

# Homework -1

By Pragya Kathpalia

- CPU Configurations :

Device name :DESKTOP-D5H0NVO

Processor: 1th Gen Intel(R) Core(TM) i7-11370H @ 3.30GHz 3.00 GHz

Installed RAM: 8.00 GB (7.74 GB usable)

Device ID: AA2B6EC1-2B9C-4460-8482-D9B3A872A4A7

Product ID: 00330-53645-01297-AAOEM

System type: 64-bit operating system, x64-based processor

Edition: Windows 10 Pro

Version: 21H1

OS build: 19043.1288

- Virtual Machine used On Ubuntu
- Installed Docker on Ubuntu by setting up Docker's repositories and install from them, for ease of installation and upgrade tasks. The following code was used :

To Set up the repository:

```
$sudo apt-get update
$sudo apt-get install \
    apt-transport-https \
    ca-certificates \
    curl \
    gnupg \
```

To Add Docker's official GPG key:

```
$curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg
--dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
```

To set up the **stable** repository:

```
$echo \"deb [arch=$(dpkg --print-architecture)
signed-by=/usr/share/keyrings/docker-archive-keyring.gpg]
https://download.docker.com/linux/ubuntu \
```

```
$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

To Verify that Docker Engine is installed correctly by running the `hello-world` image.

```
$ sudo docker run hello-world
```

- 

## Proof Of Experiment

### **Sysbench 4.1 :**

- Check for any current processes by executing “top -i” & make sure no other tasks are running in qemu and container
- Execute the bash script created to get the results
- Check the memory during each execution for user-level and kernel -level usage.

TEST 1 :sysbench --test=cpu --cpu-max-prime=20000 run

Ubuntu :

```
pragya@pragya-Virtual-Machine: ~  
Initializing worker threads...  
Threads started!  
CPU speed:  
  events per second: 1057.67  
General statistics:  
  total time: 10.0004s  
  total number of events: 10579  
Latency (ms):  
  min: 0.76  
  avg: 0.94  
  max: 6.39  
  95th percentile: 1.16  
  sum: 9990.75  
Threads fairness:  
  events (avg/stddev): 10579.0000/0.00  
  execution time (avg/stddev): 9.9907/0.00  
pragya@pragya-Virtual-Machine:~$
```

## Docker :

```
root@b9d528e1b278: /  
root@b9d528e1b278:/# sysbench --test=cpu --cpu-max-prime=20000 run  
sysbench 0.4.12: multi-threaded system evaluation benchmark  
Running the test with following options:  
Number of threads: 1  
Doing CPU performance benchmark  
Threads started!  
Done.  
Maximum prime number checked in CPU test: 20000  
Test execution summary:  
  total time: 10.3605s  
  total number of events: 10000  
  total time taken by event execution: 10.3541  
  per-request statistics:  
    min: 0.82ms  
    avg: 1.04ms  
    max: 7.70ms  
    approx. 95 percentile: 1.33ms  
Threads fairness:  
  events (avg/stddev): 10000.0000/0.00  
  execution time (avg/stddev): 10.3541/0.00  
root@b9d528e1b278:/#
```

TEST 2 :sysbench --test=cpu --cpu-max-prime=20000 run

Ubuntu :

```
pragya@pragya-Virtual-Machine: ~  
Initializing worker threads...  
Threads started!  
CPU speed:  
  events per second: 1057.67  
  
General statistics:  
  total time:          10.0004s  
  total number of events: 10579  
  
Latency (ms):  
  min:                0.76  
  avg:                0.94  
  max:                6.39  
  95th percentile:    1.16  
  sum:                9990.75  
  
Threads fairness:  
  events (avg/stddev): 10579.0000/0.00  
  execution time (avg/stddev): 9.9907/0.00  
pragya@pragya-Virtual-Machine:~$
```

Docker :

```
root@b9d528e1b278: /  
  
root@b9d528e1b278:/# sysbench --test=cpu --cpu-max-prime=20000 run  
sysbench 0.4.12: multi-threaded system evaluation benchmark  
  
Running the test with following options:  
Number of threads: 1  
  
Doing CPU performance benchmark  
  
Threads started!  
Done.  
  
