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
```

-

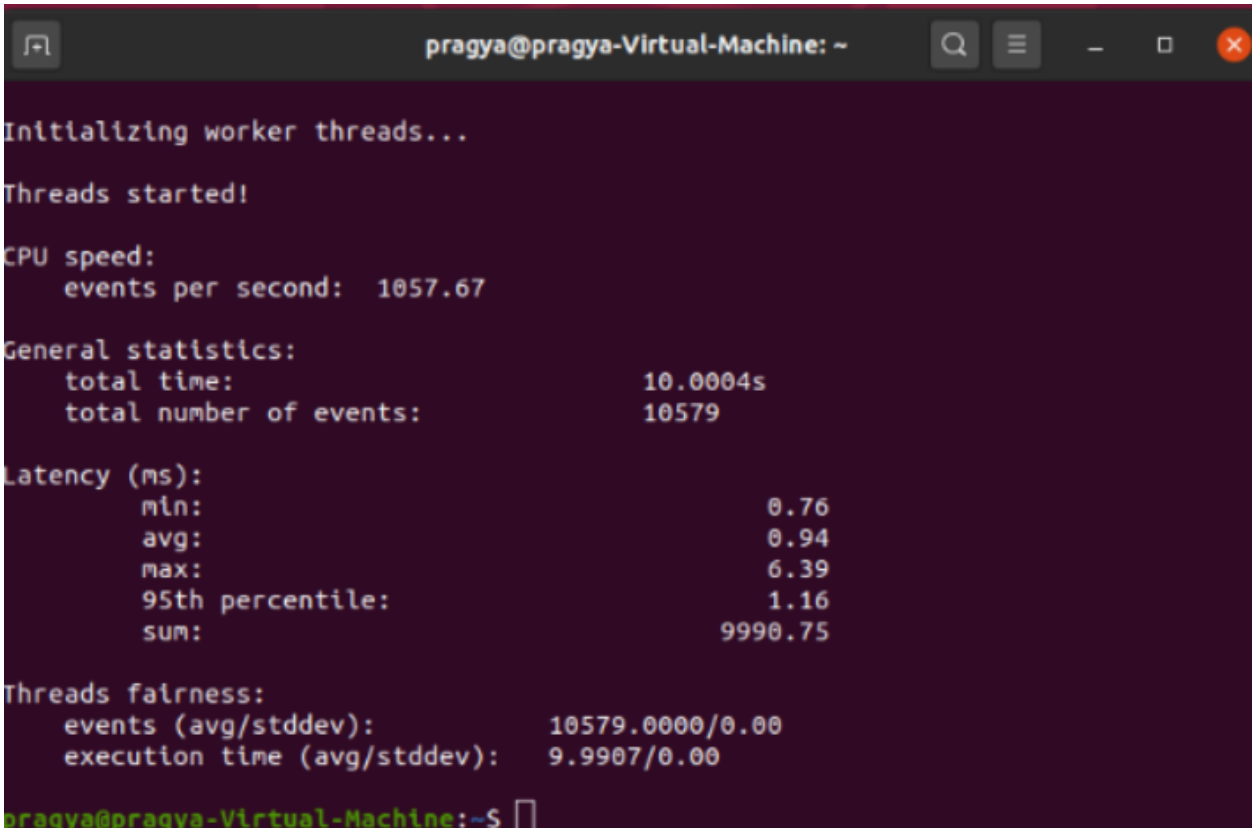
Proof Of Experiment

Sysbench 4.1 :

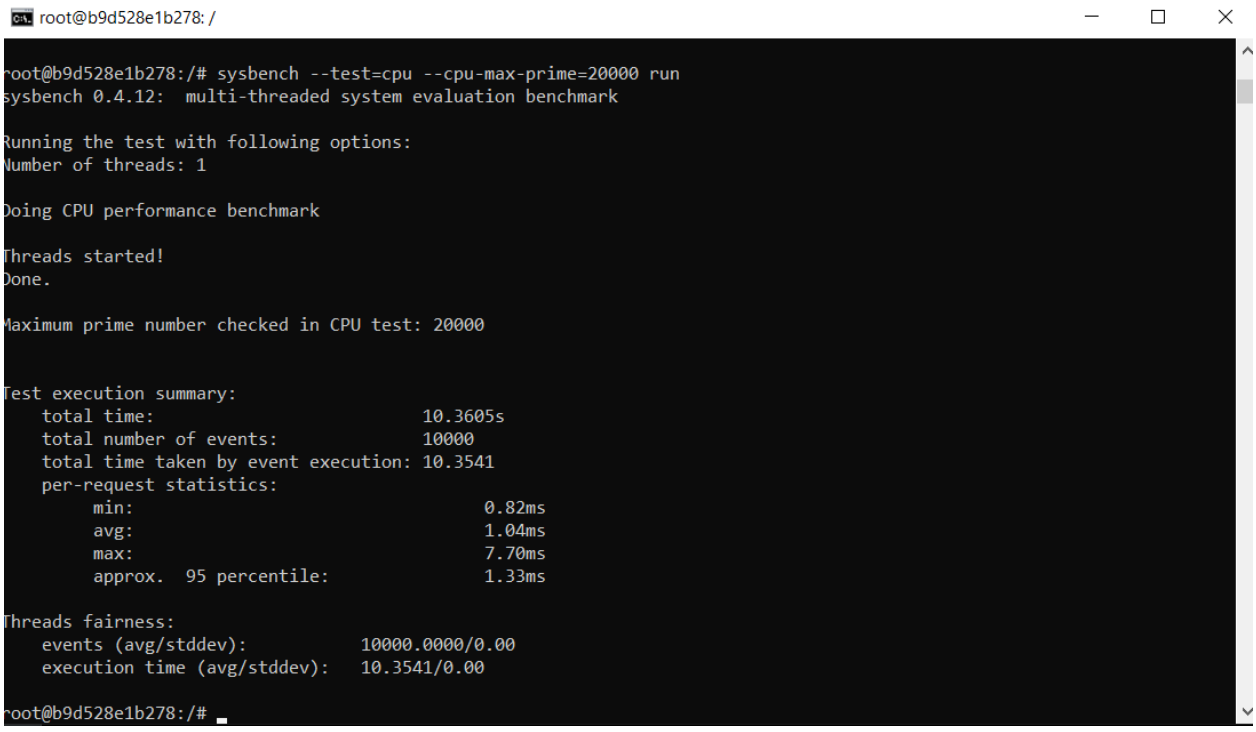
- Check for any current processes by executing “top -i” & make sure no other tasks are running in qemu and container
- Execute the bash script created to get the results
- Check the memory during each execution for user-level and kernel -level usage.

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

Ubuntu :

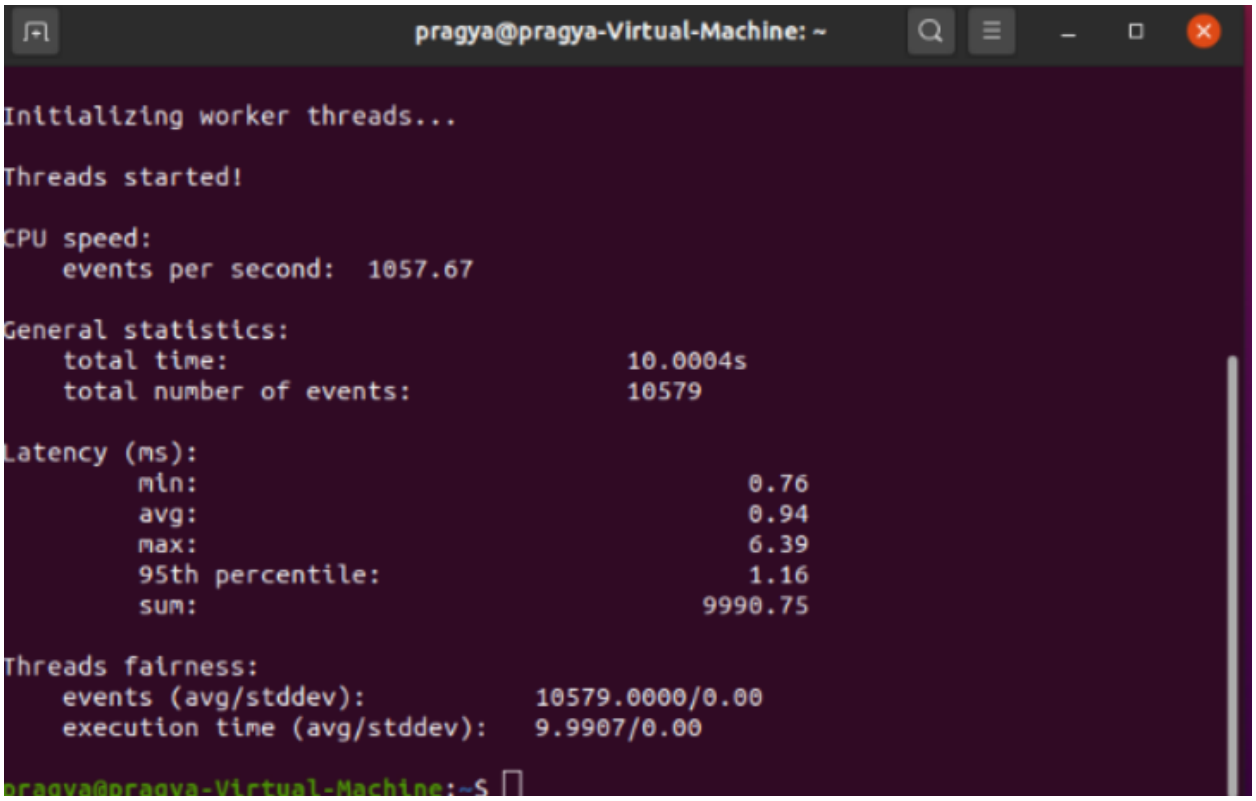

pragya@pragya-Virtual-Machine: ~
Initializing worker threads...
Threads started!
