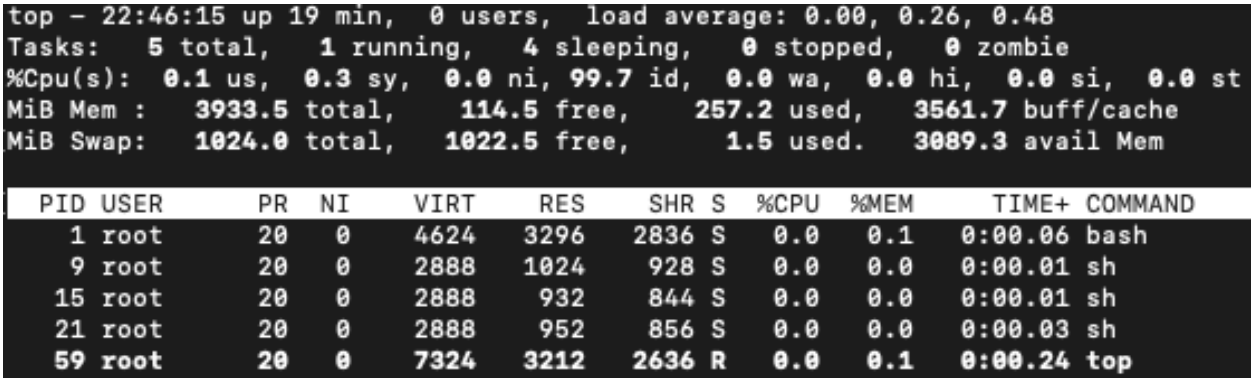


Figure 21 execution output

We can see from the figure that there are totally 5 tasks currently. One is running, the other 4 tasks are sleeping. In the %CPUs part, 0.1% of CPU is used for user-level, and 0.3% is used for kernel-level.

Scenario 2: Disk I/O performance

In both QEMU and Docker Ubuntu, we can use the command “iostat” to see the average I/O performance of the system's disks.

a. Collect disk I/O performance *in QEMU Ubuntu*

Figure 22 execution output

The figure shows that the average CPU utilization percentage for user-level processes is 24.95%. For kernel-level, it's 21.89%. For I/O waiting, it's 0.44%. For not being used, it's 38.33%.

For the disk device dm-0, the transactions per second is 43.58, the kB for read per second is 1088.17kb, for write it's 85.7. The kB that has been read is 330117, for written it's 26000kb.

For the disk device fd0, the transactions per second is 0.02, the kB for read per second is 0.07kb, for write it's 0. The kB that has been read is 20kB, for written it's 0kB.

b. Collect disk I/O performance *in Docker Ubuntu*

Figure 23 execution output

The figure shows that the average CPU utilization percentage for user-level processes is 0.45%. For kernel-level, it's 3.90%. For I/O waiting, it's 8.64%. For not being used, it's 87%.

For the disk device vda, the transactions per second is 2091.15, kB for read per second is 236.24kb, for write it's 25729.57kb. The kB that has been read is 353867kb, for written it's 38540832 kb.

Scenario 3: Memory usage

In both QEMU and Docker Ubuntu, we can use the command "free" to see the system's total available and used memory, as well as swap space.

a. Collect memory usage *in QEMU Ubuntu*

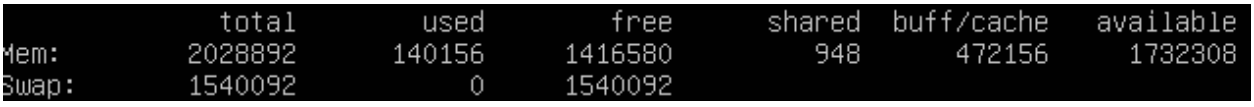
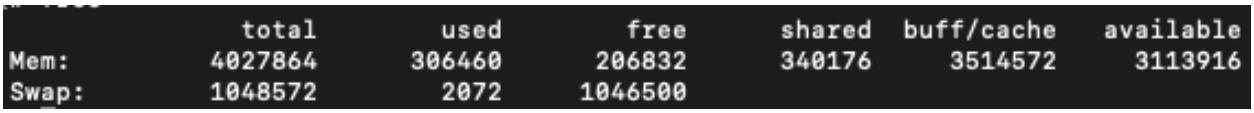


Figure 24 execution output

The figure shows that the total physical memory of the system is 2028892kB. 140156kB are used. 1416580kB are still available. 948kB are shared between processes. 472156kB are used as buffer and cache. 1732308kB are available including cache and buffer.

For the swap space usage, there are 1540092kB in total. 0kB are in use. And 1540092kB are free to use.

b. Collect* memory usage *in Docker Ubuntu



	total	used	free	shared	buff/cache	available
Mem:	4027864	306460	206832	340176	3514572	3113916
Swap:	1048572	2072	1046500			

Figure 25 execution output

The figure shows that the total physical memory of the system is 4027864kB. 306460kB are used. 206832kB are still available. 340176kB are shared between processes. 3514572kB are used as buffer and cache. 3113916kB are available including cache and buffer.

For the swap space usage, there are 1048572kB in total. 2072kB are in use. And 1046500kB are free to use.

6. Sysbench test under CPU mode

For this part of the experiment, I will conduct CPU performance test using sysbench in both QEMU Ubuntu and Docker Ubuntu. The experiment is conducted with 4 different testcases:

Single-threaded CPU performance test.

Multi-threaded CPU performance test.

Single threaded CPU performance test with large maximum prime number.

Multithreaded CPU performance test with large maximum prime number.

Test case 1: Single-threaded CPU performance test

Object: test out CPU performance with single thread.

The shell script that I created for this scenario:

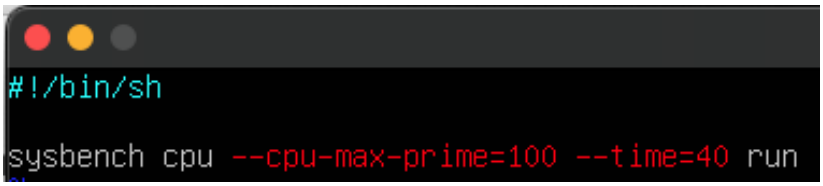


Figure 26 shell script

After test, the maximum prime number is set to 100, and the time limit is set to 40. In this case, there will be more time spent on doing I/O, so the execution time will be smaller than the time limit.

a. Results for executing five times in QEMU Ubuntu:

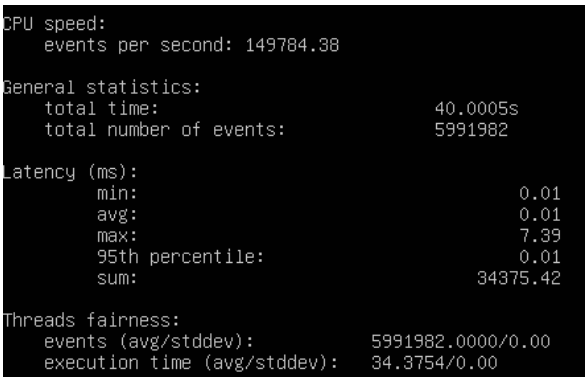


Figure 27 1st execution

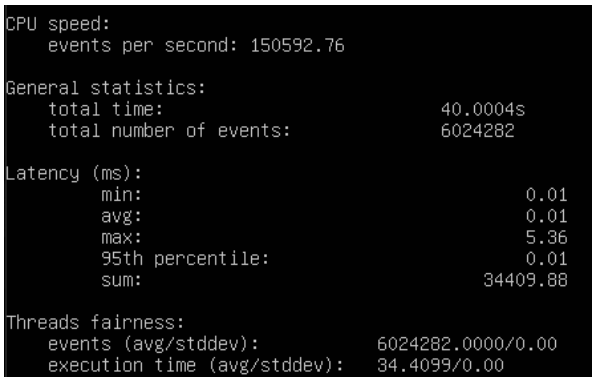


Figure 28 2nd execution

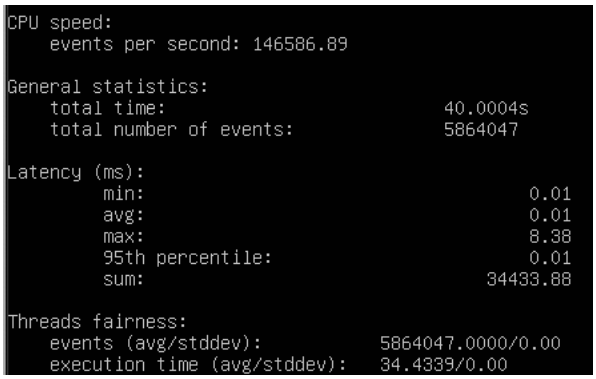


Figure 29 3rd execution

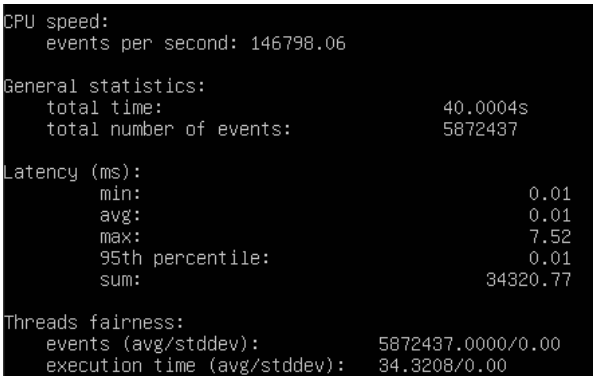


Figure 30 4th execution

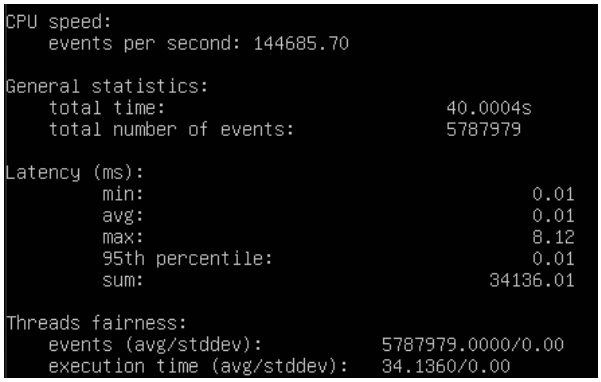


Figure 31 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th
Events/sec	149784.38	150592.76	146586.89	146798.06	144695.70
Total # of events	5991982	6024282	5864047	5872437	5787979
Min latency (ms)	0.01	0.01	0.01	0.01	0.01
Avg latency (ms)	0.01	0.01	0.01	0.01	0.01
Max latency (ms)	7.39	5.36	8.38	7.52	8.12
Execution time (s)	34.3754	34.4099	34.4339	34.3208	34.1360