Maximum prime number checked in CPU test: 20000  
  
Test execution summary:  
total time: 9.4387s  
total number of events: 10000  
total time taken by event execution: 9.4332  
per-request statistics:  
  min: 0.80ms  
  avg: 0.94ms  
  max: 7.81ms  
  approx. 95 percentile: 1.19ms  
  
Threads fairness:  
events (avg/stddev): 10000.0000/0.00  
execution time (avg/stddev): 9.4332/0.00  
root@b9d528e1b278:/#
```

TEST 3 : sysbench --test=cpu --cpu-max-prime=30000 run

Ubuntu :

```
pragya@pragya-Virtual-Machine: ~  
Initializing worker threads...  
Threads started!  
CPU speed:  
  events per second: 636.13  
General statistics:  
  total time: 10.0015s  
  total number of events: 6364  
Latency (ms):  
  min: 1.28  
  avg: 1.57  
  max: 3.79  
  95th percentile: 1.93  
  sum: 9995.26  
Threads fairness:  
  events (avg/stddev): 6364.0000/0.00  
  execution time (avg/stddev): 9.9953/0.00  
pragya@pragya-Virtual-Machine:~$
```

DOCKER :

```
root@b9d528e1b278: /  
  
root@b9d528e1b278:/# sysbench --test=cpu --cpu-max-prime=30000 run  
sysbench 0.4.12: multi-threaded system evaluation benchmark  
  
Running the test with following options:  
Number of threads: 1  
  
Doing CPU performance benchmark  
  
Threads started!  
Done.  
  
Maximum prime number checked in CPU test: 30000  
  
Test execution summary:  
total time: 15.6419s  
total number of events: 10000  
total time taken by event execution: 15.6364  
per-request statistics:  
  min: 1.31ms  
  avg: 1.56ms  
  max: 5.78ms  
  approx. 95 percentile: 1.90ms  
  
Threads fairness:  
events (avg/stddev): 10000.0000/0.00  
execution time (avg/stddev): 15.6364/0.00  
  
root@b9d528e1b278:/#
```

TEST 4 : sysbench --test=cpu --cpu-max-prime=35000 run  
UBUNTU :

```
pragya@pragya-Virtual-Machine: ~  
Initializing worker threads...  
Threads started!  
CPU speed:  
  events per second:   500.02  
  
General statistics:  
  total time:           10.0008s  
  total number of events: 5002  
  
Latency (ms):  
  min:                  1.57  
  avg:                  2.00  
  max:                  8.42  
  95th percentile:     2.43  
  sum:                  9995.02  
  
Threads fairness:  
  events (avg/stddev):   5002.0000/0.00  
  execution time (avg/stddev): 9.9950/0.00  
pragya@pragya-Virtual-Machine:~$
```

DOCKER :

```
root@b9d528e1b278: /  
root@b9d528e1b278:/# sysbench --test=cpu --cpu-max-prime=35000 run  
sysbench 0.4.12: multi-threaded system evaluation benchmark  
  
Running the test with following options:  
Number of threads: 1  
  
Doing CPU performance benchmark  
  
Threads started!  
Done.  
  
Maximum prime number checked in CPU test: 35000  
  
Test execution summary:  
  total time:           19.4386s  
  total number of events: 10000  
  total time taken by event execution: 19.4343  
  per-request statistics:  
    min:                 1.60ms  
    avg:                 1.94ms  
    max:                 5.96ms  
    approx. 95 percentile: 2.40ms  
  
Threads fairness:  
  events (avg/stddev):   10000.0000/0.00  
  execution time (avg/stddev): 19.4343/0.00  
root@b9d528e1b278:/#
```



TEST 5 :sysbench --test=cpu --cpu-max-prime=40000 run

UBUNTU :