CPU speed:
events per second: 1057.67
General statistics:
total time: 10.0004s
total number of events: 10579
Latency (ms):
min: 0.76
avg: 0.94
max: 6.39
95th percentile: 1.16
sum: 9990.75
Threads fairness:
events (avg/stddev):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

Docker :

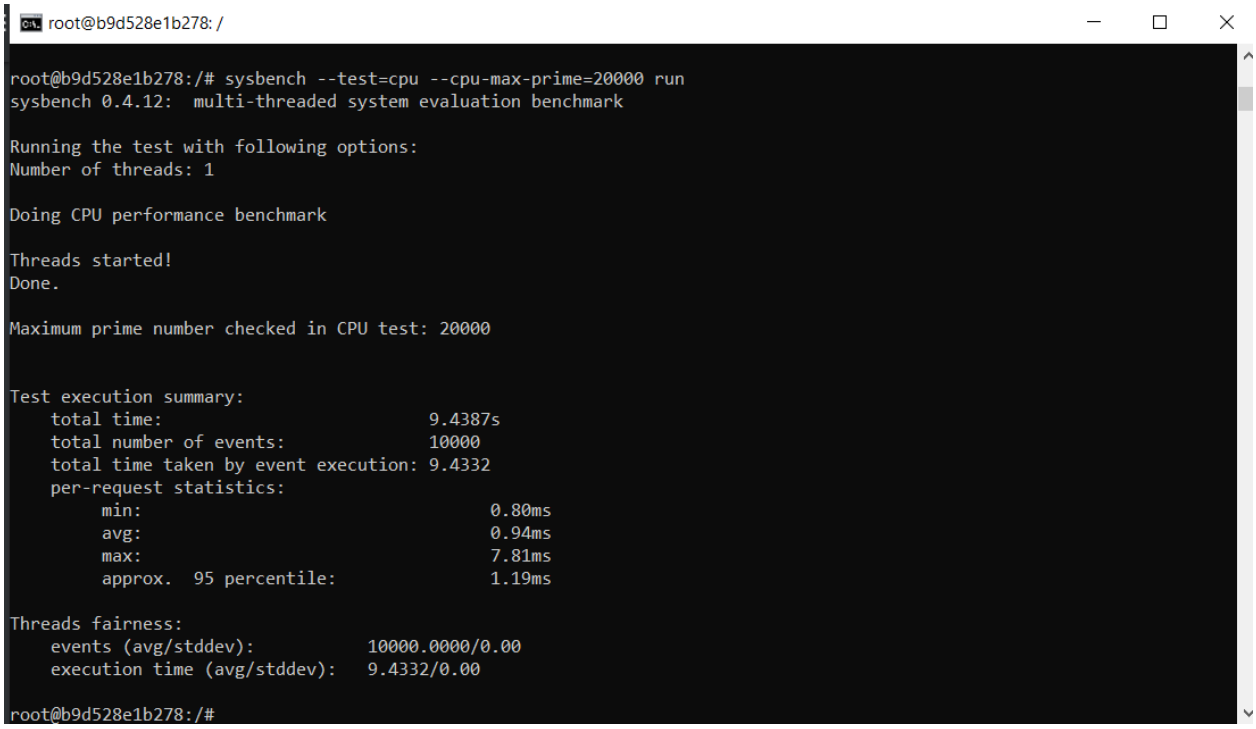

root@b9d528e1b278: /
root@b9d528e1b278:/# sysbench --test=cpu --cpu-max-prime=20000 run
sysbench 0.4.12: multi-threaded system evaluation benchmark
Running the test with following options:
Number of threads: 1
Doing CPU performance benchmark
Threads started!