	Min	Max	Mean	Variance
Events/sec	144695.7	150592.76	147691.558	4758134.44
Total # of events	5787979	6024282	5908145.4	7635201552
Min latency (ms)	0.01	0.01	0.01	0
Avg latency (ms)	0.01	0.01	0.01	0
Max latency (ms)	5.36	8.38	7.354	1.128864
Execution time (s)	34.136	34.4339	34.3352	0.01136516

b. Results for executing five times in Docker Ubuntu:

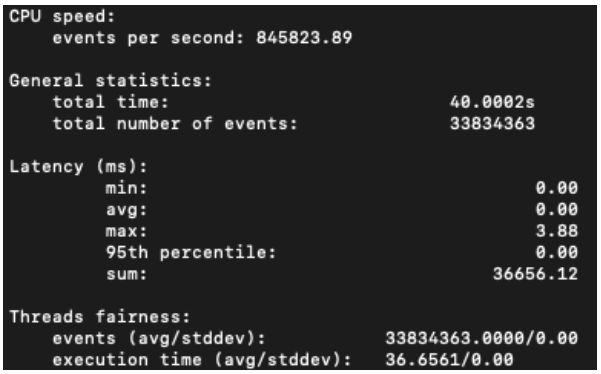


Figure 32 1st execution

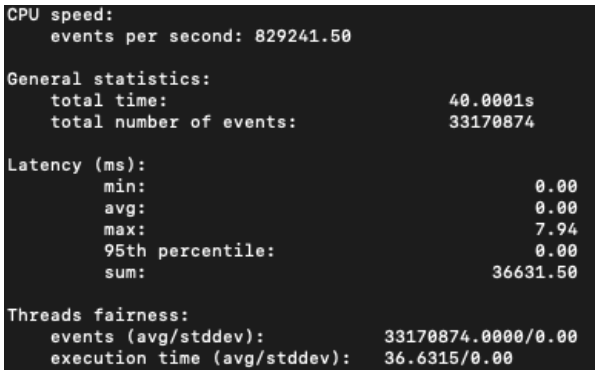


Figure 33 2nd execution

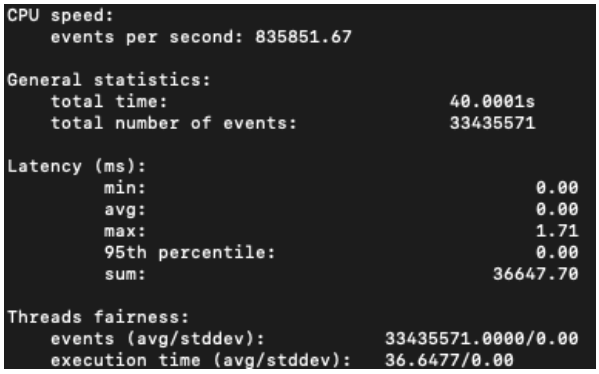


Figure 34 3rd execution

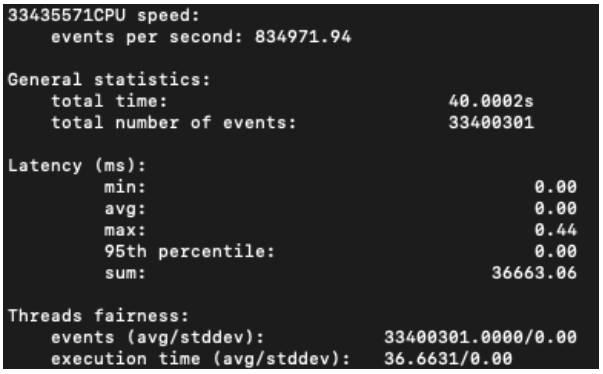


Figure 35 4th execution

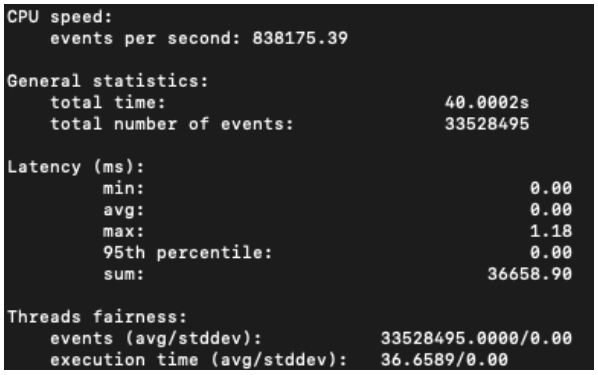


Figure 36 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th
Events/sec	845823.89	829241.50	835851.67	834971.94	838175.39
Total # of events	33834363	33170874	33435571	33400301	33528495
Min latency (ms)	0.00	0.00	0.00	0.00	0.01
Avg latency (ms)	0.00	0.00	0.00	0.00	0.01
Max latency (ms)	3.88	7.94	1.71	0.44	1.18
Execution time (s)	36.6561	36.6315	36.6477	36.6631	36.6589

	Min	Max	Mean	Variance
Events/sec	829241.5	845823.89	836812.878	28938702.9
Total # of events	33170874	33834363	33473920.8	4.6325E+10
Min latency (ms)	0	0.01	0.002	0.000016
Avg latency (ms)	0	0.01	0.002	0.000016
Max latency (ms)	0.44	7.94	3.03	7.34072
Execution time (s)	36.6315	36.6631	36.65146	0.00012498

Test case 2: Multi-threaded CPU performance test

Object: test out CPU performance with multiple threads.

The shell script that I created for this scenario:

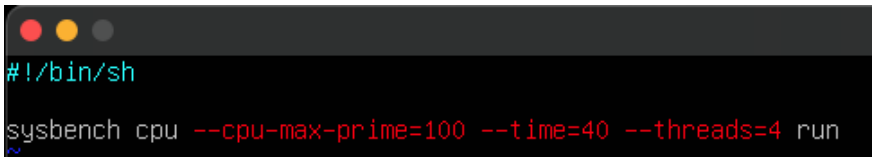


Figure 37 shell script

After test, the maximum prime number is set to 100, and the time limit is set to 40, the number of threads is 4. In this case, there will be more time spent on doing I/O, so the execution time will be smaller than the time limit.

a. Results for executing five times in QEMU Ubuntu:

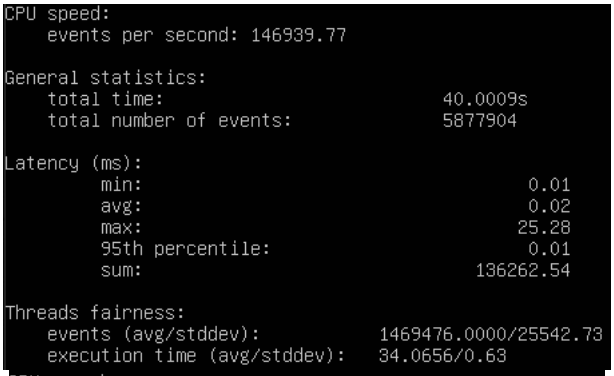


Figure 38 1st execution

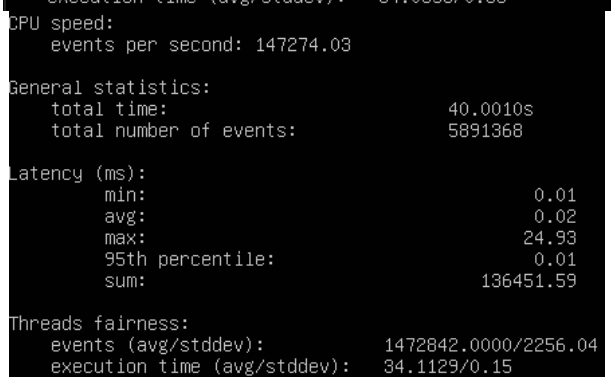


Figure 39 2nd execution

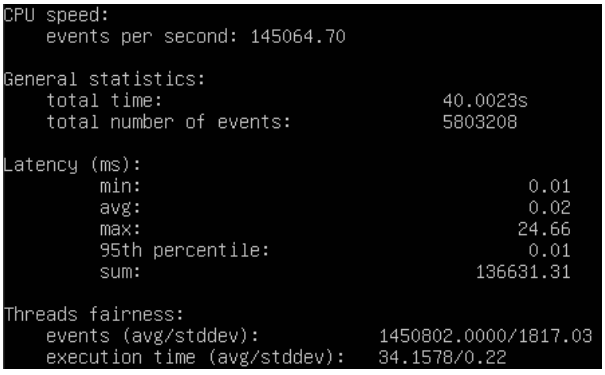


Figure 40 3rd execution

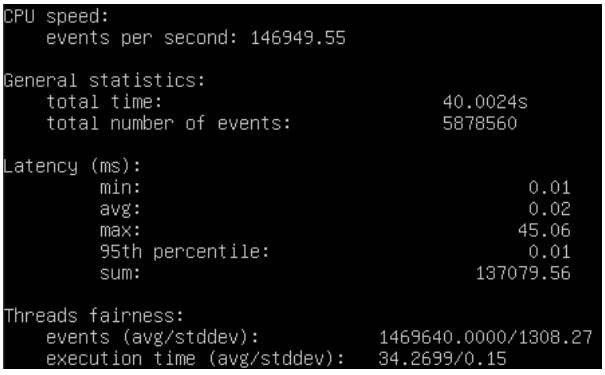


Figure 41 4th execution

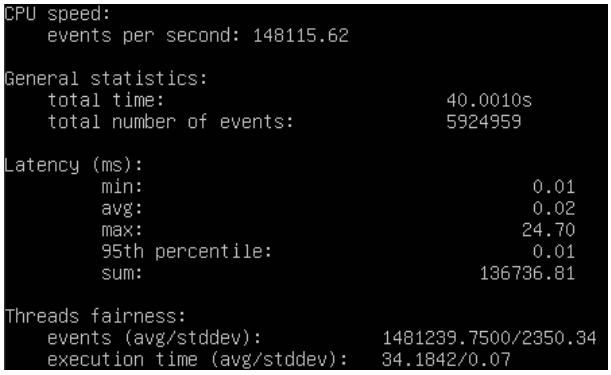


Figure 42 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th
Events/sec	146939.77	147274.03	145064.70	146949.55	148115.62
Total # of events	5877904	5891368	5803208	5878560	5924959
Min latency (ms)	0.01	0.01	0.01	0.01	0.01
Avg latency (ms)	0.02	0.02	0.02	0.02	0.02
Max latency (ms)	25.28	24.93	24.66	45.06	24.70
Execution time (s)	34.0656	34.1129	34.1578	34.2699	34.1842