```
pragya@pragya-Virtual-Machine: ~  
Initializing worker threads...  
Threads started!  
CPU speed:  
  events per second:   421.31  
  
General statistics:  
  total time:           10.0002s  
  total number of events: 4214  
  
Latency (ms):  
  min:                   1.90  
  avg:                   2.37  
  max:                   12.22  
  95th percentile:      2.86  
  sum:                   9994.98  
  
Threads fairness:  
  events (avg/stddev):   4214.0000/0.00  
  execution time (avg/stddev): 9.9950/0.00  
pragya@pragya-Virtual-Machine:~$
```

DOCKER :

```
root@b9d528e1b278: /  
root@b9d528e1b278:/# sysbench --test=cpu --cpu-max-prime=40000 run  
sysbench 0.4.12: multi-threaded system evaluation benchmark  
  
Running the test with following options:  
Number of threads: 1  
  
Doing CPU performance benchmark  
  
Threads started!  
Done.  
  
Maximum prime number checked in CPU test: 40000  
  
Test execution summary:  
total time: 23.1978s  
total number of events: 10000  
total time taken by event execution: 23.1947  
per-request statistics:  
min: 1.92ms  
avg: 2.32ms  
max: 11.12ms  
approx. 95 percentile: 2.88ms  
  
Threads fairness:  
events (avg/stddev): 10000.0000/0.00  
execution time (avg/stddev): 23.1947/0.00  
root@b9d528e1b278:/#
```

## Sysbench 4.5 :

- Check for any current processes by executing “top -i” & make sure no other tasks are running in qemu and container
- Execute the bash script created to get the results
- Check the memory during each execution for user-level and kernel -level usage.

### TEST 1 :

\$ sysbench --num-threads=16 --test=fileio --file-total-size=3G --file-test-mode=rndrw prepare

```
pragya@pragya-Virtual-Machine: ~  
--file-io-mode=STRING      file operations mode {sync,async,mmap} [sync]  
--file-async-backlog=N     number of asynchronous operations to queue per thread  
read [128]  
--file-extra-flags=[LIST,...] list of additional flags to use to open files {sync,dsync,direct} []  
--file-fsync-freq=N        do fsync() after this number of requests (0 - don't use fsync()) [100]  
--file-fsync-all[=on|off] do fsync() after each write operation [off]  
--file-fsync-end[=on|off] do fsync() at the end of test [on]  
--file-fsync-mode=STRING   which method to use for synchronization {fsync,fdatasync} [fsync]  
--file-merged-requests=N   merge at most this number of IO requests if possible (0 - don't merge) [0]  
--file-rw-ratio=N          reads/writes ratio for combined test [1.5]  
  
pragya@pragya-Virtual-Machine:~$ sysbench --num-threads=16 --test=fileio --file-total-size=4G --file-test-mode=rndrw cleanup  
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.  
WARNING: --num-threads is deprecated, use --threads instead  
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)  
  
Removing test files...  
pragya@pragya-Virtual-Machine:~$
```

\$ sysbench --num-threads=16 --test=fileio --file-total-size=3G --file-test-mode=rndrw run

```
pragya@pragya-Virtual-Machine: ~  
FATAL: Missing required argument: --file-test-mode  
  
fileio options:  
--file-num=N               number of files to create [128]  
--file-block-size=N        block size to use in all IO operations [16384]  
--file-total-size=SIZE     total size of files to create [2G]  
--file-test-mode=STRING    test mode {seqwr, seqrewr, seqrd, rndrd, rndwr, rndrw}  
--file-io-mode=STRING      file operations mode {sync,async,mmap} [sync]  
--file-async-backlog=N     number of asynchronous operations to queue per thread  
read [128]  
--file-extra-flags=[LIST,...] list of additional flags to use to open files {sync,dsync,direct} []  
--file-fsync-freq=N        do fsync() after this number of requests (0 - don't use fsync()) [100]  
--file-fsync-all[=on|off] do fsync() after each write operation [off]  
--file-fsync-end[=on|off] do fsync() at the end of test [on]  
--file-fsync-mode=STRING   which method to use for synchronization {fsync,fdatasync} [fsync]  
--file-merged-requests=N   merge at most this number of IO requests if possible (0 - don't merge) [0]  
--file-rw-ratio=N          reads/writes ratio for combined test [1.5]  
  
pragya@pragya-Virtual-Machine:~$
```

\$ sysbench --num-threads=16 --test=fileio --file-total-size=3G --file-test-mode=rndrw cleanup

