CSEN 241 HW1

System vs OS Virtualization

Ruihao Wei

ID: W1648525

**Computer Architecture:**

This experiment ran on a computer with architecture: x86\_64

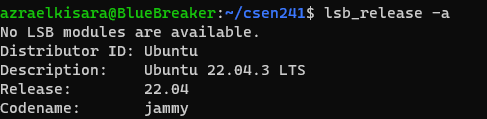
**Operating System:**

This experiment ran on Windows 11 with [WSL2](https://learn.microsoft.com/en-us/windows/wsl/install)

The Windows Subsytem for Linux (WSL) let developers install a Linux distribution directly on Windows, without the overhead of traditional virtual machine or dualboot setup.

**Installed Linux distribution:**

Ubuntu 22.04.3 LTS



**Git Repository:**

**URL**

https://github.com/RuihaoWei95/CSE241.git

**Commit ID**

e3db21d1d3186e18051990882214c61fc8a8e23d

**Create QEMU disk Images:**

1. Install QEMU

**sudo apt-get install qemu-system-x86**

**sudo apt install qemu-utils**

1. Download Ubuntu 20.04 Server

**wget <https://releases.ubuntu.com/focal/ubuntu-20.04.6-live-server-amd64.iso>**

1. Create QEMU disk image with qcow2 format

Run the command to create qcow2 format disk image with name ‘ubuntu.img’, storage space ‘10G’, format ‘-f qcow2’

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

1. Create QEMU disk image with raw format

Run the command to create raw format disk image with name ‘ubuntu-raw.img’, storage space ‘10G’, format ‘-f raw’

**sudo qemu-img create ubuntu-raw.img 10G -f raw**

1. Created Images



**Enable QEMU VM**

1. Install VM

Run the following command to install VM with downloaded Ubuntu server iso and images we just created. Follow the instruction to finish the installation.

**sudo qemu-system-x86\_64 -hda <`Your Image Name`> -boot d -cdrom ./ubuntu-20.04.6-live-server-amd64.iso -m 2046 -boot strict=on**

-hda <`Your Image Name`>: specifies the virtual hard disk drive

-boot d: sets the boot device order, `d` means it will first try to boot from the CD-ROM drive

-cdrom ./ubuntu-20.04.6-live-server-amd64.iso: specifies an ISO image file as the CD-ROM drive content

-m 2046: sets the memory size to 2046 MB (2GB)

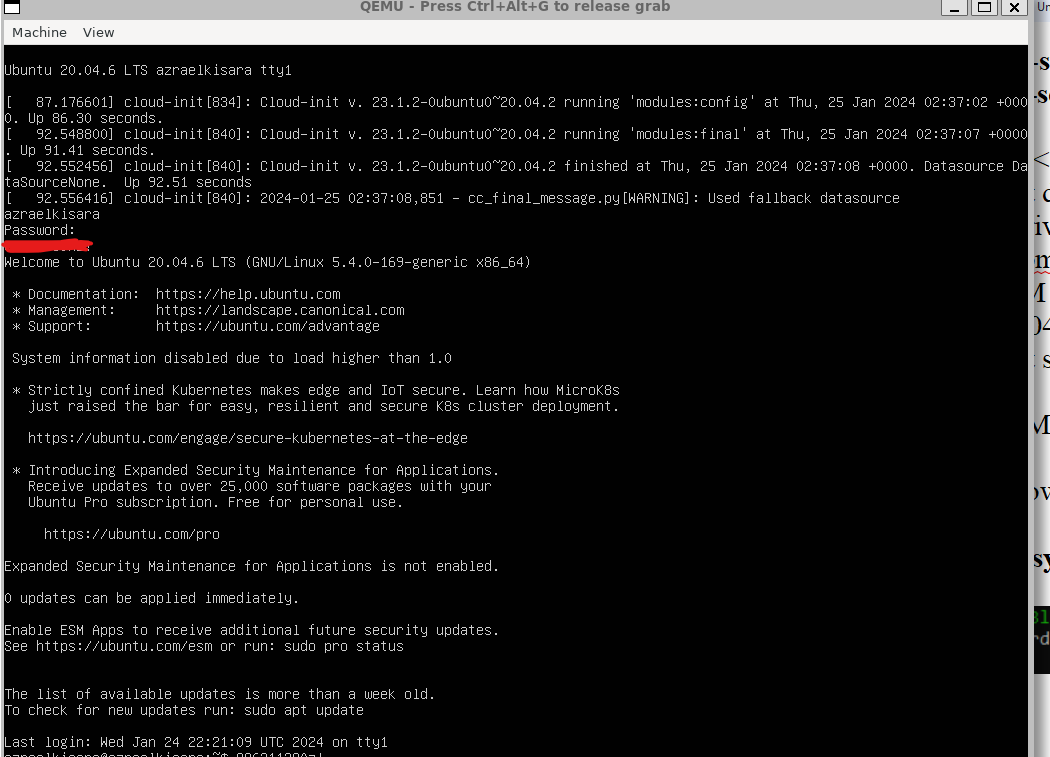
-boot strict=on: enables strict boot order