Done.
Maximum prime number checked in CPU 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 :

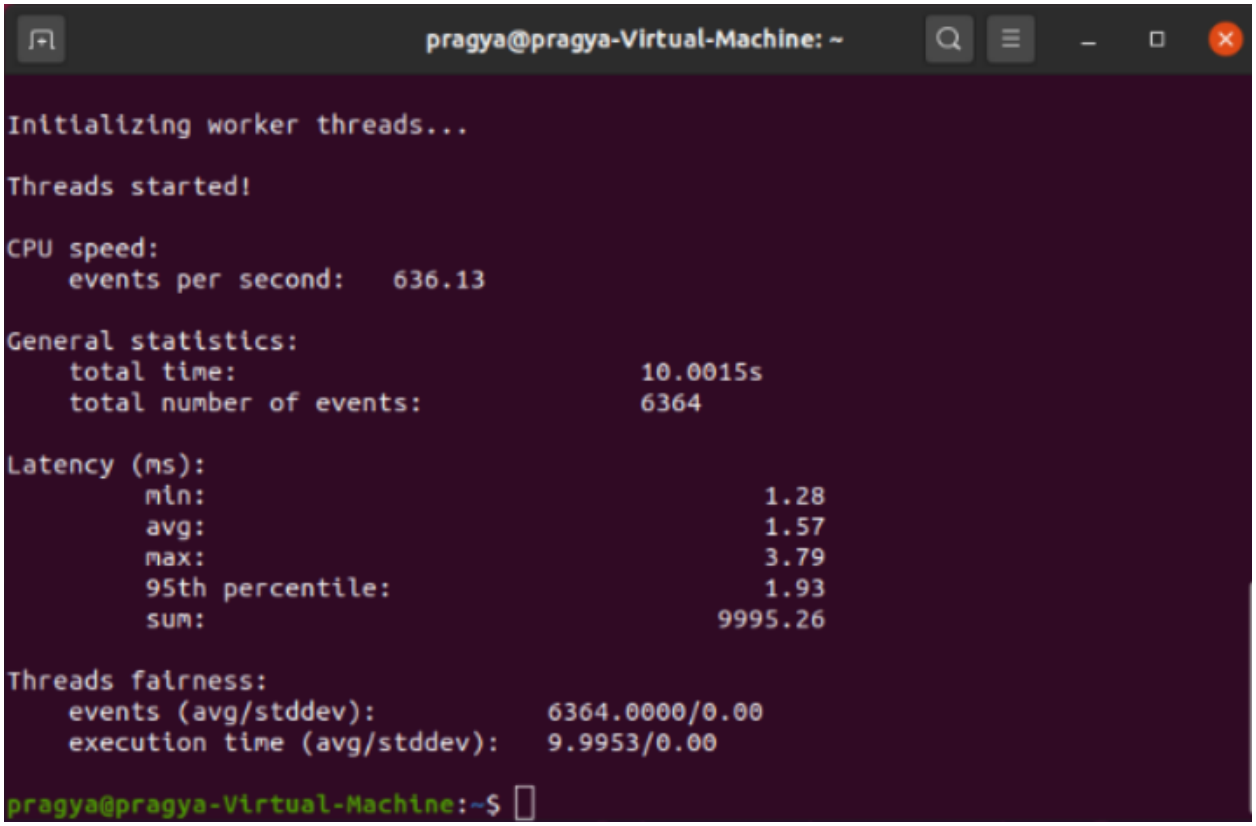


Docker :

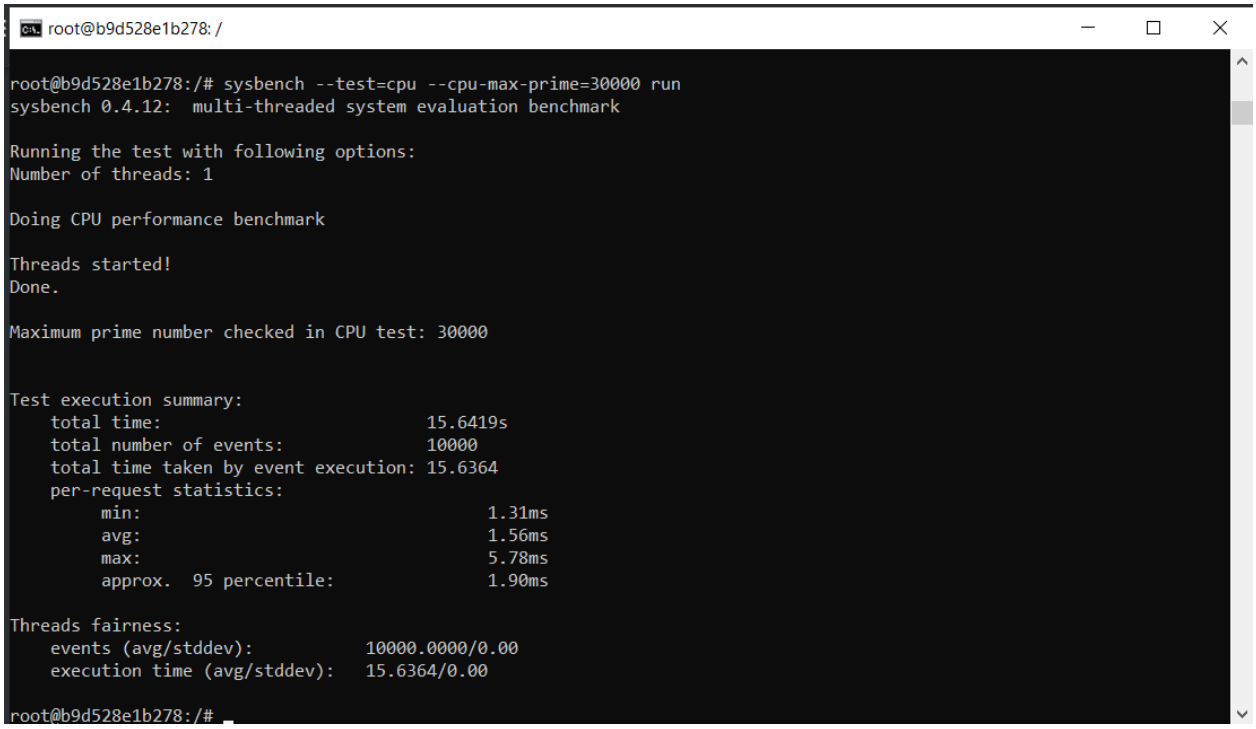


TEST 3 : sysbench --test=cpu --cpu-max-prime=30000 run

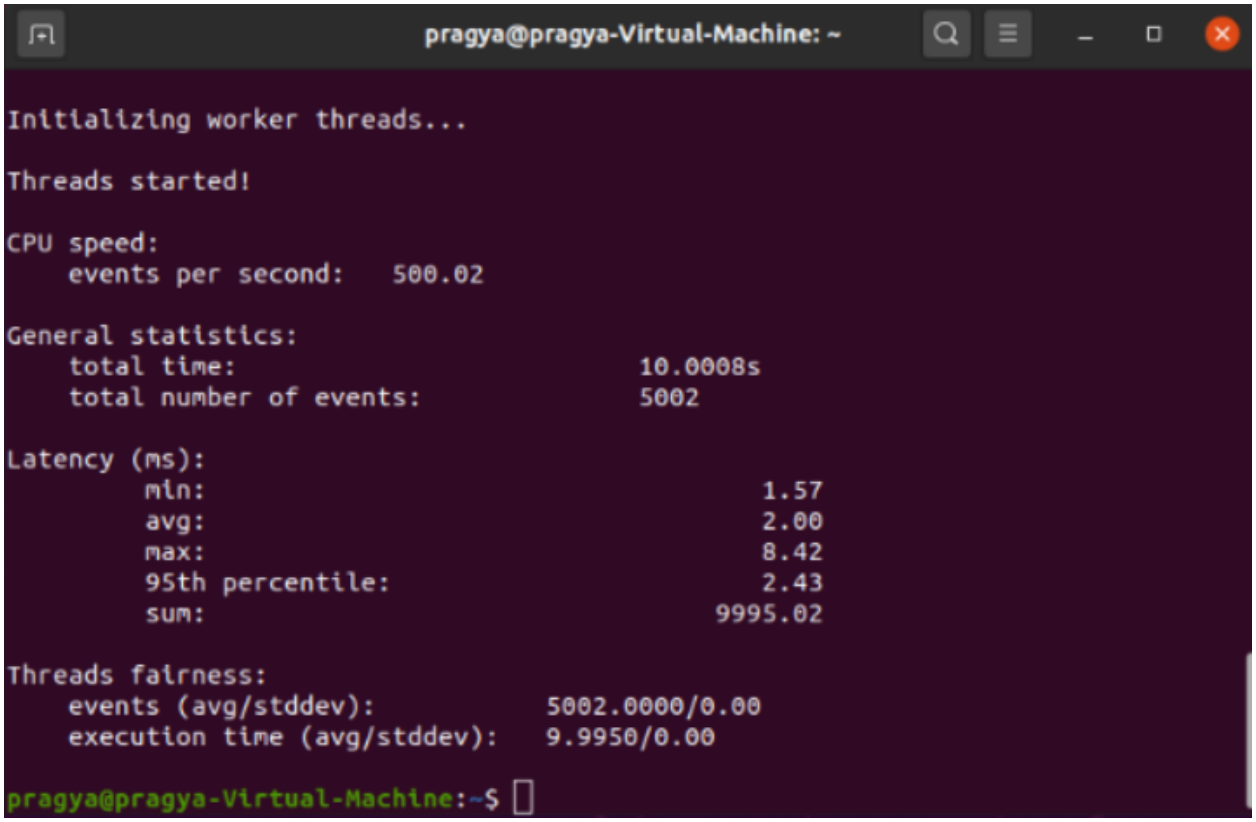
Ubuntu :


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:~$

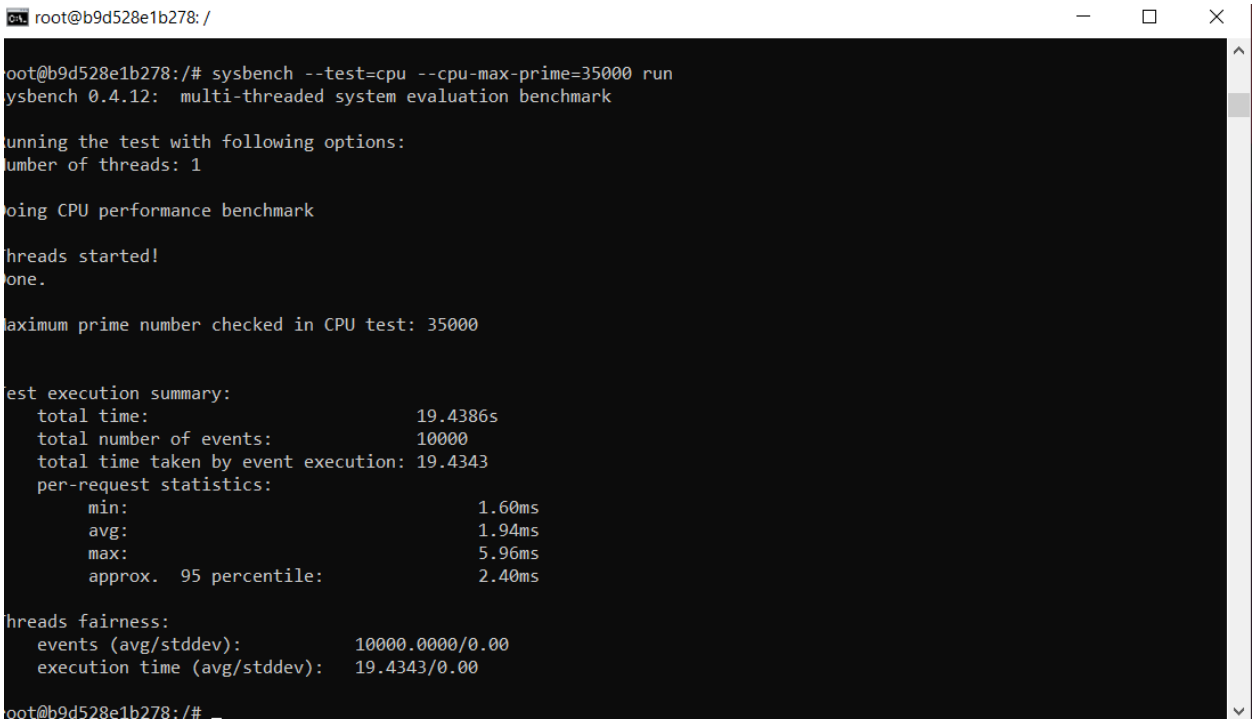
DOCKER :



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

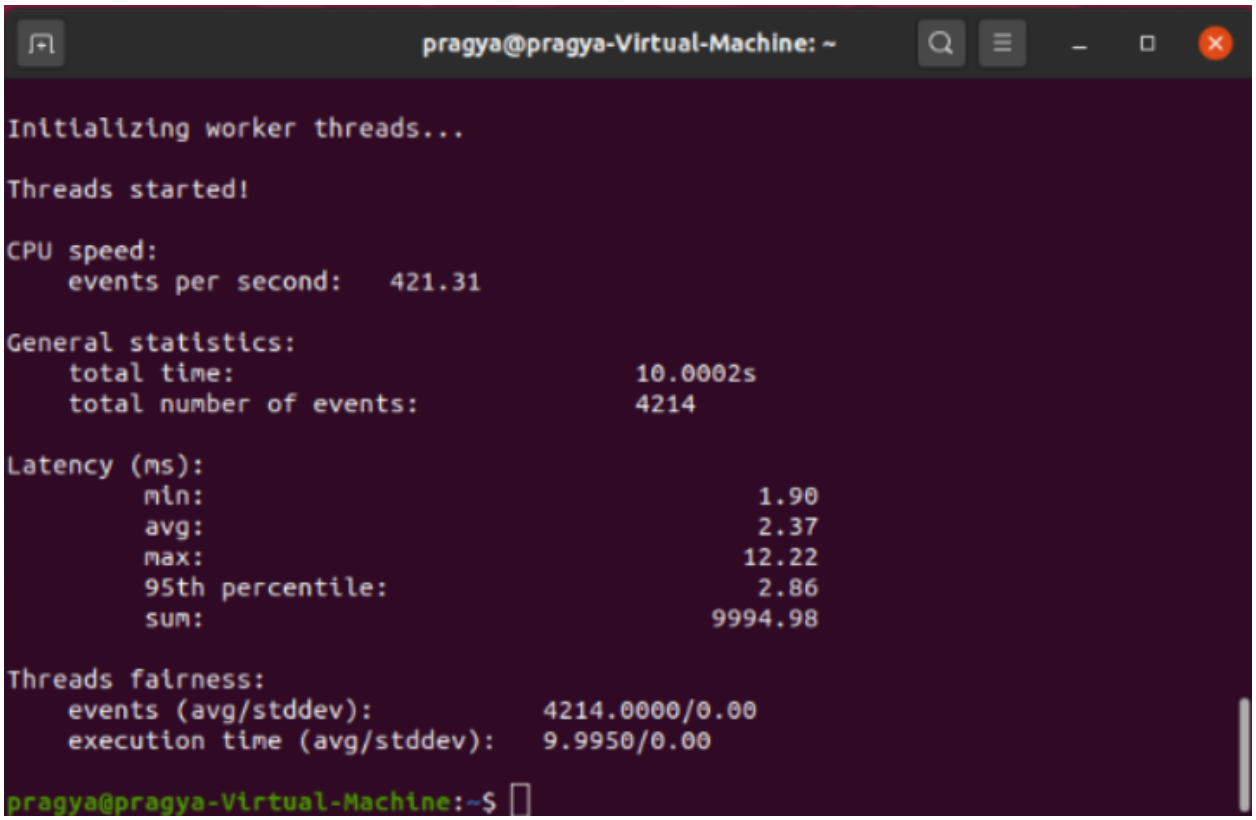

pragya@pragya-Virtual-Machine: ~
Initializing worker threads...
Threads started!