```
pragya@pragya-Virtual-Machine: ~  
Creating file test_file.106  
Creating file test_file.107  
Creating file test_file.108  
Creating file test_file.109  
Creating file test_file.110  
Creating file test_file.111  
Creating file test_file.112  
Creating file test_file.113  
Creating file test_file.114  
Creating file test_file.115  
Creating file test_file.116  
Creating file test_file.117  
Creating file test_file.118  
Creating file test_file.119  
Creating file test_file.120  
Creating file test_file.121  
Creating file test_file.122  
Creating file test_file.123  
Creating file test_file.124  
Creating file test_file.125  
Creating file test_file.126  
Creating file test_file.127  
4294967296 bytes written in 5.49 seconds (746.48 MiB/sec).  
pragya@pragya-Virtual-Machine:~$
```

TEST 2 :

```
$ sysbench --num-threads=16 --test=fileio --file-total-size=3G --file-test-mode=rndrw prepare
```

```
pragya@pragya-Virtual-Machine: ~  
--file-io-mode=STRING      file operations mode {sync,async,mmap} [sync]  
--file-async-backlog=N     number of asynchronous operations to queue per thread [128]  
--file-extra-flags=[LIST,...] list of additional flags to use to open files {sync,dsync,direct} []  
--file-fsync-freq=N        do fsync() after this number of requests (0 - do not use fsync()) [100]  
--file-fsync-all[=on|off] do fsync() after each write operation [off]  
--file-fsync-end[=on|off]  do fsync() at the end of test [on]  
--file-fsync-mode=STRING   which method to use for synchronization {fsync,fdatasync} [fsync]  
--file-merged-requests=N   merge at most this number of IO requests if possible (0 - don't merge) [0]  
--file-rw-ratio=N          reads/writes ratio for combined test [1.5]  
  
pragya@pragya-Virtual-Machine:~$ sysbench --num-threads=32 --test=fileio --file-total-size=4G --file-test-mode=rndrw cleanup  
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.  
WARNING: --num-threads is deprecated, use --threads instead  
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)  
  
Removing test files...  
pragya@pragya-Virtual-Machine:~$
```

```
$ sysbench --num-threads=16 --test=fileio --file-total-size=3G --file-test-mode=rndrw run
```

```
pragya@pragya-Virtual-Machine: ~  
FATAL: Missing required argument: --file-test-mode  
  
fileio options:  
  --file-num=N                number of files to create [128]  
  --file-block-size=N         block size to use in all IO operations [16384]  
  --file-total-size=SIZE      total size of files to create [2G]  
  --file-test-mode=STRING     test mode {seqwr, seqrewr, seqrd, rndrd, rndwr,  
rndrw}  
  --file-io-mode=STRING       file operations mode {sync,async,mmap} [sync]  
  --file-async-backlog=N      number of asynchronous operations to queue per thread  
read [128]  
  --file-extra-flags=[LIST,...] list of additional flags to use to open files {sync,  
dsync,direct} []  
  --file-fsync-freq=N         do fsync() after this number of requests (0 - do  
n't use fsync()) [100]  
  --file-fsync-all[=on|off]  do fsync() after each write operation [off]  
  --file-fsync-end[=on|off]   do fsync() at the end of test [on]  
  --file-fsync-mode=STRING    which method to use for synchronization {fsync,  
fdatasync} [fsync]  
  --file-merged-requests=N    merge at most this number of IO requests if possible  
  --file-rw-ratio=N           reads/writes ratio for combined test [1.5]  
  
pragya@pragya-Virtual-Machine:~$
```

```
$ sysbench --num-threads=16 --test=fileio --file-total-size=3G --file-test-mode=rndrw cleanup
```