1. Enable VM

Run the following command to enable VM after the installation is succeed

**sudo qemu-system-x86\_64 -m 2046 -hda <`Your Image Name`>**





**Create Docker container Image**

1. Install Docker

Follow [instruction](https://docs.docker.com/engine/install/ubuntu/) to install Docker engine on Ubuntu.

1. Create a Dockerfile

**touch Dockerfile**

1. Edit the Dockerfile to add sysbench to a base Ubuntu 20.04 image

**FROM ubuntu:20.04**

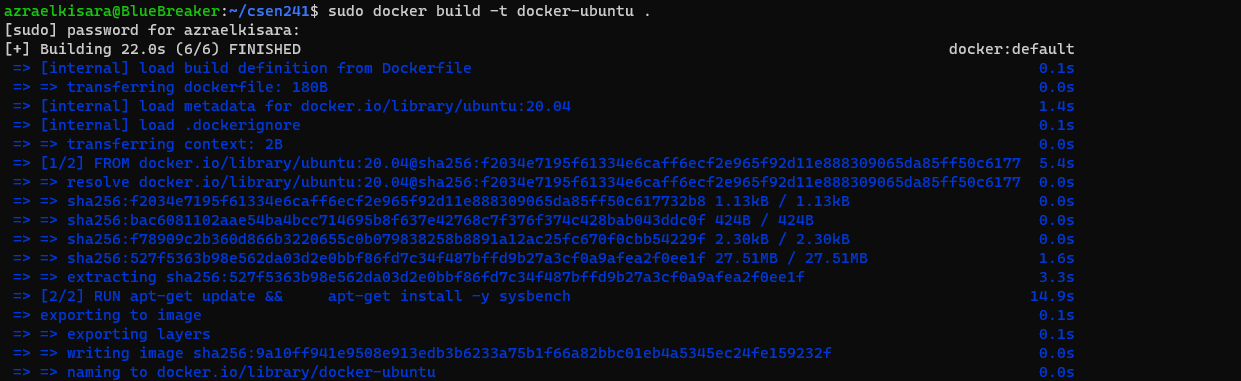
**Run apt-get update && apt-get install -y sysbench**

1. Build the image from edited Dockerfile

Run following command to build image from a dockerfile. Replace `<tag>` with

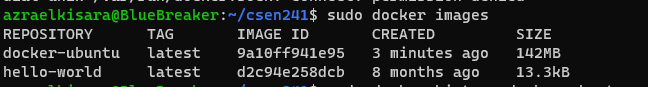
image name.