CPU speed:
events per second: 500.02
General statistics:
total time: 10.0008s
total number of events: 5002
Latency (ms):
min: 1.57
avg: 2.00
max: 8.42
95th percentile: 2.43
sum: 9995.02
Threads fairness:
events (avg/stddev):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

DOCKER :

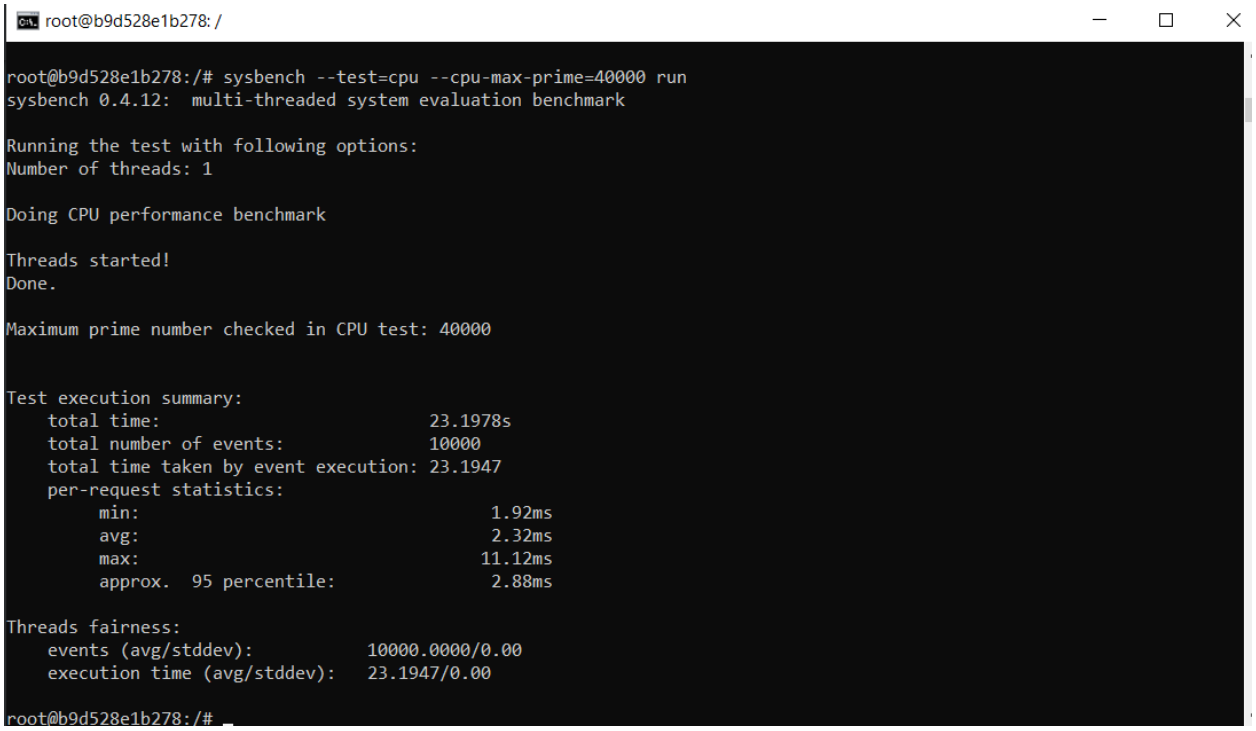

root@b9d528e1b278: /
oot@b9d528e1b278:/# sysbench --test=cpu --cpu-max-prime=35000 run
ysbench 0.4.12: multi-threaded system evaluation benchmark
unning the test with following options:
umber of threads: 1
oing CPU performance benchmark
hreads started!
one.
aximum prime number checked in CPU test: 35000
est 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
hreads fairness:
events (avg/stddev): 10000.0000/0.00
execution time (avg/stddev): 19.4343/0.00
oot@b9d528e1b278:/#

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

UBUNTU :



DOCKER :

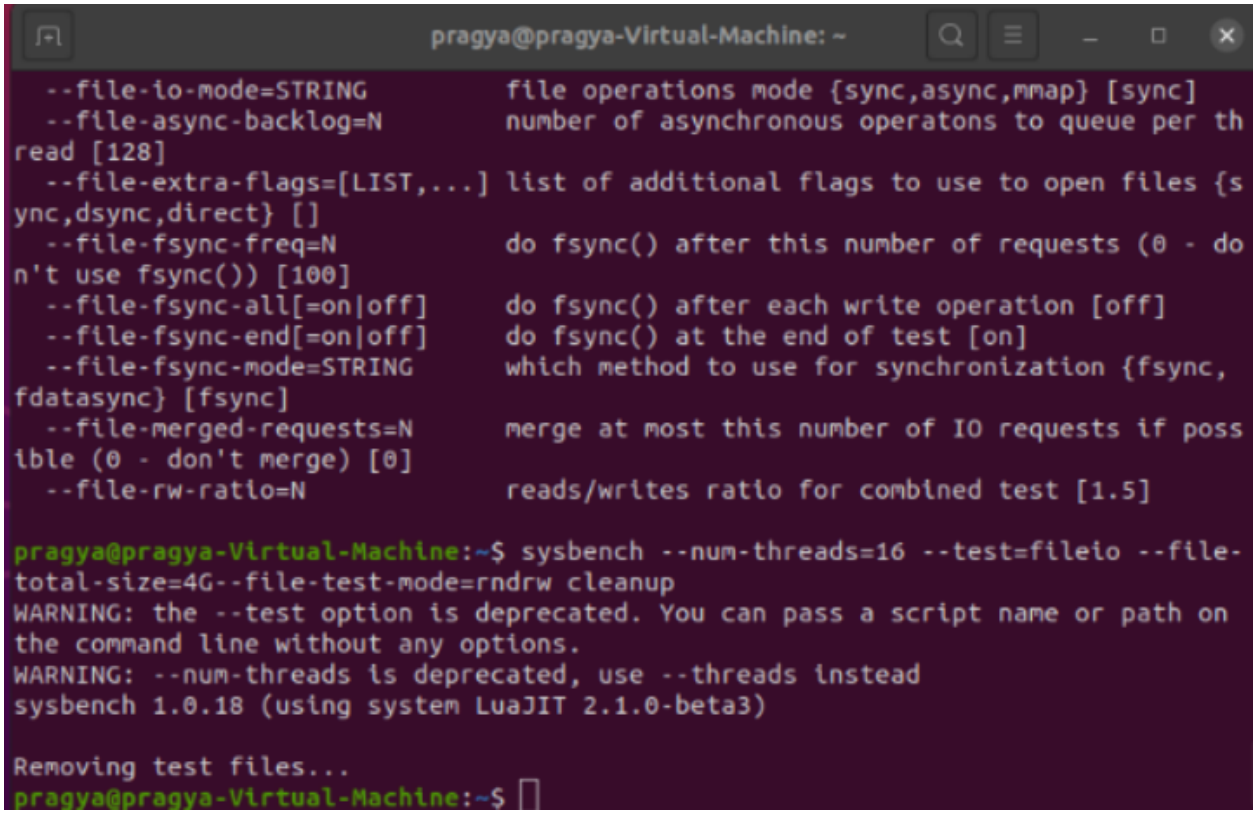


Sysbench 4.5 :

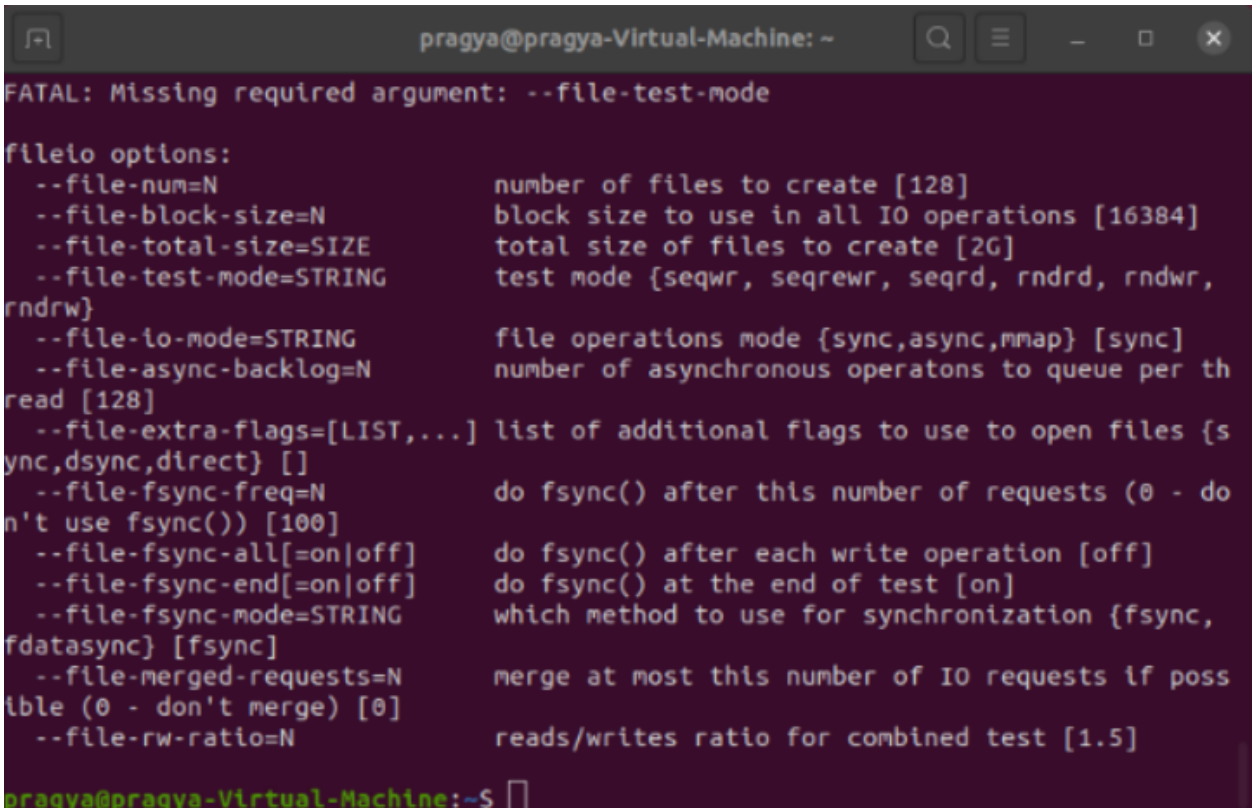
- Check for any current processes by executing “top -i” & make sure no other tasks are running in qemu and container
- Execute the bash script created to get the results
- Check the memory during each execution for user-level and kernel -level usage.