	Min	Max	Mean	Variance
Events/sec	145064.7	148115.62	146868.734	997021.111
Total # of events	5803208	5924959	5875199.8	1587762317
Min latency (ms)	0.01	0.01	0.01	0
Avg latency (ms)	0.02	0.02	0.02	0
Max latency (ms)	24.66	45.06	28.926	65.125024
Execution time (s)	34.0656	34.2699	34.15808	0.00475597

b. Results for executing five times in QEMU Ubuntu:

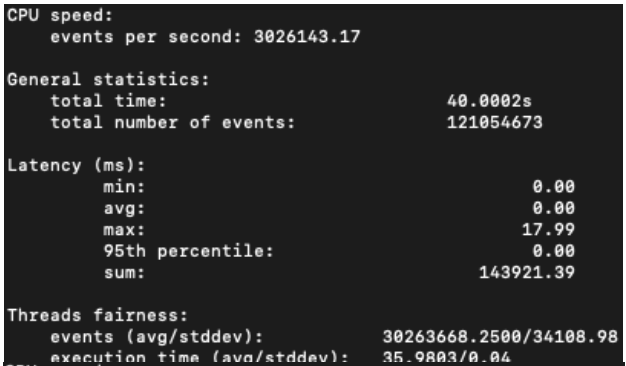


Figure 43 1st execution

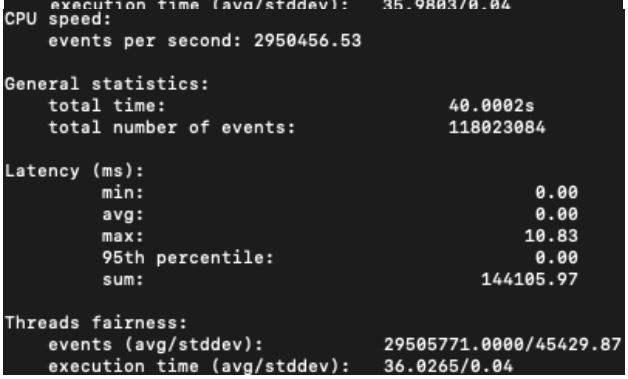


Figure 44 2nd execution

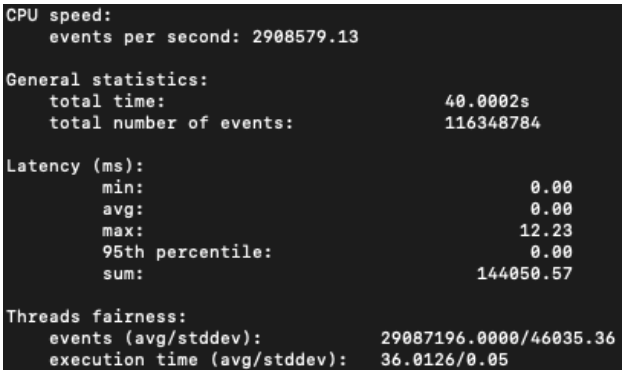


Figure 45 3rd execution

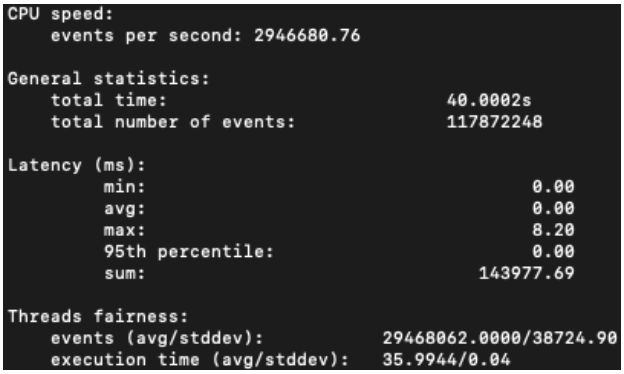


Figure 46 4th execution

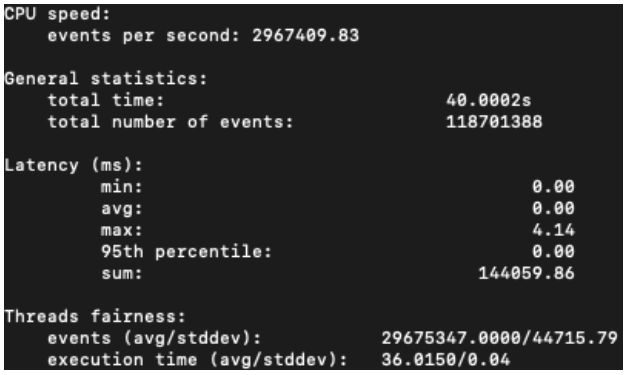


Figure 47 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th

Events/sec	3026143.17	2950456.53	2908579.13	2946680.76	2967409.83
Total # of events	121054673	118023084	116348784	117872248	118701388
Min latency (ms)	0.00	0.00	0.00	0.00	0.00
Avg latency (ms)	0.00	0.00	0.00	0.00	0.00
Max latency (ms)	17.99	10.83	12.23	8.20	4.14
Execution time (s)	35.9803	36.0265	36.0126	35.9944	36.0150

	Min	**Max**	**Mean**	**Variance**
Events/sec	2908579.13	3026143.17	2959853.88	1468460723
Total # of events	116348784	121054673	118400035	2.3532E+12
Min latency (ms)	0	0	0	0
Avg latency (ms)	0	0	0	0
Max latency (ms)	4.14	17.99	10.678	20.956616
Execution time (s)	35.9803	36.0265	36.00576	0.00026791

Test case 3: Single threaded CPU performance test (large max prime number)

Object: test out CPU performance with single thread and large max prime number.

The shell script that I created for this scenario:

Figure 48 shell script

The max prime number is set to 20000, and time limit is set to 35. In this case, since the maximum prime number is very large, there will be more time spent on CPU's calculation and less time spent on doing I/O, so the execution time is very close to the time limit.

a. Results for executing five times in QEMU Ubuntu:

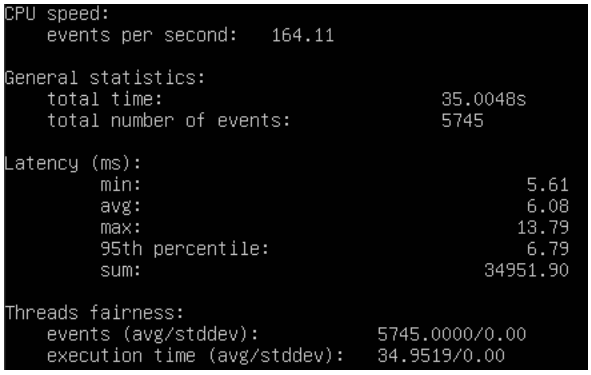


Figure 49 1st execution

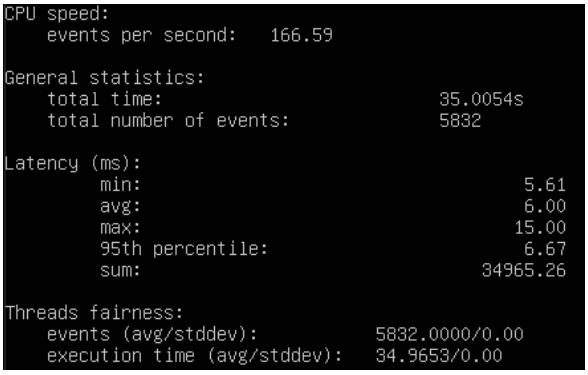


Figure 50 2nd execution

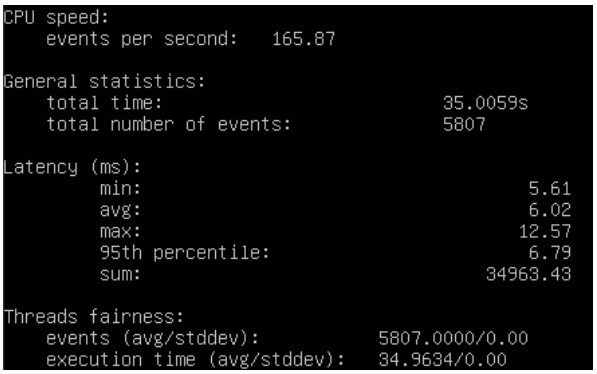


Figure 51 3rd execution

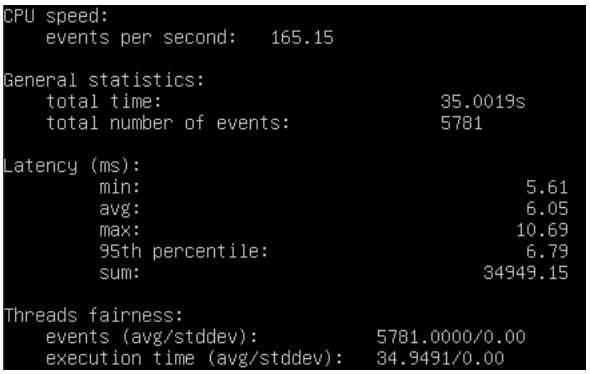


Figure 52 4th execution

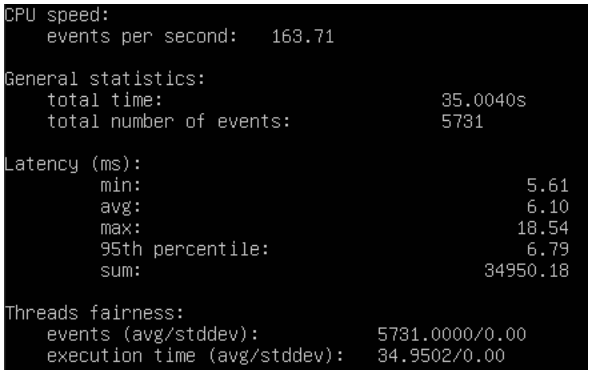


Figure 53 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th
Events/sec	164.11	166.59	165.87	165.15	163.71
Total # of events	5745	5832	5807	5781	5731
Min latency (ms)	5.61	5.61	5.61	5.61	5.61
Avg latency (ms)	6.08	6.00	6.02	6.05	6.10
Max latency (ms)	13.79	15.00	12.57	10.69	18.54
Execution time (s)	35.0048	35.0054	35.0059	35.0019	35.0040