**sudo docker build -t <tag> .**



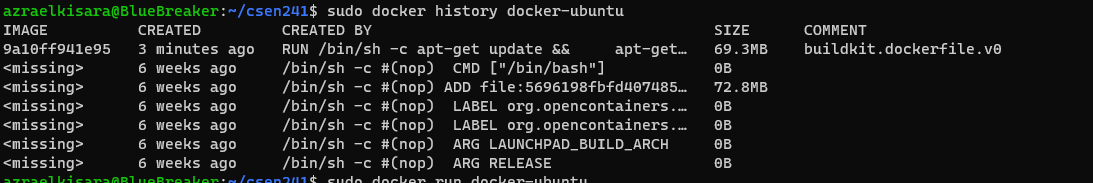
1. Check image ID

**sudo docker images**



1. Check image history

**sudo docker history <tag>**



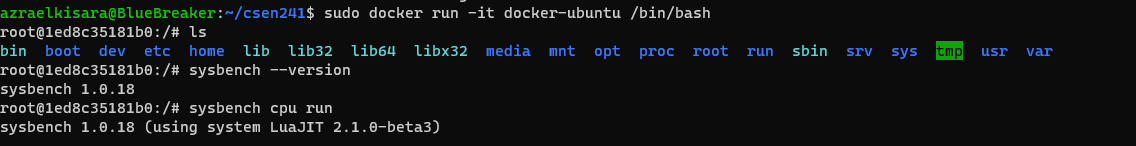
1. Run the container

Run the following command to start our container

**sudo docker run -it <tag> /bin/bash**

-it: Run container in interactive mode with terminal

/bin/bash: Bash command to start a shell



1. Other useful docker commands

**docker ps**

This command list all running containers



**docker stop <NAMES>**

This command stop the container with given name



**QEMU Experiment**

VM configuration arguments**:**

CPU：2，4

RAM: 2048, 4096

CPU 2, RAM 2048 is the basic multi-core setup. The idea is to start with a modest number of cores and amount of RAM and potentially increase to see if there is a significant improvement in performance.

1. Enable VMs:

Run following command to enable VM with different configurations. We also want to redirect port so we can SSH to VM.

**qemu-system-x86\_64 -hda <img name> -smp <cpu argument> -m <memory argument> -net user, hostfwd-tcp::<port>-:22 -net nic**

Run following command on VM to enable SSH

**sudo apt-get update**

**sudo apt-get install openssh-server**

**sudo systemctl start ssh**

Run following command on Host to SSH VM

**ssh -p <port> user@localhost**

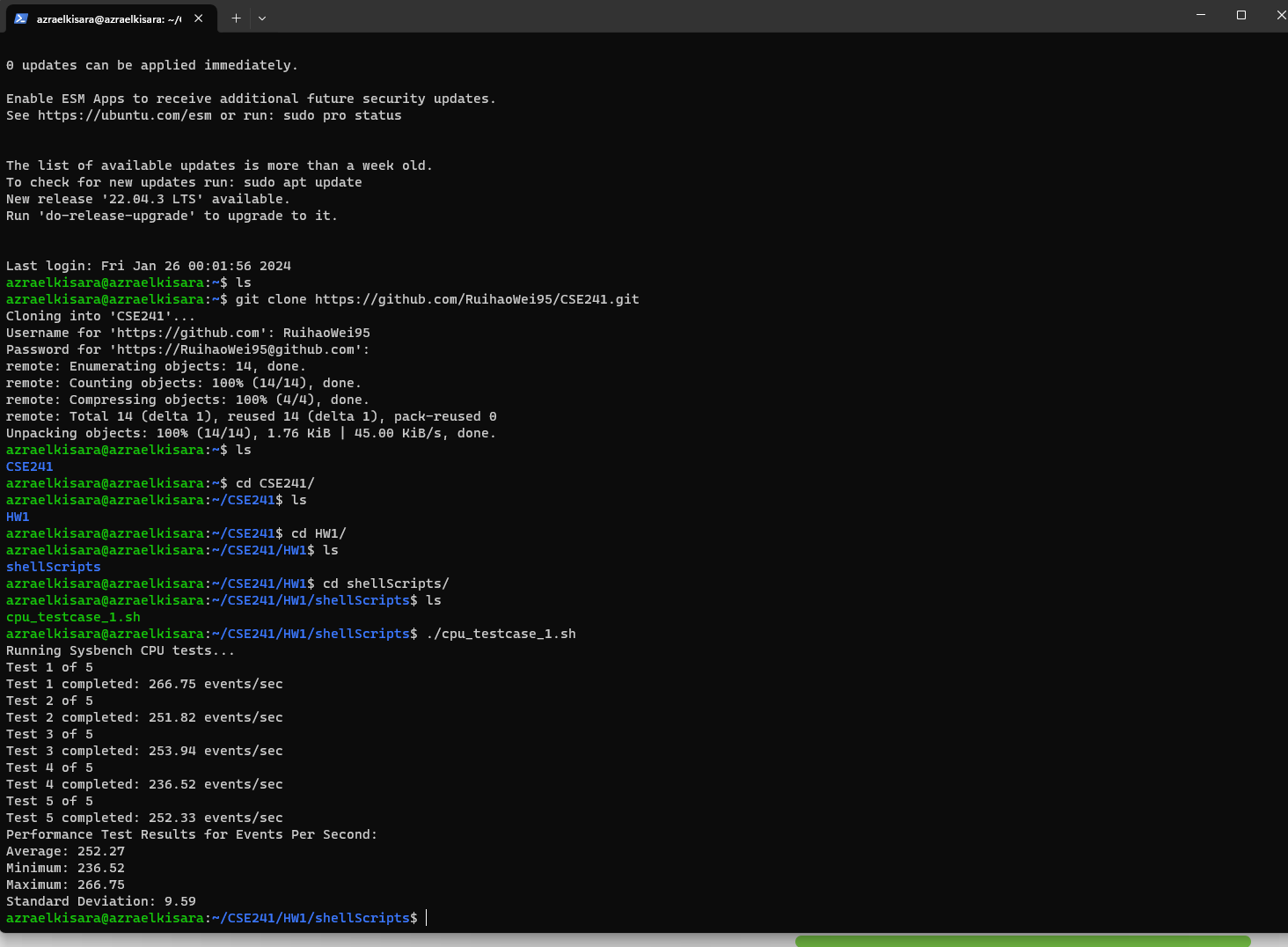
1. Sysbench test cases:
   1. CPU test case 1: --cpu-max-prime=20000 --time=20 run
   2. CPU test case 2: --cpu-max-prime=80000 --time=20 run
   3. Memory test case 1: --memory-block-size=1K
   4. Memory test case 2: --memory-block-size=2K
   5. FileIO test case 1: --file-total-size=1G --file-test-mode=rndrw
   6. FileIO test case 2: --file-total-size=2G --file-test-mode=rndrw
2. Experiment with qcow2 image:

**Proof of experiment**

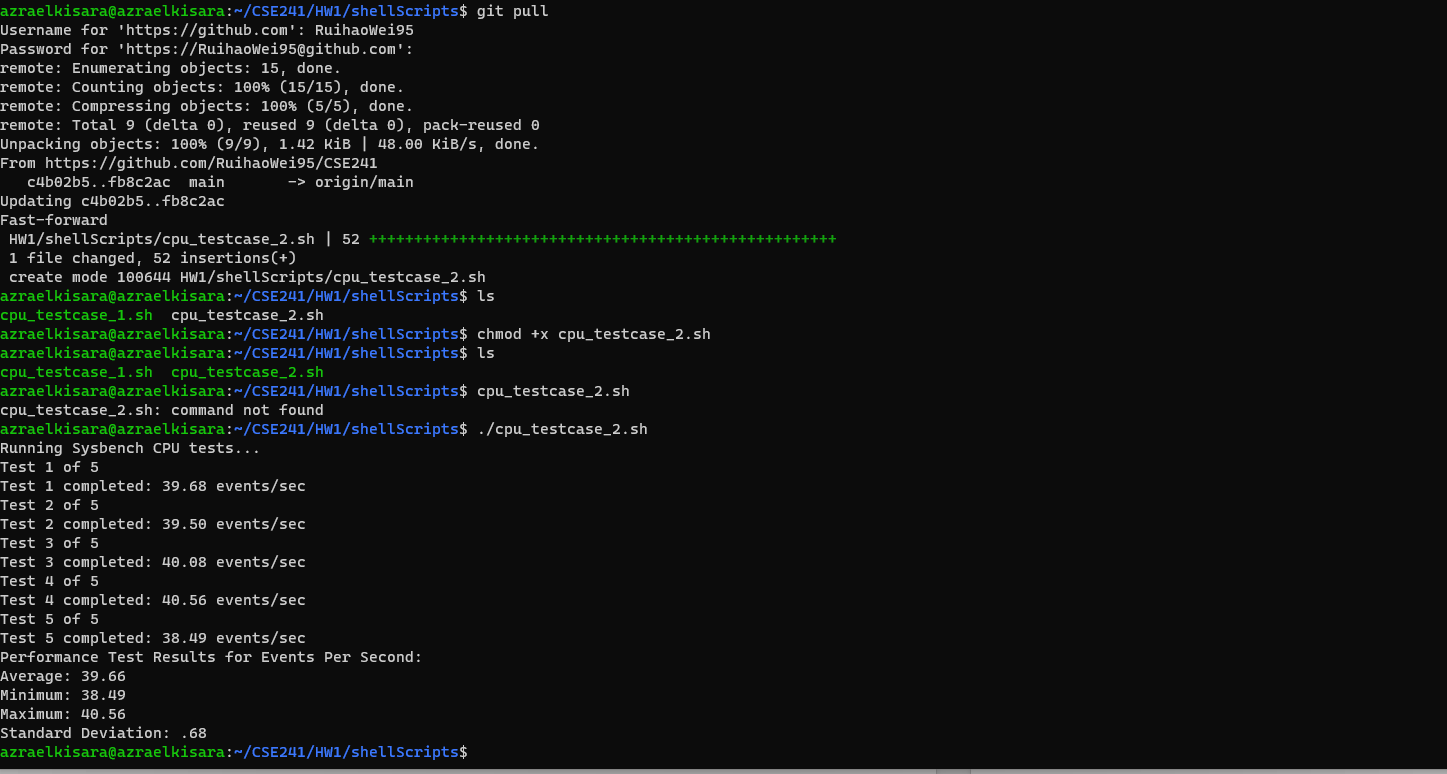
* 1. cpu: 2, RAM: 2048



CPU test case 1



CPU test case 2

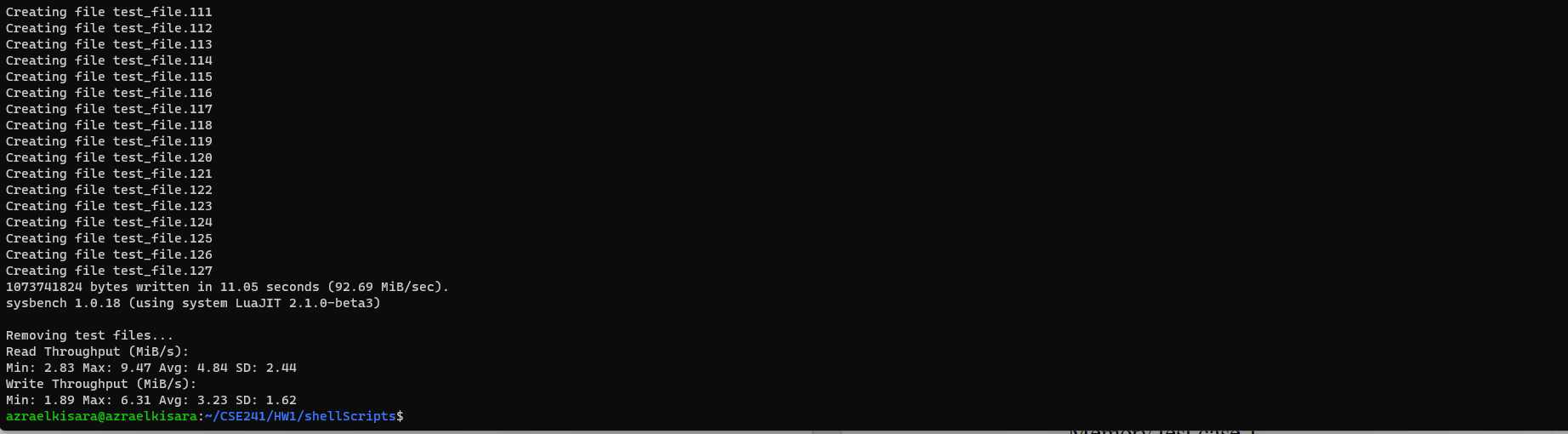
Memory test case 1



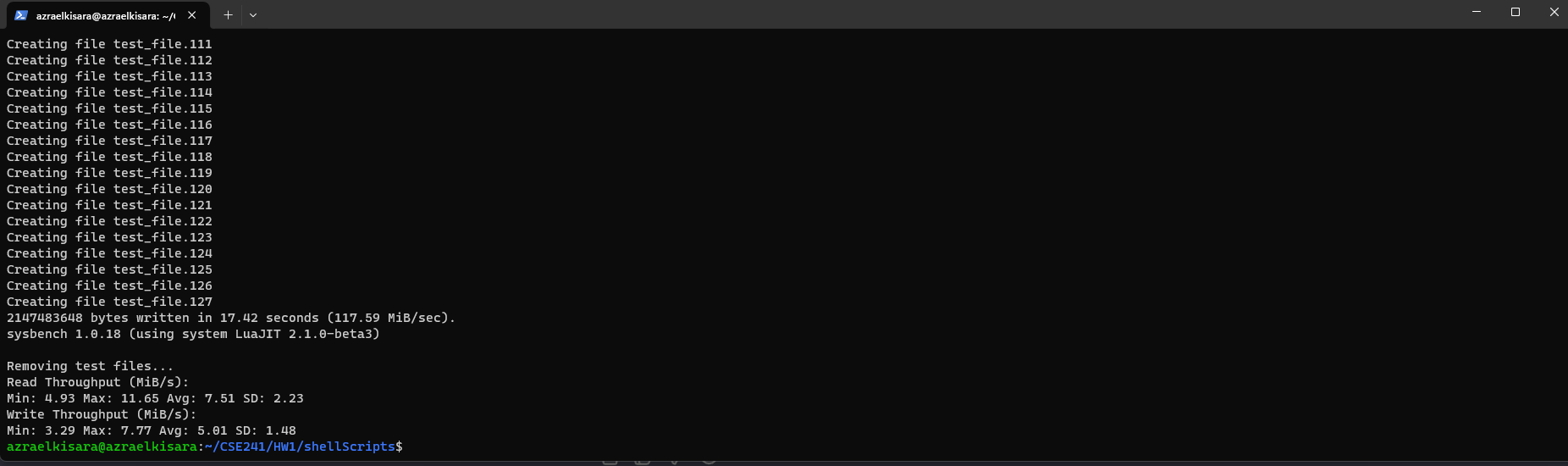