```
pragya@pragya-Virtual-Machine: ~  
Creating file test_file.106  
Creating file test_file.107  
Creating file test_file.108  
Creating file test_file.109  
Creating file test_file.110  
Creating file test_file.111  
Creating file test_file.112  
Creating file test_file.113  
Creating file test_file.114  
Creating file test_file.115  
Creating file test_file.116  
Creating file test_file.117  
Creating file test_file.118  
Creating file test_file.119  
Creating file test_file.120  
Creating file test_file.121  
Creating file test_file.122  
Creating file test_file.123  
Creating file test_file.124  
Creating file test_file.125  
Creating file test_file.126  
Creating file test_file.127  
4294967296 bytes written in 5.05 seconds (811.64 MiB/sec).  
pragya@pragya-Virtual-Machine:~$
```

TEST 3 :

\$ sysbench --num-threads=16 --test=fileio --file-total-size=4G --file-test-mode=rndrw prepare

```
pragya@pragya-Virtual-Machine: ~  
--file-io-mode=STRING      file operations mode {sync,async,mmap} [sync]  
--file-async-backlog=N     number of asynchronous operations to queue per thread  
read [128]  
--file-extra-flags=[LIST,...] list of additional flags to use to open files {sync,dsync,direct} []  
--file-fsync-freq=N        do fsync() after this number of requests (0 - don't use fsync()) [100]  
--file-fsync-all=[on|off]  do fsync() after each write operation [off]  
--file-fsync-end=[on|off]  do fsync() at the end of test [on]  
--file-fsync-mode=STRING   which method to use for synchronization {fsync,fdatasync} [fsync]  
--file-merged-requests=N   merge at most this number of IO requests if possible (0 - don't merge) [0]  
--file-rw-ratio=N          reads/writes ratio for combined test [1.5]  
  
pragya@pragya-Virtual-Machine:~$ sysbench --num-threads=16 --test=fileio --file-total-size=4G --file-test-mode=rndrw cleanup  
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.  
WARNING: --num-threads is deprecated, use --threads instead  
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)  
  