TEST 1 :

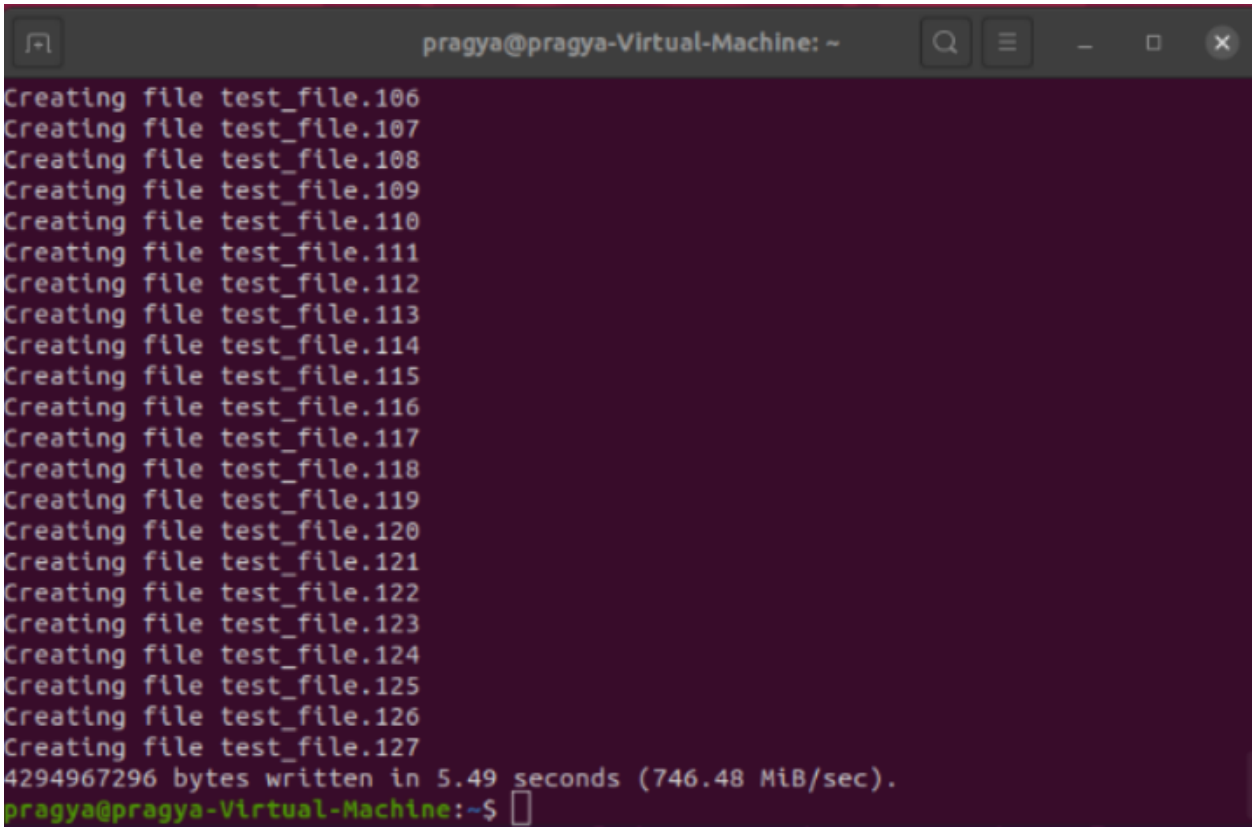
$ sysbench --num-threads=16 --test=fileio --file-total-size=3G --file-test-mode=rndrw prepare



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

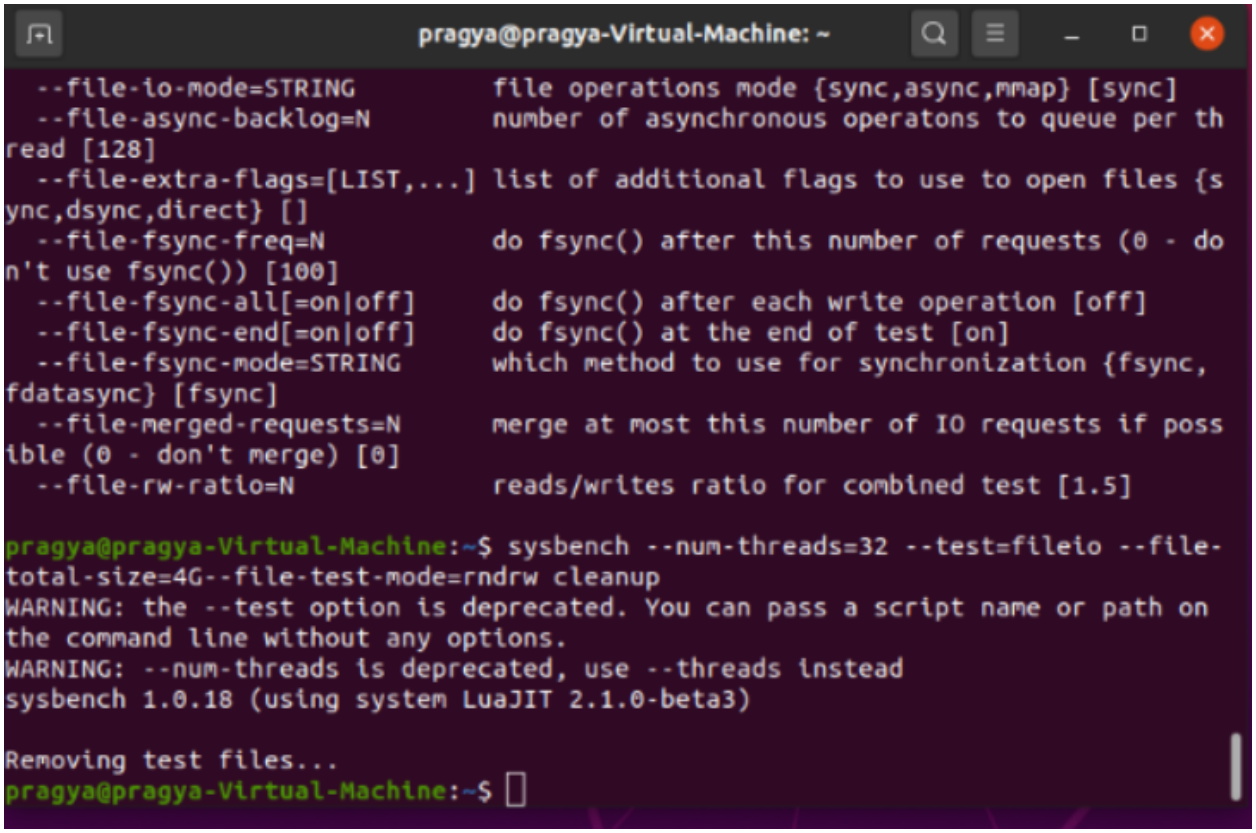


$ sysbench --num-threads=16 --test=fileio --file-total-size=3G --file-test-mode=rndrw cleanup

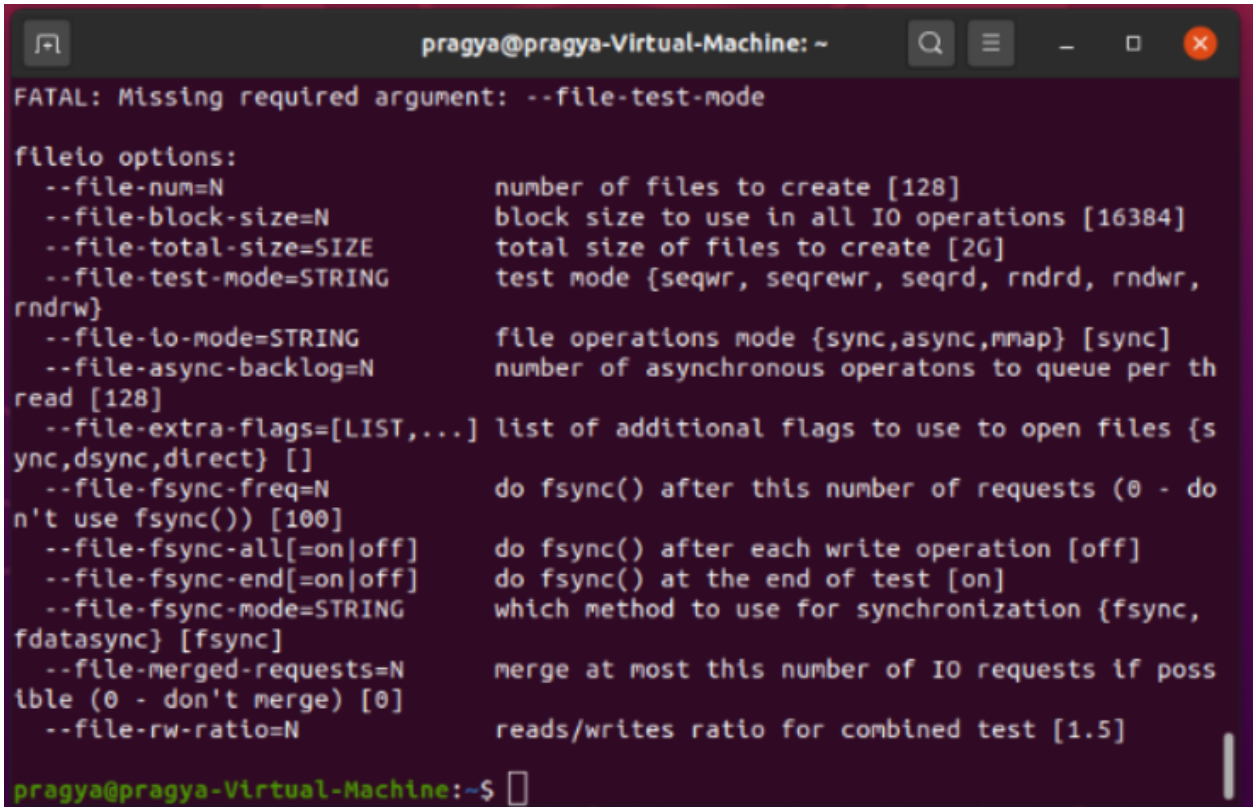


TEST 2 :

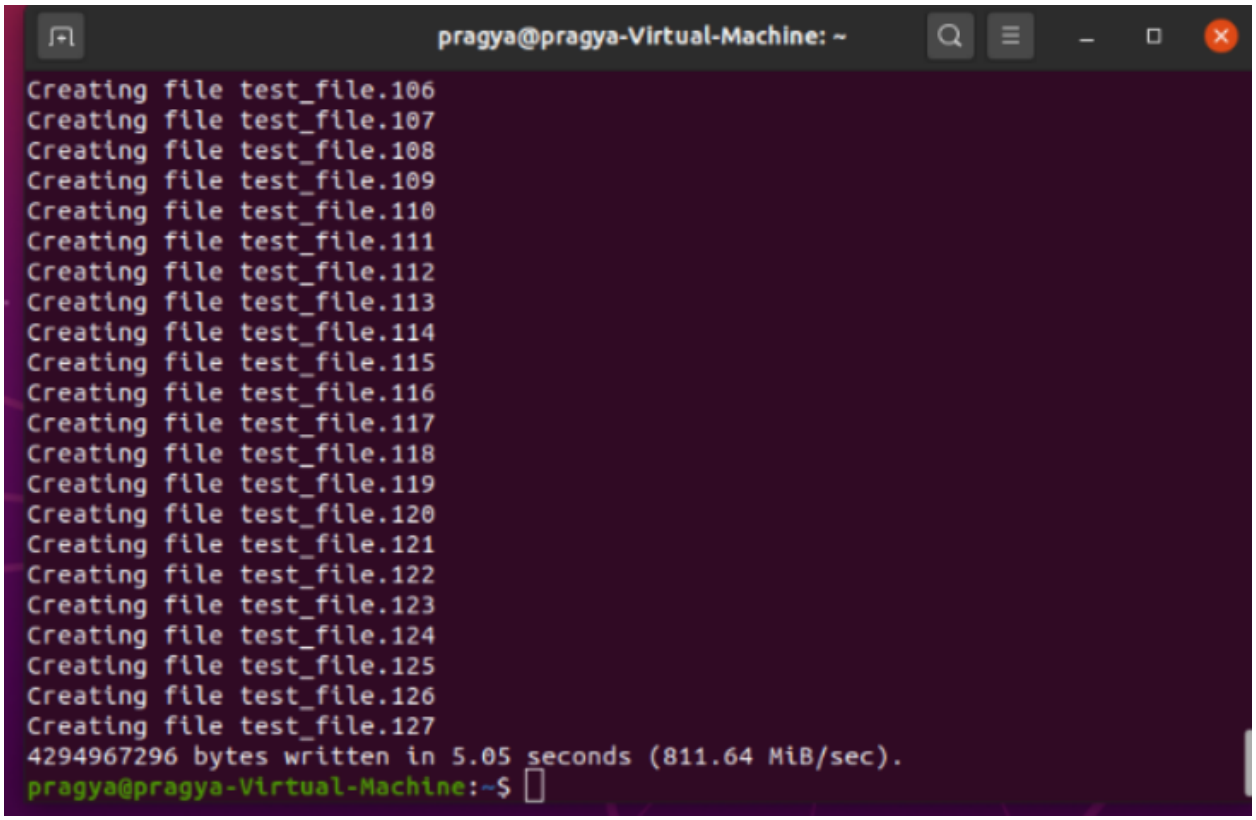