	Min	Max	Mean	Variance
Events/sec	163.71	166.59	165.086	1.145344

Total # of events	5731	5832	5779.2	1411.36
Min latency (ms)	5.61	5.61	5.61	0
Avg latency (ms)	6	6.1	6.05	0.00136
Max latency (ms)	10.69	18.54	14.118	6.917416
Execution time (s)	35.0019	35.0059	35.0044	1.964E-06

b. Results for executing five times in QEMU Ubuntu:

```
CPU speed:
    events per second:   526.73

General statistics:
    total time:                          35.0015s
    total number of events:              18437

Latency (ms):
         min:                                    1.80
         avg:                                    1.90
         max:                                    3.85
         95th percentile:                        2.11
         sum:                                34989.58

Threads fairness:
    events (avg/stddev):           18437.0000/0.00
    execution time (avg/stddev):   34.9896/0.00
```

Figure 54 1st execution

```
CPU speed:
    events per second:   502.01

General statistics:
    total time:                          35.0016s
    total number of events:              17572

Latency (ms):
         min:                                    1.80
         avg:                                    1.99
         max:                                   21.94
         95th percentile:                        2.43
         sum:                                34979.59

Threads fairness:
    events (avg/stddev):           17572.0000/0.00
    execution time (avg/stddev):   34.9796/0.00
```

Figure 55 2nd execution

```
CPU speed:
    events per second:   507.91

General statistics:
    total time:                          35.0004s
    total number of events:              17778

Latency (ms):
         min:                                    1.80
         avg:                                    1.97
         max:                                    6.26
         95th percentile:                        2.39
         sum:                                34986.04

Threads fairness:
    events (avg/stddev):           17778.0000/0.00
    execution time (avg/stddev):   34.9860/0.00
```

Figure 56 3rd execution

```
CPU speed:
    events per second:   506.62

General statistics:
    total time:                          35.0014s
    total number of events:              17733

Latency (ms):
         min:                                    1.80
         avg:                                    1.97
         max:                                    4.85
         95th percentile:                        2.39
         sum:                                34986.20

Threads fairness:
    events (avg/stddev):           17733.0000/0.00
    execution time (avg/stddev):   34.9862/0.00
```

Figure 57 4th execution

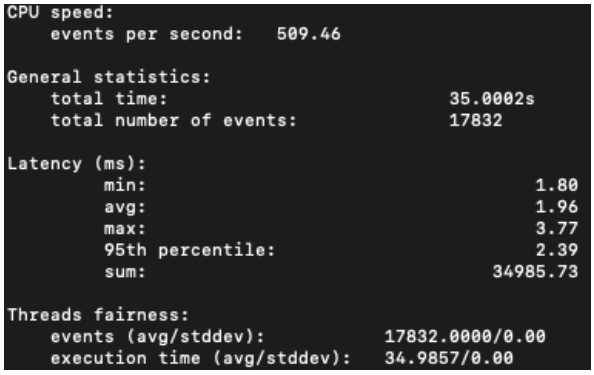


Figure 58 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th
Events/sec	526.73	502.01	507.91	506.62	509.46
Total # of events	18437	17572	17778	17733	17832
Min latency (ms)	1.80	1.80	1.80	1.80	1.80
Avg latency (ms)	1.90	1.99	1.97	1.97	1.96
Max latency (ms)	3.85	21.94	6.26	4.85	3.77
Execution time (s)	34.9896	34.9796	34.9860	34.9862	34.9857

	Min	Max	Mean	Variance
Events/sec	502.01	526.73	510.546	71.665304
Total # of events	17572	18437	17870.4	87793.84
Min latency (ms)	1.8	1.8	1.8	0
Avg latency (ms)	1.9	1.99	1.958	0.000936
Max latency (ms)	3.77	21.94	8.134	48.459864
Execution time (s)	34.9796	34.9896	34.98542	1.0474E-05

Test case 4: Multi-threaded CPU performance test (large max prime number)

Object: test out CPU performance with multiple threads and large max prime number.

The shell script that I created for this scenario:

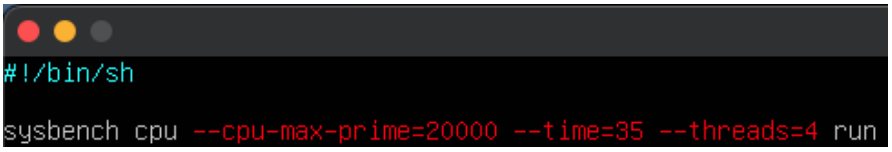


Figure 59 shell script

The max prime number is set to 20000, time limit is set to 35 and the number of threads is set to 4. In this case, since the maximum prime number is very large, there will be more time spent on CPU's calculation and less time spent on doing I/O, so the execution time is very close to the time limit.

a. Results for executing five times in QEMU Ubuntu:

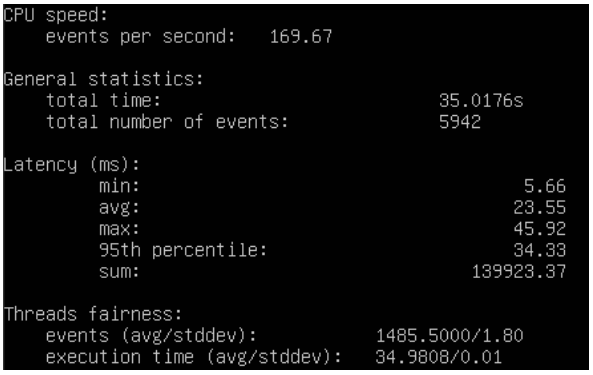


Figure 60 1st execution

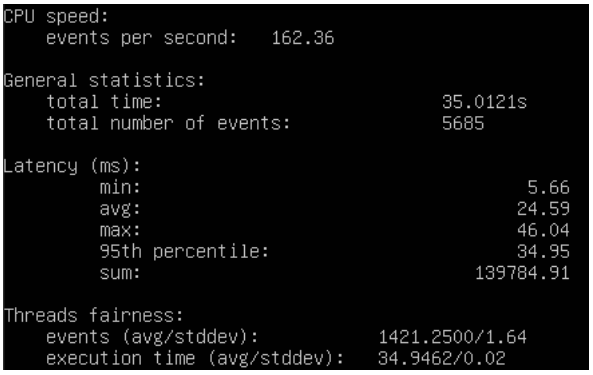


Figure 61 2nd execution

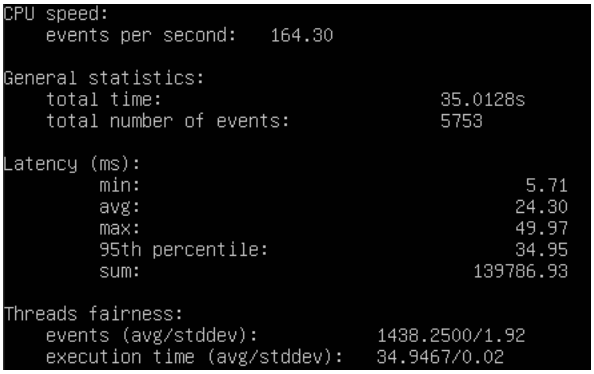


Figure 62 3rd execution

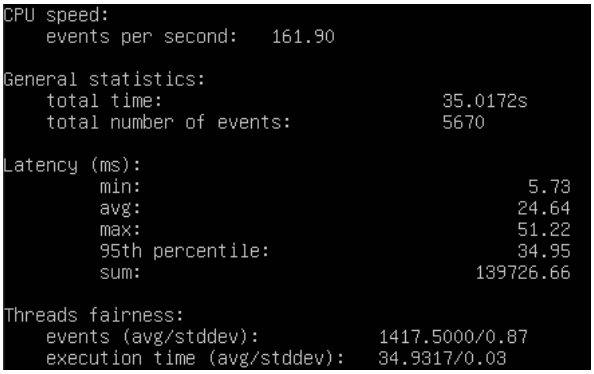


Figure 63 4th execution

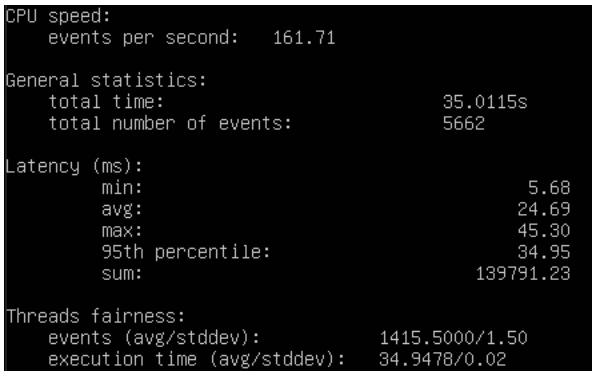


Figure 64 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th
Events/sec	169.67	162.36	164.30	161.90	161.71
Total # of events	5942	5685	5753	5670	5662
Min latency (ms)	5.66	5.66	5.71	5.73	5.68

Avg latency (ms)	23.55	24.59	24.30	24.64	24.69
Max latency (ms)	45.92	46.04	49.97	51.22	45.30
Execution time (s)	35.0176	35.0121	35.0128	35.0172	35.0115

	Min	Max	Mean	Variance
Events/sec	161.71	169.67	163.988	8.916376
Total # of events	5662	5942	5742.4	10990.64
Min latency (ms)	5.66	5.73	5.688	0.000776
Avg latency (ms)	23.55	24.69	24.354	0.179944
Max latency (ms)	45.3	51.22	47.69	5.84536
Execution time (s)	35.0115	35.0176	35.01424	6.8424E-06

b. Results for executing five times in QEMU Ubuntu:

```
CPU speed:
    events per second:  1877.81

General statistics:
    total time:                          35.0019s
    total number of events:              65730

Latency (ms):
         min:                                    1.87
         avg:                                    2.13
         max:                                   20.33
         95th percentile:                        2.61
         sum:                               139959.35

Threads fairness:
    events (avg/stddev):           16432.5000/5.55
    execution time (avg/stddev):   34.9898/0.00
```