Removing test files...  
pragya@pragya-Virtual-Machine:~$
```

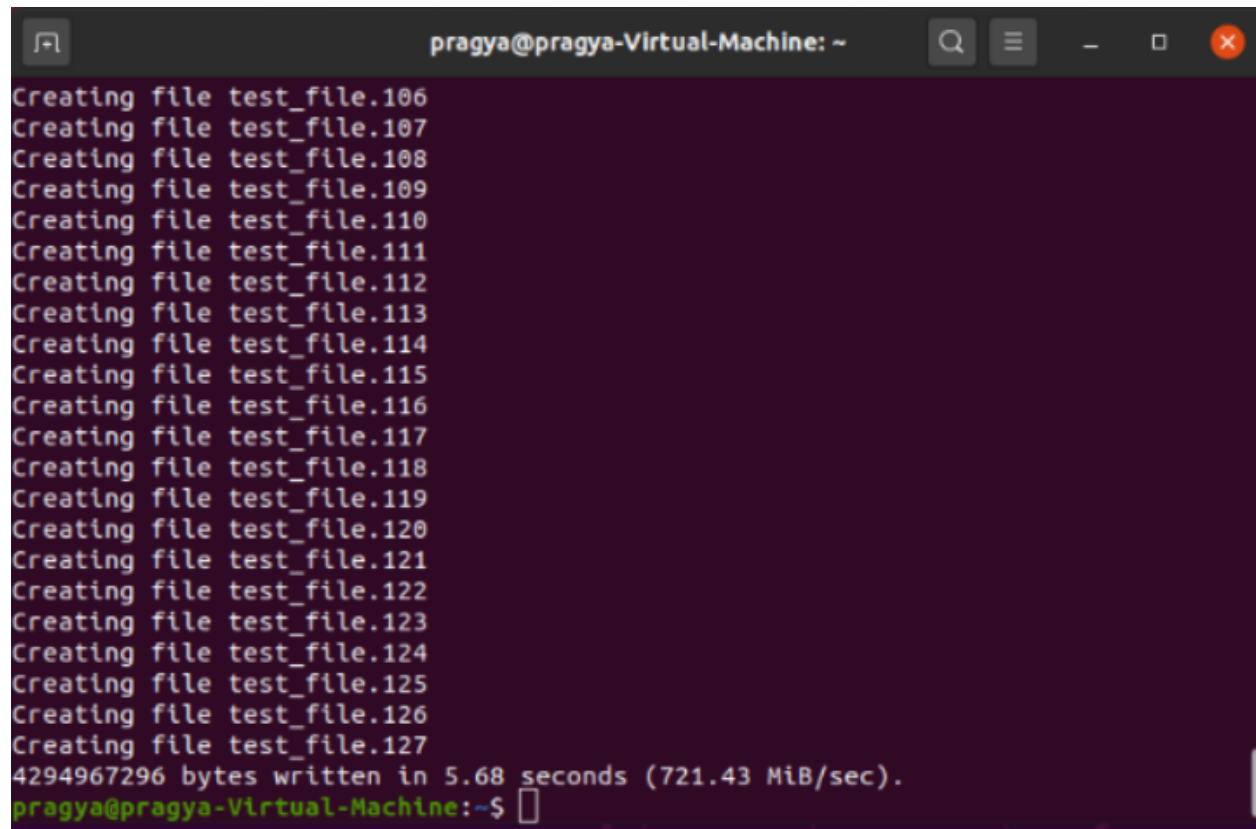


```
$ sysbench --num-threads=16 --test=fileio --file-total-size=4G --file-test-mode=rndrw run
```

```
pragya@pragya-Virtual-Machine: ~  
FATAL: Missing required argument: --file-test-mode  
  
fileio options:  
  --file-num=N                number of files to create [128]  
  --file-block-size=N         block size to use in all IO operations [16384]  
  --file-total-size=SIZE      total size of files to create [2G]  
  --file-test-mode=STRING     test mode {seqwr, seqrewr, seqrd, rndrd, rndwr, rndrw}  
  --file-io-mode=STRING       file operations mode {sync,async,mmap} [sync]  
  --file-async-backlog=N      number of asynchronous operations to queue per thread [128]  
  --file-extra-flags=[LIST,...] list of additional flags to use to open files {sync,dsync,direct} []  
  --file-fsync-freq=N         do fsync() after this number of requests (0 - don't use fsync()) [100]  
  --file-fsync-all[=on|off]  do fsync() after each write operation [off]  
  --file-fsync-end[=on|off]   do fsync() at the end of test [on]  
  --file-fsync-mode=STRING    which method to use for synchronization {fsync, fdatasync} [fsync]  
  --file-merged-requests=N    merge at most this number of IO requests if possible (0 - don't merge) [0]  
  --file-rw-ratio=N           reads/writes ratio for combined test [1.5]  
  
pragya@pragya-Virtual-Machine:~$
```



\$ sysbench --num-threads=16 --test=fileio --file-total-size=4G --file-test-mode=rndrw cleanup

A terminal window titled 'pragya@pragya-Virtual-Machine: ~' with standard window controls. The terminal output shows a list of 22 file creation messages from 'test\_file.106' to 'test\_file.127'. Below these, it reports '4294967296 bytes written in 5.68 seconds (721.43 MiB/sec)'. The prompt 'pragya@pragya-Virtual-Machine:~\$' is visible at the bottom.