$ sysbench --num-threads=16 --test=fileio --file-total-size=3G --file-test-mode=rndrw prepare



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

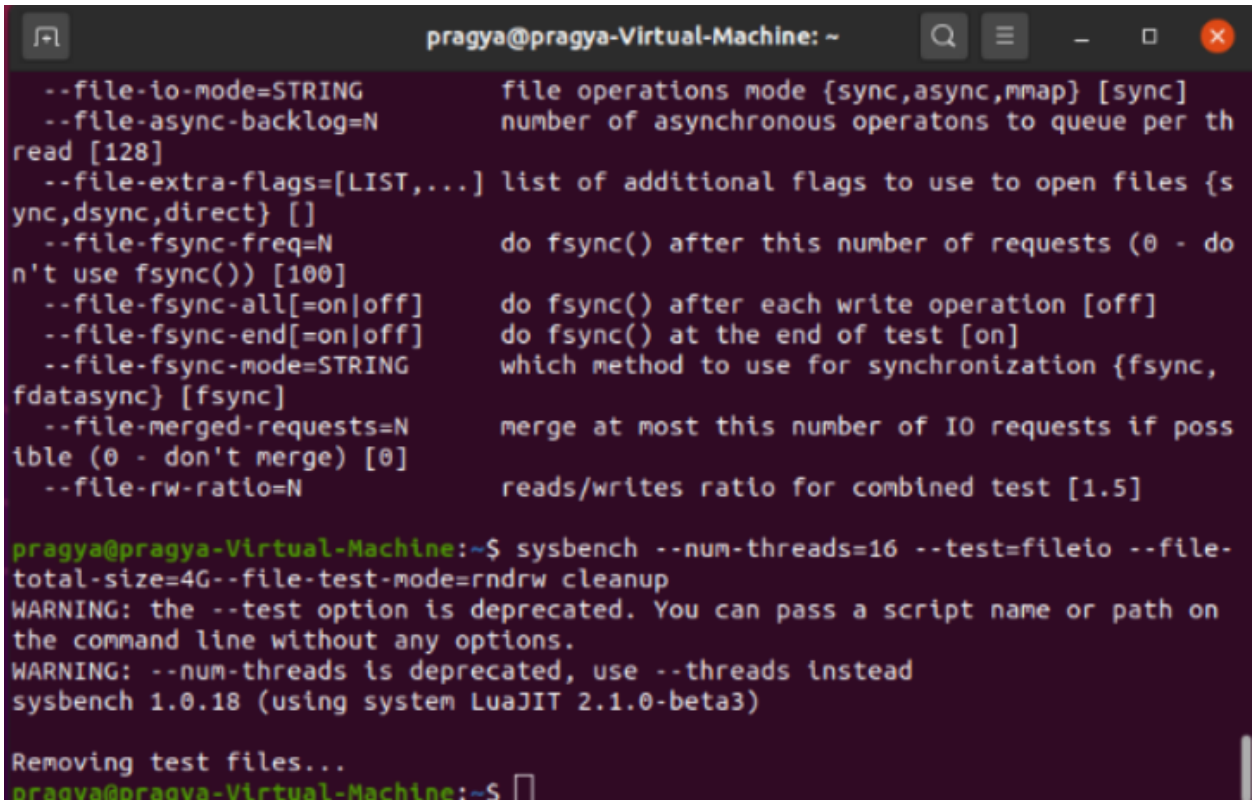


$ sysbench --num-threads=16 --test=fileio --file-total-size=3G --file-test-mode=rndrw cleanup

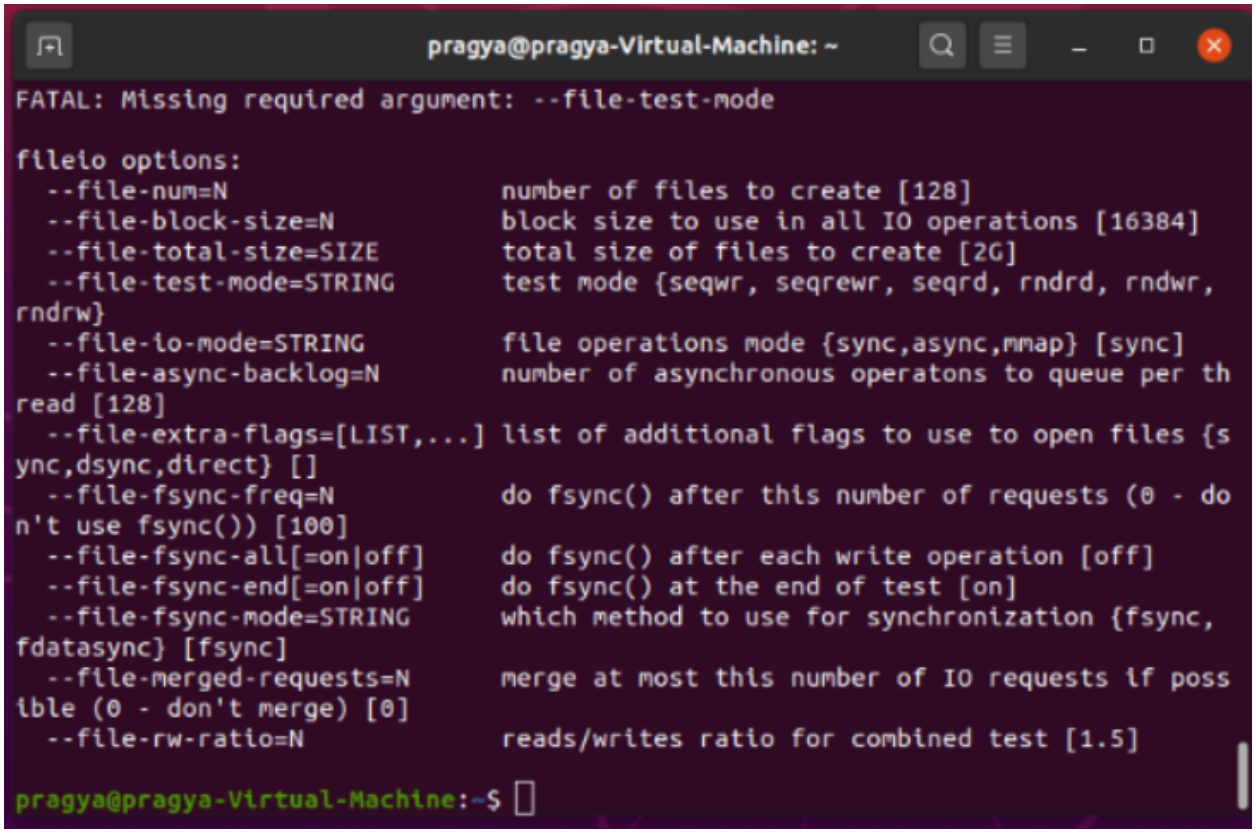


TEST 3 :

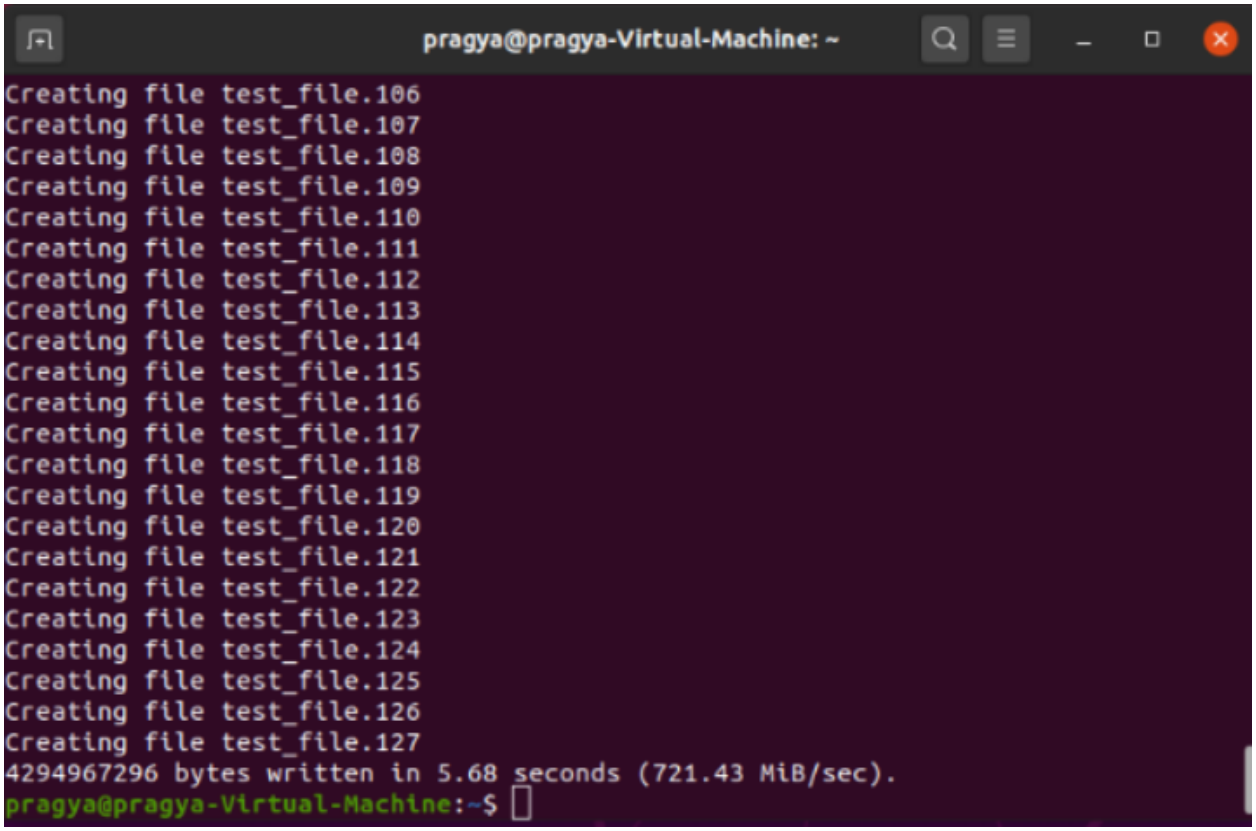