Figure 65 1st execution

```
CPU speed:
    events per second:  1822.12

General statistics:
    total time:                          35.0016s
    total number of events:              63780

Latency (ms):
         min:                                    1.86
         avg:                                    2.19
         max:                                   12.19
         95th percentile:                        2.71
         sum:                               139951.37

Threads fairness:
    events (avg/stddev):           15945.0000/9.67
    execution time (avg/stddev):   34.9878/0.00
```

Figure 66 2nd execution

```
CPU speed:
    events per second:  1865.32

General statistics:
    total time:                          35.0015s
    total number of events:              65292

Latency (ms):
         min:                                    1.87
         avg:                                    2.14
         max:                                    8.43
         95th percentile:                        2.57
         sum:                               139955.03

Threads fairness:
    events (avg/stddev):           16323.0000/6.78
    execution time (avg/stddev):   34.9888/0.00
```

Figure 67 3rd execution

```
CPU speed:
    events per second:  1872.39

General statistics:
    total time:                          35.0019s
    total number of events:              65540

Latency (ms):
         min:                                    1.85
         avg:                                    2.14
         max:                                   14.33
         95th percentile:                        2.52
         sum:                               139960.30

Threads fairness:
    events (avg/stddev):           16385.0000/13.44
    execution time (avg/stddev):   34.9901/0.00
```

Figure 68 4th execution

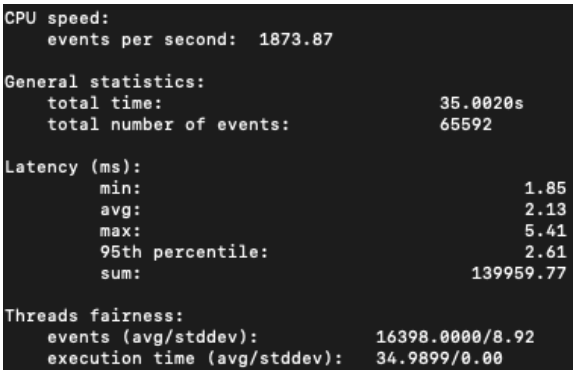


Figure 69 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th
Events/sec	1877.81	1822.12	1865.32	1872.39	1873.87
Total # of events	65730	63780	65292	65540	65592
Min latency (ms)	1.87	1.86	1.87	1.85	1.85
Avg latency (ms)	2.13	2.19	2.14	2.14	2.13
Max latency (ms)	20.33	12.19	8.43	14.33	5.41
Execution time (s)	34.9898	34.9878	34.9888	34.9901	34.9899

	Min	Max	Mean	Variance
Events/sec	1822.12	1877.81	1862.302	419.957176
Total # of events	63780	65730	65186.8	514831.36
Min latency (ms)	1.85	1.87	1.86	8E-05
Avg latency (ms)	2.13	2.19	2.146	0.000504
Max latency (ms)	5.41	20.33	12.138	26.186336
Execution time (s)	34.9878	34.9901	34.98928	7.496E-07

7. Sysbench test under fileio mode

For this part of the experiment, I will conduct fileio performance test using sysbench in both QEMU Ubuntu and Docker Ubuntu. The experiment is conducted with 3 different testcases:

Multi-threaded fileio performance test with random read write.

Multi-threaded fileio performance test with sequential read.

Multi-threaded fileio performance test with sequential write.

Test case 1: Multi-threaded fileio performance test with random read write.

Object: test out fileio performance with 4 threads under random read write mode.

The shell script that I created for this scenario:

```
#!/bin/sh

sysbench fileio --file-total-size=3G prepare
sysbench fileio --file-total-size=3G --file-test-mode=rndrw --time=40 --threads=4 run
sysbench fileio cleanup
```

Figure 70 shell script

a. Results for executing five times in QEMU Ubuntu:

```
File operations:
    reads/s:                      997.95
    writes/s:                     665.30
    fsyncs/s:                     2138.68

Throughput:
    read, MiB/s:                  15.59
    written, MiB/s:               10.40

General statistics:
    total time:                          40.1574s
    total number of events:              152171

Latency (ms):
         min:                                    0.01
         avg:                                    1.04
         max:                                   45.88
         95th percentile:                        3.25
         sum:                               157881.43

Threads fairness:
    events (avg/stddev):           38042.7500/146.11
    execution time (avg/stddev):   39.4704/0.01

sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)
```

Figure 71 1st execution

```
File operations:
    reads/s:                      1022.52
    writes/s:                     681.62
    fsyncs/s:                     2192.60

Throughput:
    read, MiB/s:                  15.98
    written, MiB/s:               10.65

General statistics:
    total time:                          40.1040s
    total number of events:              155770

Latency (ms):
         min:                                    0.01
         avg:                                    1.01
         max:                                   82.26
         95th percentile:                        3.19
         sum:                               157854.50

Threads fairness:
    events (avg/stddev):           38942.5000/437.56
    execution time (avg/stddev):   39.4636/0.00
```

Figure 72 2nd execution

```
File operations:
    reads/s:                      1019.57
    writes/s:                     679.67
    fsyncs/s:                     2185.50

Throughput:
    read, MiB/s:                  15.93
    written, MiB/s:               10.62

General statistics:
    total time:                          40.1328s
    total number of events:              155404

Latency (ms):
         min:                                    0.01
         avg:                                    1.02
         max:                                   97.12
         95th percentile:                        3.13
         sum:                               157860.95

Threads fairness:
    events (avg/stddev):           38851.0000/158.27
    execution time (avg/stddev):   39.4652/0.03
```

```
File operations:
    reads/s:                      1024.97
    writes/s:                     683.36
    fsyncs/s:                     2197.32

Throughput:
    read, MiB/s:                  16.02
    written, MiB/s:               10.68

General statistics:
    total time:                          40.0953s
    total number of events:              156095

Latency (ms):
         min:                                    0.01
         avg:                                    1.01
         max:                                   84.33
         95th percentile:                        3.19
         sum:                               157832.72

Threads fairness:
    events (avg/stddev):           39023.7500/320.51
    execution time (avg/stddev):   39.4582/0.02
```

Figure 73 3rd execution

Figure 74 4th execution

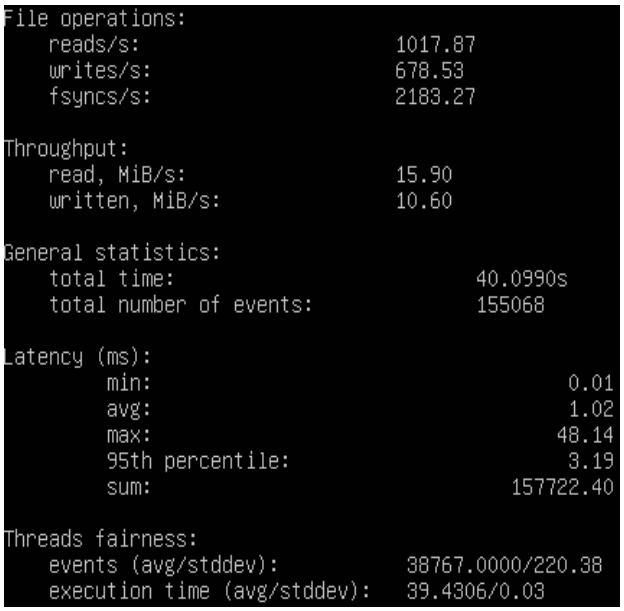


Figure 75 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th
Reads/sec	997.95	1022.52	1019.57	1024.97	1017.87
Writes/sec	665.30	681.62	679.67	683.36	678.53
Fsyncs/sec	2138.68	2192.60	2185.50	2197.32	2183.27
Throughput (r)	15.59	15.98	15.93	16.02	15.90
Throughput (w)	10.40	10.65	10.62	10.68	10.60
Total # of events	152171	155770	155404	156095	155068
Min latency (ms)	0.01	0.01	0.01	0.01	0.01
Avg latency (ms)	1.04	1.01	1.02	1.01	1.02
Max latency (ms)	45.88	82.26	97.12	84.33	48.14
Execution time (s)	39.4704	39.4636	39.4652	39.4582	39.4306

	Min	Max	Mean	Variance
Reads/sec	997.95	1024.97	1016.576	92.671344
Writes/sec	665.3	683.36	677.696	41.146344
Fsyncs/sec	2138.68	2197.32	2179.474	441.128864
Throughput (r)	15.59	16.02	15.884	0.023304
Throughput (w)	10.4	10.68	10.59	0.00976
Total # of events	152171	156095	154901.6	1982918.64

Min latency (ms)	0.01	0.01	0.01	0
Avg latency (ms)	1.01	1.04	1.02	0.00012
Max latency (ms)	45.88	97.12	71.546	427.766864
Execution time (s)	39.4306	39.4704	39.4576	0.00019739

b. Results for executing five times in Docker Ubuntu:

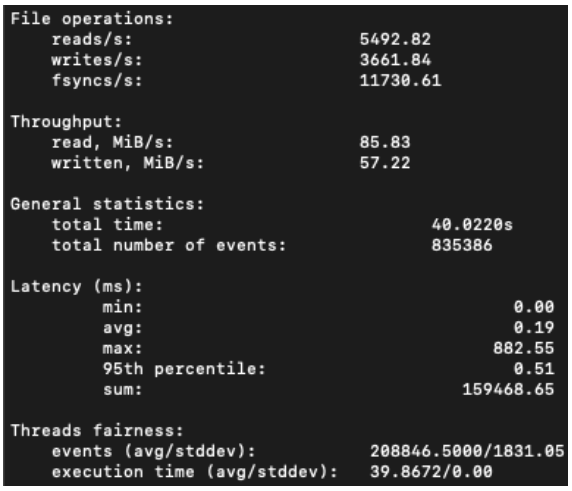


Figure 76 1st execution

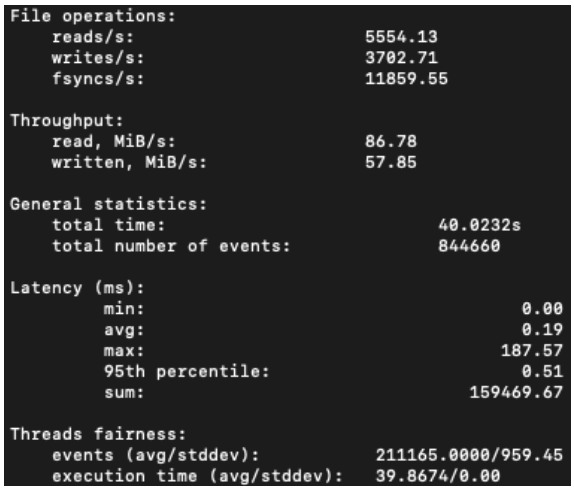


Figure 77 2nd execution

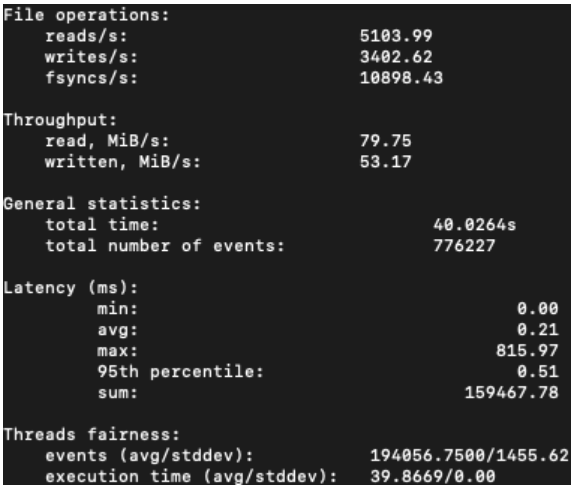


Figure 78 3rd execution

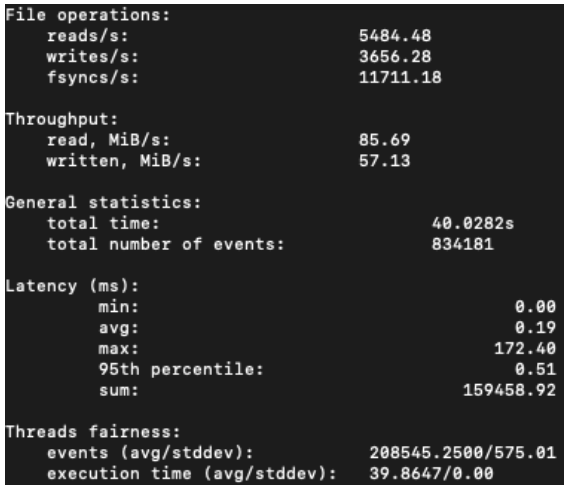


Figure 79 4th execution

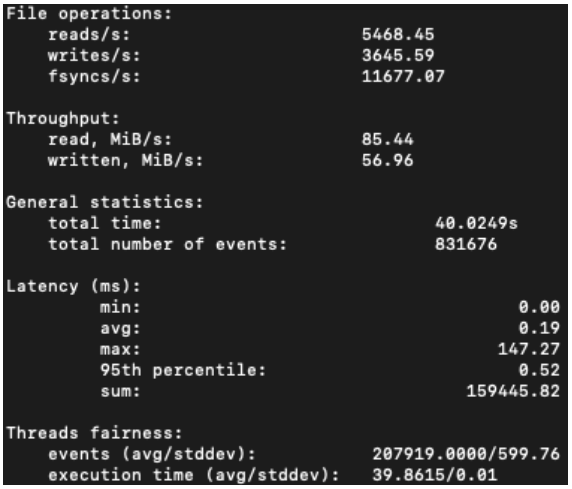


Figure 80 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th
Reads/sec	5492.82	5554.13	5103.99	5484.48	5468.45
Writes/sec	3661.84	3702.71	3402.62	3656.28	3645.59
Fsyncs/sec	11730.61	11859.55	10898.43	11711.18	11677.07
Throughput (r)	85.83	86.78	79.75	85.69	85.44
Throughput (w)	57.22	57.85	53.17	57.13	56.96
Total # of events	835386	844660	776227	834181	831676
Min latency (ms)	0.00	0.00	0.00	0.00	0.00
Avg latency (ms)	0.19	0.19	0.21	0.19	0.19
Max latency (ms)	882.55	187.57	815.97	172.40	147.27
Execution time (s)	39.8672	39.8674	39.8669	39.8647	39.8615

	Min	Max	Mean	Variance
Reads/sec	5103.99	5554.13	5420.774	25931.6014
Writes/sec	3402.62	3702.71	3613.808	11524.9953
Fsyncs/sec	10898.43	11859.55	11575.368	118378.548
Throughput (r)	79.75	86.78	84.698	6.326696
Throughput (w)	53.17	57.85	56.466	2.806504
Total # of events	776227	844660	824426	600080496
Min latency (ms)	0	0	0	0
Avg latency (ms)	0.19	0.21	0.194	6.4E-05

Max latency (ms)	147.27	882.55	441.152	111643.765
Execution time (s)	39.8615	39.8674	39.86554	5.0184E-06

Test case 2: Multi-threaded fileio performance test with sequential read.

Object: test out fileio performance with 4 threads under sequential read mode.

The shell script that I created for this scenario:

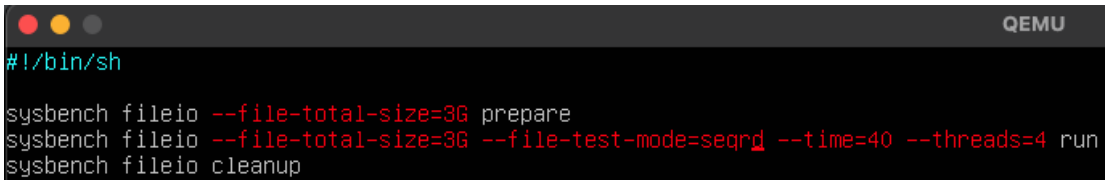


Figure 81 shell script

a. Results for executing five times in QEMU Ubuntu:

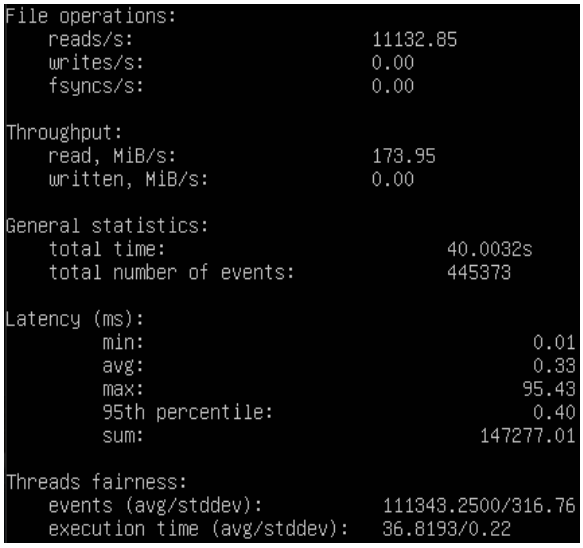


Figure 82 1st execution

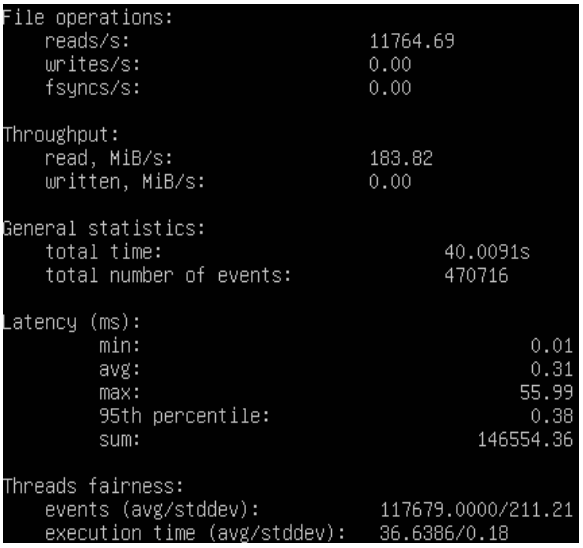


Figure 83 2nd execution

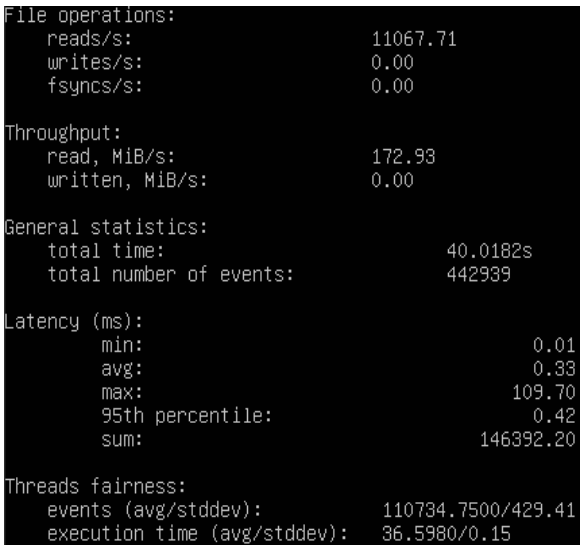


Figure 84 3rd execution

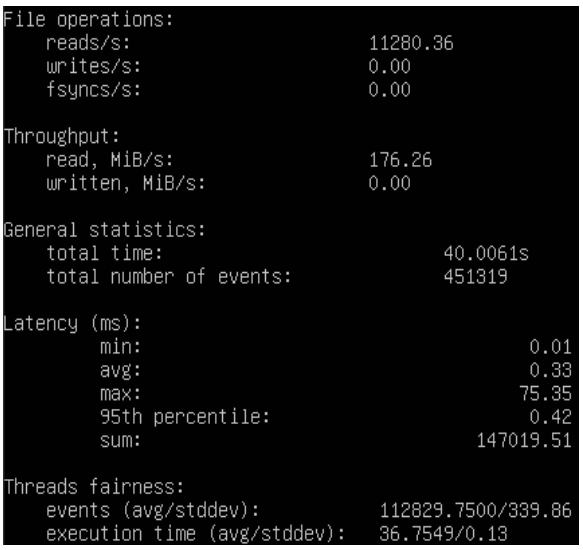


Figure 85 4th execution

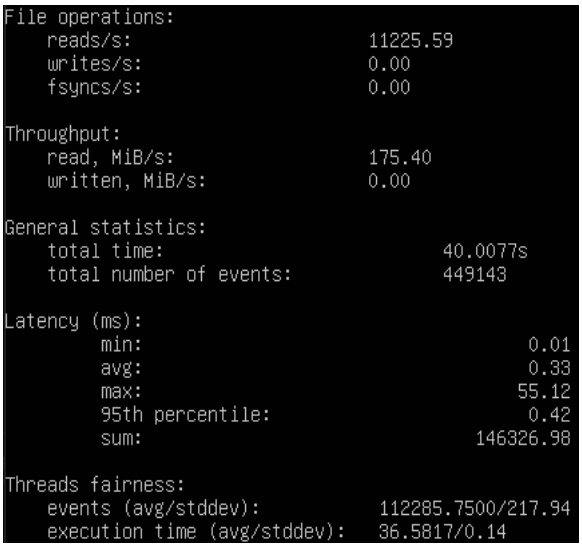


Figure 86 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th
Reads/sec	11132.85	11764.69	11067.71	11280.36	11225.59
Throughput (r)	173.95	183.82	172.93	176.26	175.40
Total # of events	445373	470716	442939	451319	449143
Min latency (ms)	0.01	0.01	0.01	0.01	0.01
Avg latency (ms)	0.33	0.31	0.33	1.33	0.33
Max latency (ms)	95.43	55.99	109.70	75.35	55.12
Execution time (s)	36.8193	36.6386	36.5980	36.7549	36.5817

	Min	Max	Mean	Variance
Reads/sec	11067.71	11764.69	11294.24	60718.2505

Throughput (r)	172.93	183.82	176.472	14.818696
Total # of events	442939	470716	451898	96976339.2
Min latency (ms)	0.01	0.01	0.01	0
Avg latency (ms)	0.31	1.33	0.526	0.161664
Max latency (ms)	55.12	109.7	78.318	464.629256
Execution time (s)	36.5817	36.8193	36.6785	0.00862082

b. Results for executing five times in Docker Ubuntu:

```
File operations:
    reads/s:                      793346.70
    writes/s:                     0.00
    fsyncs/s:                     0.00

Throughput:
    read, MiB/s:                  12396.04
    written, MiB/s:               0.00

General statistics:
    total time:                          40.0043s
    total number of events:              31739299

Latency (ms):
         min:                                    0.00
         avg:                                    0.00
         max:                                 1002.89
         95th percentile:                        0.01
         sum:                               143212.54

Threads fairness:
    events (avg/stddev):          7934824.7500/395114.04
    execution time (avg/stddev):  35.8031/0.32
```

Figure 87 1st execution

```
File operations:
    reads/s:                      813338.97
    writes/s:                     0.00
    fsyncs/s:                     0.00

Throughput:
    read, MiB/s:                  12708.42
    written, MiB/s:               0.00

General statistics:
    total time:                          40.0034s
    total number of events:              32537849

Latency (ms):
         min:                                    0.00
         avg:                                    0.00
         max:                                  685.00
         95th percentile:                        0.00
         sum:                               144194.06

Threads fairness:
    events (avg/stddev):          8134462.2500/331562.46
    execution time (avg/stddev):  36.0485/0.76
```

Figure 88 2nd execution

```
File operations:
    reads/s:                      734416.70
    writes/s:                     0.00
    fsyncs/s:                     0.00

Throughput:
    read, MiB/s:                  11475.26
    written, MiB/s:               0.00

General statistics:
    total time:                          40.0135s
    total number of events:              29388249

Latency (ms):
         min:                                    0.00
         avg:                                    0.00
         max:                                 1001.34
         95th percentile:                        0.00
         sum:                               146734.95

Threads fairness:
    events (avg/stddev):          7347062.2500/224794.56
    execution time (avg/stddev):  36.6837/0.10
```

```
File operations:
    reads/s:                      870657.53
    writes/s:                     0.00
    fsyncs/s:                     0.00

Throughput:
    read, MiB/s:                  13604.02
    written, MiB/s:               0.00

General statistics:
    total time:                          40.0028s
    total number of events:              34830422

Latency (ms):
         min:                                    0.00
         avg:                                    0.00
         max:                                  565.92
         95th percentile:                        0.01
         sum:                               143409.38

Threads fairness:
    events (avg/stddev):          8707605.5000/165956.65
    execution time (avg/stddev):  35.8523/0.33
```

Figure 89 3rd execution

Figure 90 4th execution

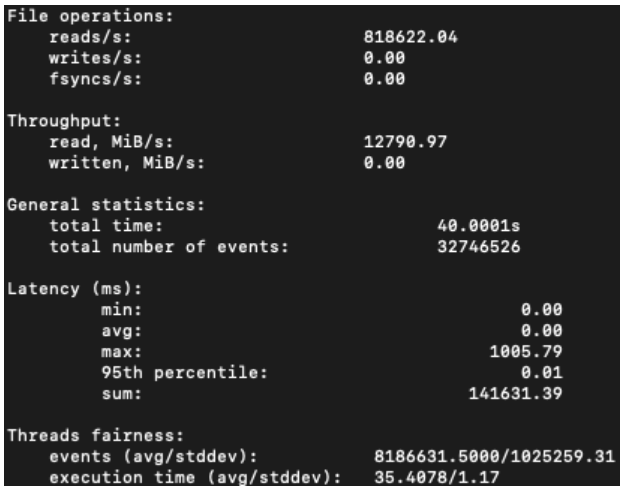


Figure 91 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th
Reads/sec	793346.70	813338.97	734416.70	870657.53	818622.04
Throughput (r)	12396.04	12708.42	11475.26	13604.02	12790.97
Total # of events	31739299	32537849	29388249	34830422	32746526
Min latency (ms)	0.00	0.00	0.00	0.00	0.00
Avg latency (ms)	0.00	0.00	0.00	0.00	0.00
Max latency (ms)	1002.89	685.00	1001.34	565.92	1005.79
Execution time (s)	35.8031	36.0485	36.6837	35.8523	35.4078

	Min	Max	Mean	Variance
Reads/sec	734416.7	870657.53	806076.388	1935603645
Throughput (r)	11475.26	13604.02	12594.942	472558.486
Total # of events	29388249	34830422	32248469	3.0877E+12
Min latency (ms)	0	0	0	0
Avg latency (ms)	0	0	0	0
Max latency (ms)	565.92	1005.79	852.188	35690.4363

Execution time (s)	35.4078	36.6837	35.95908	0.17454229

Test case 3: Multi-threaded fileio performance test with sequential write.

Object: test out fileio performance with 4 threads under sequential write mode.

The shell script that I created for this scenario:

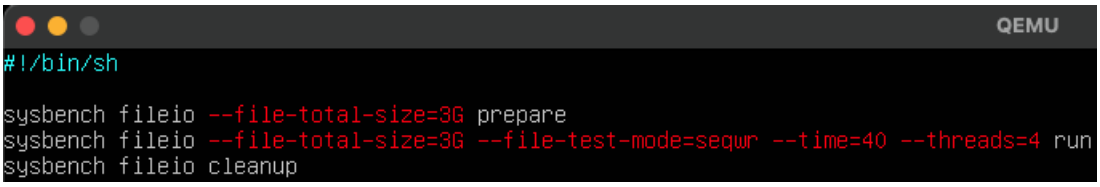


Figure 92 shell script

a. Results for executing five times in QEMU Ubuntu:

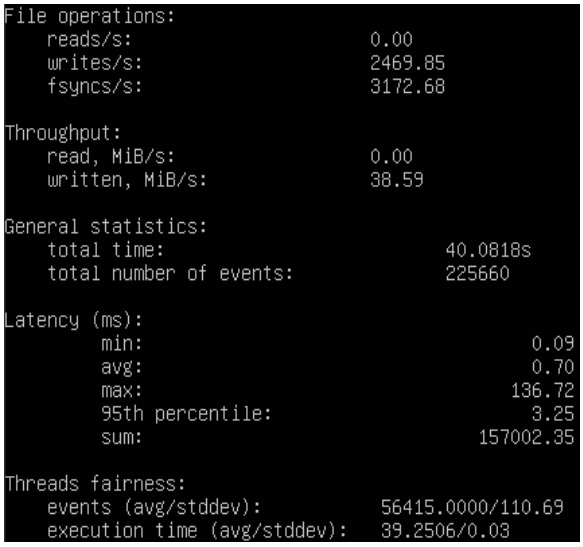


Figure 93 1st execution

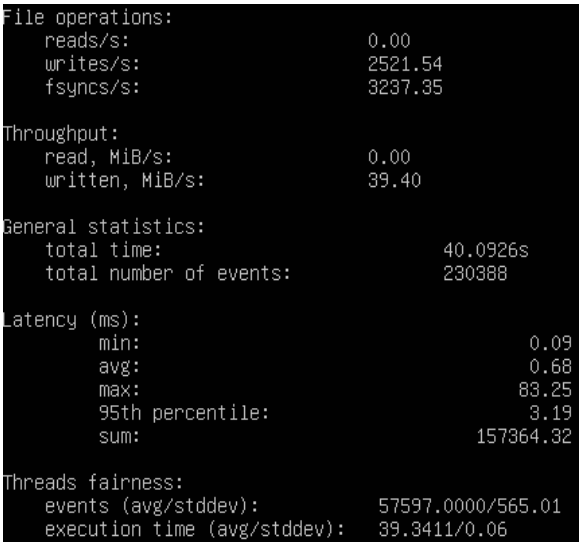


Figure 94 2nd execution

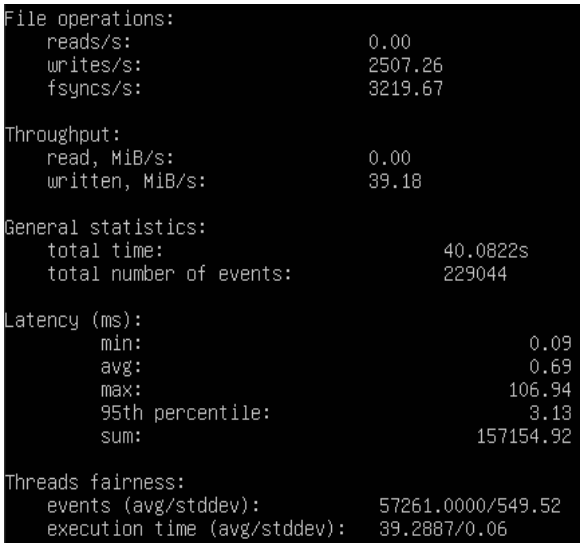


Figure 95 3rd execution

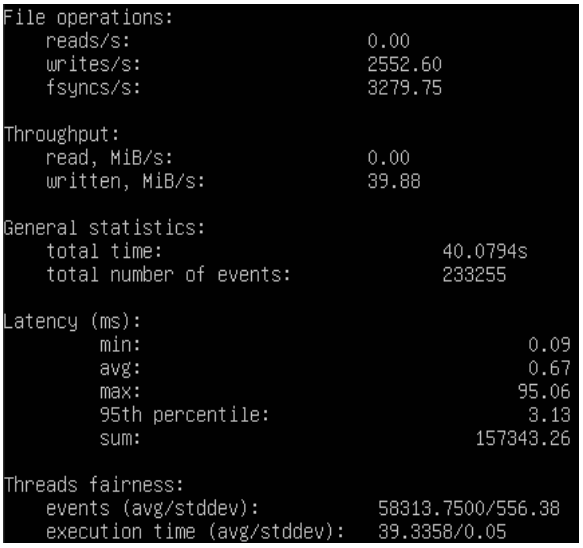


Figure 96 4th execution

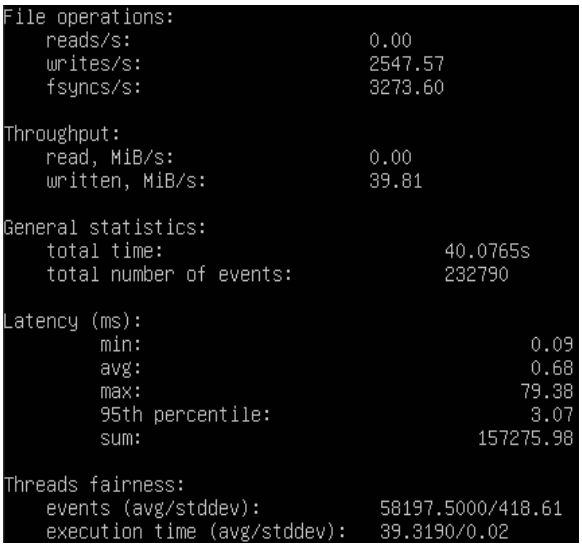


Figure 97 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th
Writes/sec	2469.85	2521.54	2507.26	2552.60	2547.57
Fsyncs/sec	3172.68	3237.35	3219.67	3279.75	3273.60
Throughput (w)	38.59	39.40	39.18	39.88	39.81
Total # of events	225660	230388	229044	233255	232790
Min latency (ms)	0.09	0.09	0.09	0.09	0.09
Avg latency (ms)	0.70	0.68	0.69	0.67	0.68
Max latency (ms)	136.72	83.25	106.94	95.06	79.38
Execution time (s)	39.2506	39.3411	39.2887	39.3358	39.3190

	Min	Max	Mean	Variance
Writes/sec	2469.85	2552.6	2519.764	900.457624

Fsyncs/sec	3172.68	3279.75	3236.61	1520.77516
Throughput (w)	38.59	39.88	39.372	0.219816
Total # of events	225660	233255	230227.4	7604130.24
Min latency (ms)	0.09	0.09	0.09	2.3111E-34
Avg latency (ms)	0.67	0.7	0.684	0.000104
Max latency (ms)	79.38	136.72	100.27	425.2616
Execution time (s)	39.2506	39.3411	39.30704	0.00113042

b. Results for executing five times in Docker Ubuntu:

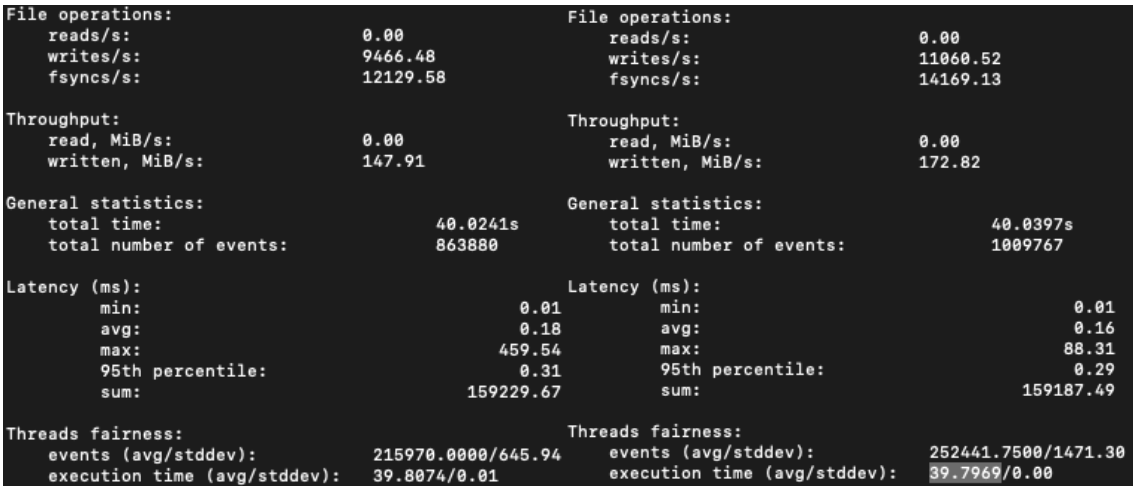


Figure 98 1st execution

Figure 99 2nd execution

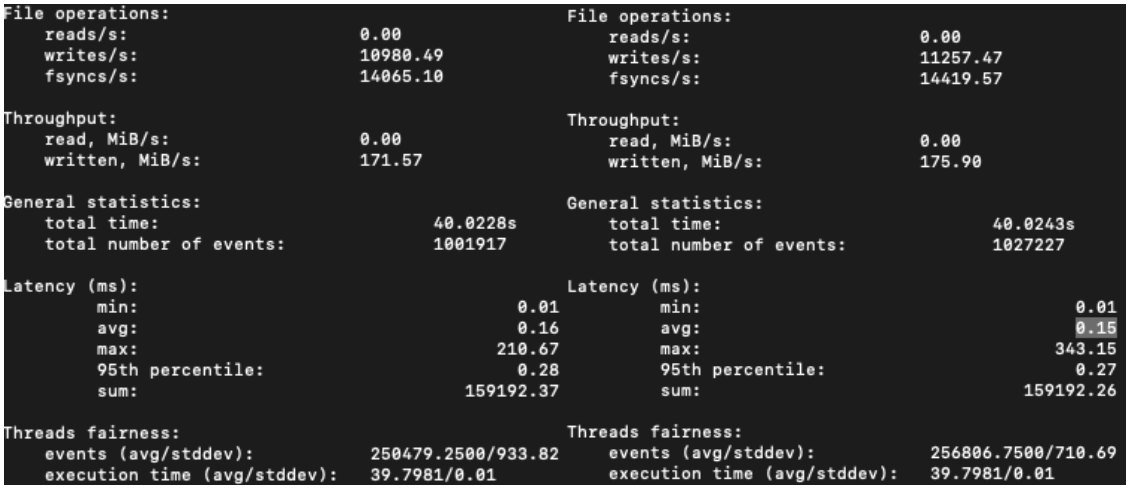


Figure 100 3rd execution

Figure 101 4th execution

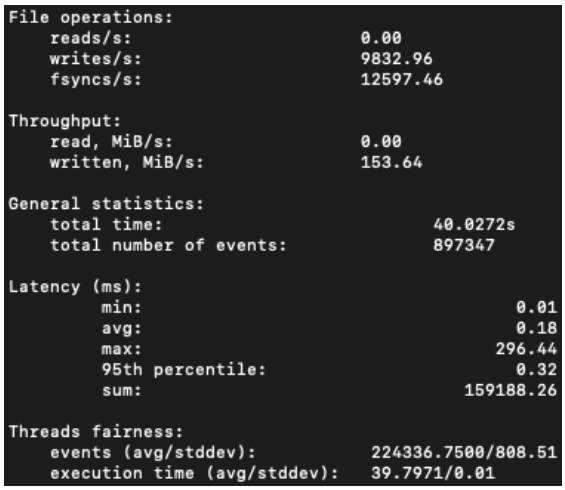


Figure 102 5th execution

The results show in table:

Time	1st	2nd	3rd	4th	5th
Writes/sec	9466.48	11060.52	10980.49	11257.47	9832.96
Fsyncs/sec	12129.58	14169.13	14065.10	14419.57	12597.46
Throughput (w)	147.91	172.82	171.57	175.90	153.64
Total # of events	863880	1009767	1001917	1027227	897347
Min latency (ms)	0.01	0.01	0.01	0.01	0.01
Avg latency (ms)	0.18	0.16	0.16	0.15	0.18
Max latency (ms)	459.54	88.31	210.67	343.15	296.44
Execution time (s)	39.8074	39.7969	39.7981	39.7981	39.7971

	Min	Max	Mean	Variance

Writes/sec	9466.48	11257.47	10519.584	526000.48
Fsyncs/sec	12129.58	14419.57	13476.168	860494.312
Throughput (w)	147.91	175.9	164.368	128.449576
Total # of events	863880	1027227	960027.6	4383541541
Min latency (ms)	0.01	0.01	0.01	0
Avg latency (ms)	0.15	0.18	0.166	0.000144
Max latency (ms)	88.31	459.54	279.622	15608.7597
Execution time (s)	39.7969	39.8074	39.79952	1.577E-05

8. Analysis

When doing CPU test, the CPU tends to do calculation much faster, and the execution time is low, when the maximum prime number is low. And when maximum prime number is high, the CPU shows a significant drop in calculation speed, and the execution time is very close to the running time. This is because when the prime number is high, the CPU will need to spend more time doing calculation, there will be less time spent on accessing memory, etc., which causes the execution time very close to the running time.

When using multiple threads to calculate, there will be a significant increase in latency.

When testing in Docker Ubuntu, the execution speed of CPU and file I/O speed tends to be larger than doing test in QEMU Ubuntu. Typically, three or more times larger.