```
Creating file test_file.106
Creating file test_file.107
Creating file test_file.108
Creating file test_file.109
Creating file test_file.110
Creating file test_file.111
Creating file test_file.112
Creating file test_file.113
Creating file test_file.114
Creating file test_file.115
Creating file test_file.116
Creating file test_file.117
Creating file test_file.118
Creating file test_file.119
Creating file test_file.120
Creating file test_file.121
Creating file test_file.122
Creating file test_file.123
Creating file test_file.124
Creating file test_file.125
Creating file test_file.126
Creating file test_file.127
4294967296 bytes written in 5.68 seconds (721.43 MiB/sec).
pragya@pragya-Virtual-Machine:~$
```

TEST 4 :

```
$ sysbench --num-threads=32 --test=fileio --file-total-size=3G --file-test-mode=rndrw prepare
```

```
pragya@pragya-Virtual-Machine: ~  
--file-io-mode=STRING      file operations mode {sync,async,mmap} [sync]  
--file-async-backlog=N     number of asynchronous operations to queue per thread  
read [128]  
--file-extra-flags=[LIST,...] list of additional flags to use to open files {sync,dsync,direct} []  
--file-fsync-freq=N        do fsync() after this number of requests (0 - don't use fsync()) [100]  
--file-fsync-all[=on|off] do fsync() after each write operation [off]  
--file-fsync-end[=on|off] do fsync() at the end of test [on]  
--file-fsync-mode=STRING   which method to use for synchronization {fsync,fdatasync} [fsync]  
--file-merged-requests=N   merge at most this number of IO requests if possible (0 - don't merge) [0]  
--file-rw-ratio=N          reads/writes ratio for combined test [1.5]  
  
pragya@pragya-Virtual-Machine:~$ sysbench --num-threads=32 --test=fileio --file-total-size=3G --file-test-mode=rndrw cleanup  
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.  
WARNING: --num-threads is deprecated, use --threads instead  
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)  
  
Removing test files...  
pragya@pragya-Virtual-Machine:~$
```

```
$ sysbench --num-threads=32 --test=fileio --file-total-size=3G --file-test-mode=rndrw run
```

```
pragya@pragya-Virtual-Machine: ~  
FATAL: Missing required argument: --file-test-mode  
  
fileio options:  
--file-num=N                number of files to create [128]  
--file-block-size=N         block size to use in all IO operations [16384]  
--file-total-size=SIZE      total size of files to create [2G]  
--file-test-mode=STRING     test mode {seqwr, seqrewr, seqrd, rndrd, rndwr,  
rndrw}  
--file-io-mode=STRING       file operations mode {sync,async,mmap} [sync]  
--file-async-backlog=N      number of asynchronous operations to queue per thread  
read [128]  
--file-extra-flags=[LIST,...] list of additional flags to use to open files {sync,  
dsync,direct} []  
--file-fsync-freq=N         do fsync() after this number of requests (0 - do  
n't use fsync()) [100]  
--file-fsync-all=[on|off]  do fsync() after each write operation [off]  
--file-fsync-end=[on|off]  do fsync() at the end of test [on]  
--file-fsync-mode=STRING    which method to use for synchronization {fsync,  
fdatasync} [fsync]  
--file-merged-requests=N    merge at most this number of IO requests if possible  
(0 - don't merge) [0]  
--file-rw-ratio=N           reads/writes ratio for combined test [1.5]  
  
pragya@pragya-Virtual-Machine:~$
```

```
$ sysbench --num-threads=32 --test=fileio --file-total-size=3G --file-test-mode=rndrw cleanup
```

```
pragya@pragya-Virtual-Machine: ~  
Creating file test_file.106  
Creating file test_file.107  
Creating file test_file.108  
Creating file test_file.109  
Creating file test_file.110  
Creating file test_file.111  
Creating file test_file.112  
Creating file test_file.113  
Creating file test_file.114  
Creating file test_file.115  
Creating file test_file.116  
Creating file test_file.117  
Creating file test_file.118  
Creating file test_file.119  
Creating file test_file.120  
Creating file test_file.121  
Creating file test_file.122  
Creating file test_file.123  
Creating file test_file.124  
Creating file test_file.125  
Creating file test_file.126  
Creating file test_file.127  
3221225472 bytes written in 4.06 seconds (755.78 MiB/sec).  
pragya@pragya-Virtual-Machine:~$
```

TEST 5 :

\$ sysbench --num-threads=32 --test=fileio --file-total-size=4G --file-test-mode=rndrw prepare