$ sysbench --num-threads=16 --test=fileio --file-total-size=4G --file-test-mode=rndrw prepare



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

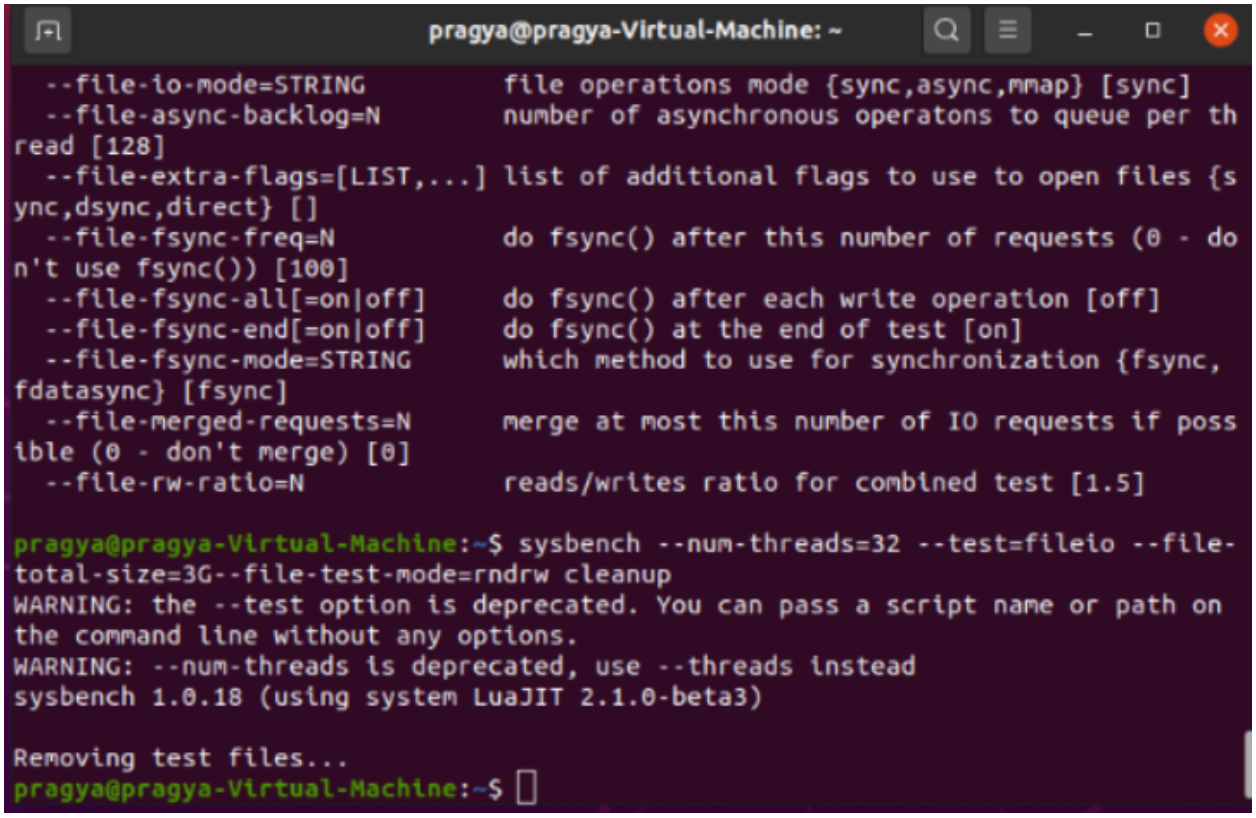


$ sysbench --num-threads=16 --test=fileio --file-total-size=4G --file-test-mode=rndrw cleanup

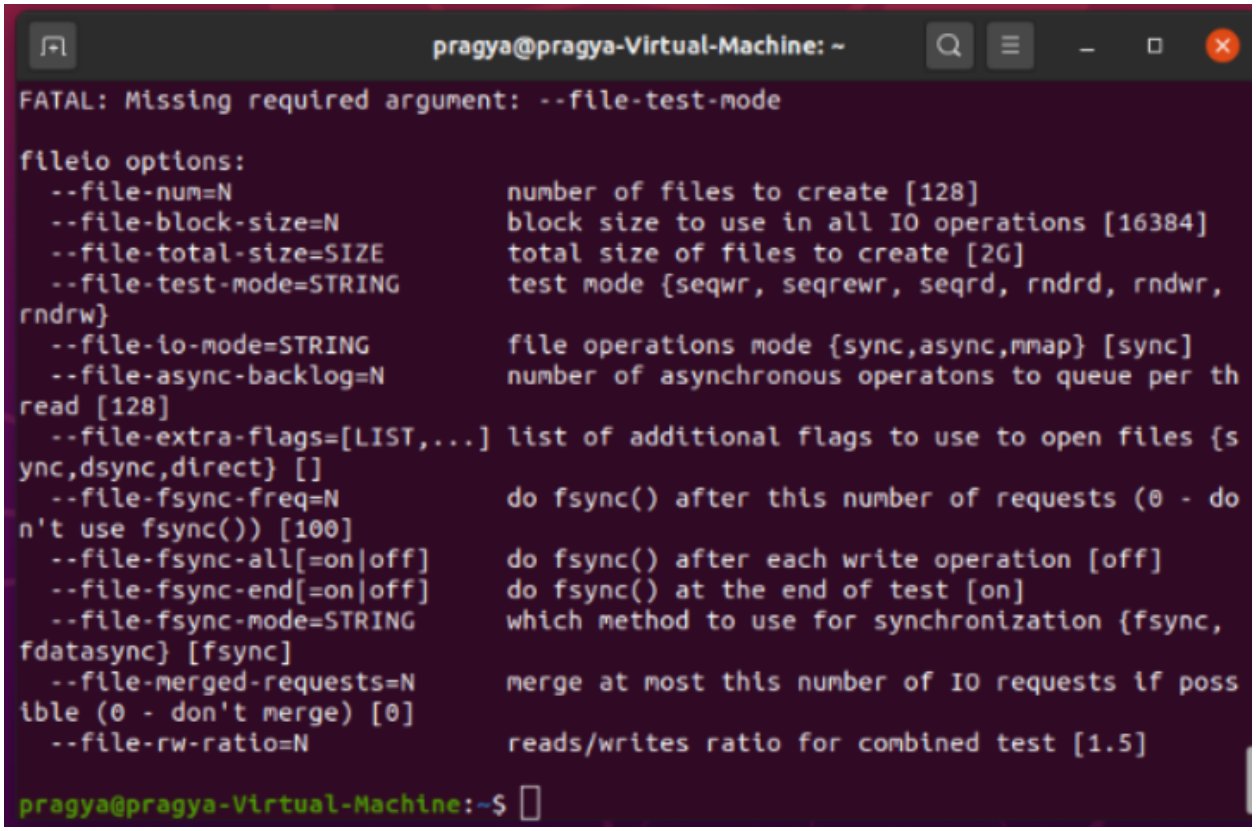


TEST 4 :

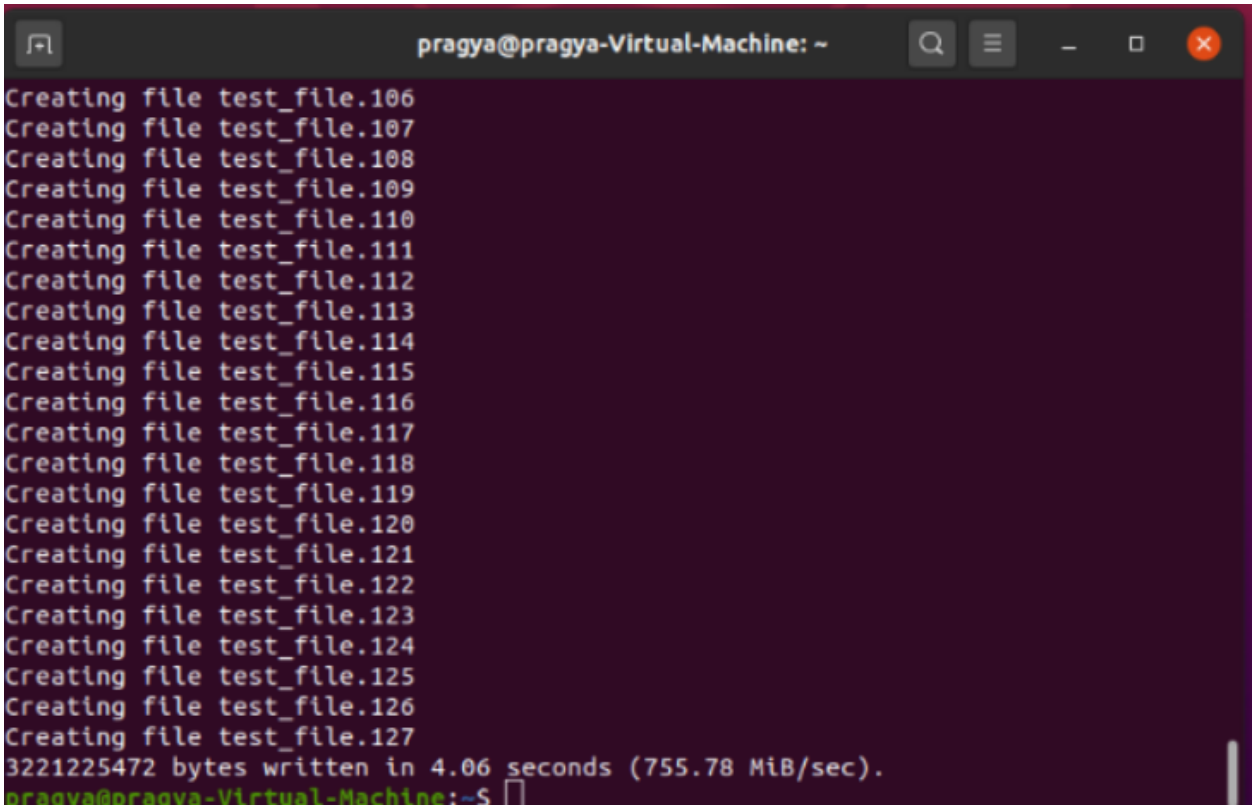
$ sysbench --num-threads=32 --test=fileio --file-total-size=3G --file-test-mode=rndrw prepare



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

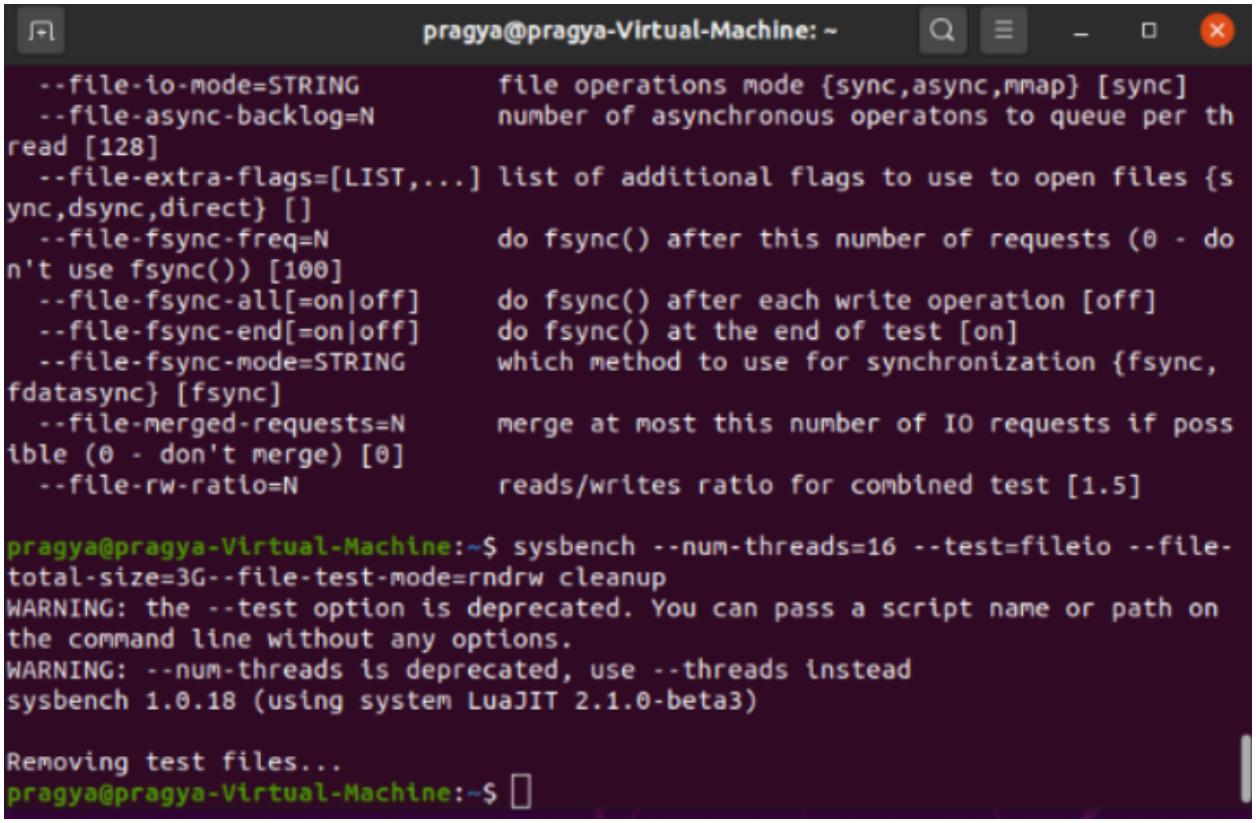


$ sysbench --num-threads=32 --test=fileio --file-total-size=3G --file-test-mode=rndrw cleanup

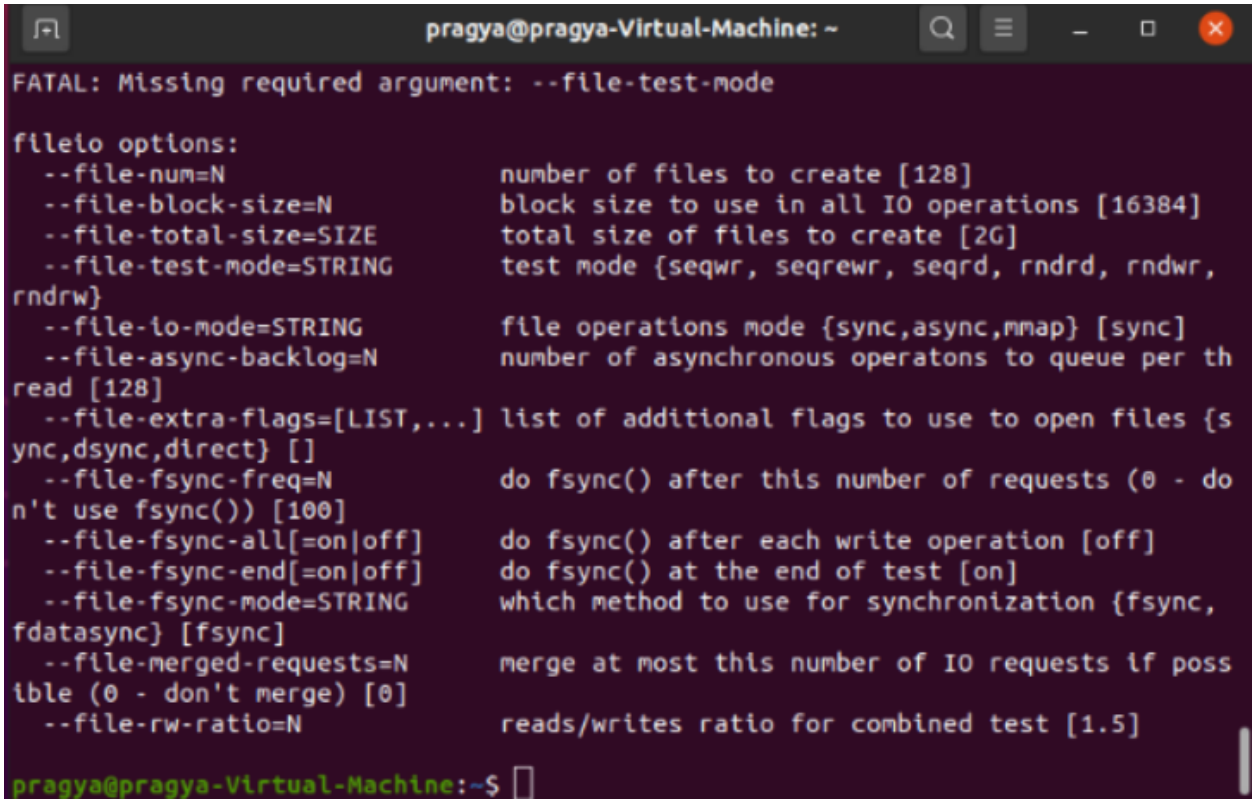


TEST 5 :

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



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



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