```
pragya@pragya-Virtual-Machine: ~  
--file-io-mode=STRING      file operations mode {sync,async,mmap} [sync]  
--file-async-backlog=N     number of asynchronous operations to queue per thread  
read [128]  
--file-extra-flags=[LIST,...] list of additional flags to use to open files {sync,dsync,direct} []  
--file-fsync-freq=N        do fsync() after this number of requests (0 - do not use fsync()) [100]  
--file-fsync-all[=on|off] do fsync() after each write operation [off]  
--file-fsync-end[=on|off] do fsync() at the end of test [on]  
--file-fsync-mode=STRING   which method to use for synchronization {fsync,fdatasync} [fsync]  
--file-merged-requests=N   merge at most this number of IO requests if possible (0 - don't merge) [0]  
--file-rw-ratio=N          reads/writes ratio for combined test [1.5]  
  
pragya@pragya-Virtual-Machine:~$ sysbench --num-threads=16 --test=fileio --file-total-size=3G--file-test-mode=rndrw cleanup  
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without any options.  
WARNING: --num-threads is deprecated, use --threads instead  
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)  
  
Removing test files...  
pragya@pragya-Virtual-Machine:~$
```

```
$ sysbench --num-threads=32 --test=fileio --file-total-size=4G --file-test-mode=rndrw run
```

```
pragya@pragya-Virtual-Machine: ~  
FATAL: Missing required argument: --file-test-mode  
  
fileio options:  
  --file-num=N          number of files to create [128]  
  --file-block-size=N   block size to use in all IO operations [16384]  
  --file-total-size=SIZE total size of files to create [2G]  
  --file-test-mode=STRING test mode {seqwr, seqrewr, seqrd, rndrd, rndwr, rndrw}  
  --file-io-mode=STRING  file operations mode {sync,async,mmap} [sync]  
  --file-async-backlog=N number of asynchronous operations to queue per thread [128]  
  --file-extra-flags=[LIST,...] list of additional flags to use to open files {sync,dsync,direct} []  
  --file-fsync-freq=N    do fsync() after this number of requests (0 - do not use fsync()) [100]  
  --file-fsync-all=[on|off] do fsync() after each write operation [off]  
  --file-fsync-end=[on|off] do fsync() at the end of test [on]  
  --file-fsync-mode=STRING which method to use for synchronization {fsync, fdatasync} [fsync]  
  --file-merged-requests=N merge at most this number of IO requests if possible (0 - don't merge) [0]  
  --file-rw-ratio=N      reads/writes ratio for combined test [1.5]  
  
pragya@pragya-Virtual-Machine:~$
```

```
$ sysbench --num-threads=32 --test=fileio --file-total-size=4G --file-test-mode=rndrw cleanup
```

```
pragya@pragya-Virtual-Machine: ~  
creating file test_file.106  
creating file test_file.107  
creating file test_file.108  
creating file test_file.109  
creating file test_file.110  
creating file test_file.111  
creating file test_file.112  
creating file test_file.113  
creating file test_file.114  
creating file test_file.115  
creating file test_file.116  
creating file test_file.117  
creating file test_file.118  
creating file test_file.119  
creating file test_file.120  
creating file test_file.121  
creating file test_file.122  
creating file test_file.123  
creating file test_file.124  
creating file test_file.125  
creating file test_file.126  
creating file test_file.127  
221225472 bytes written in 4.41 seconds (695.95 MiB/sec).  
pragya@pragya-Virtual-Machine:~$
```