Memory test case 2



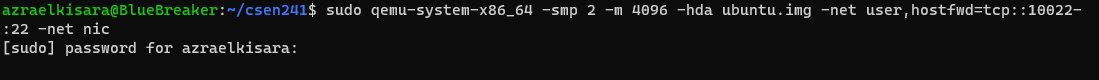
FileIO test case 1



FileIO test case 2



* 1. cpu: 2, RAM: 4096



CPU test case 1



CPU test case 2



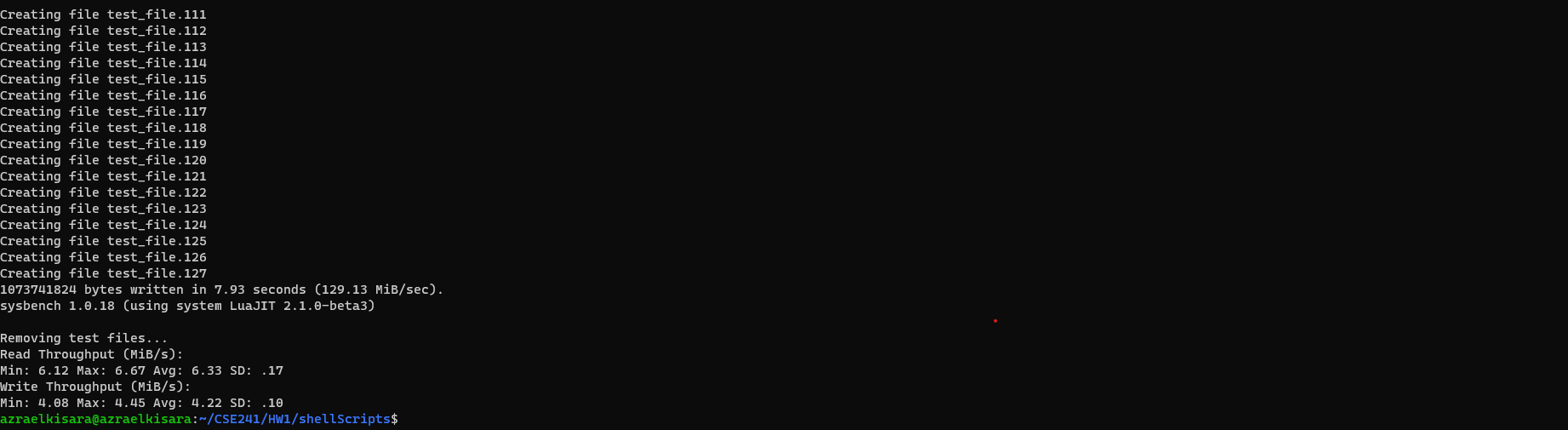
Memory test case 1



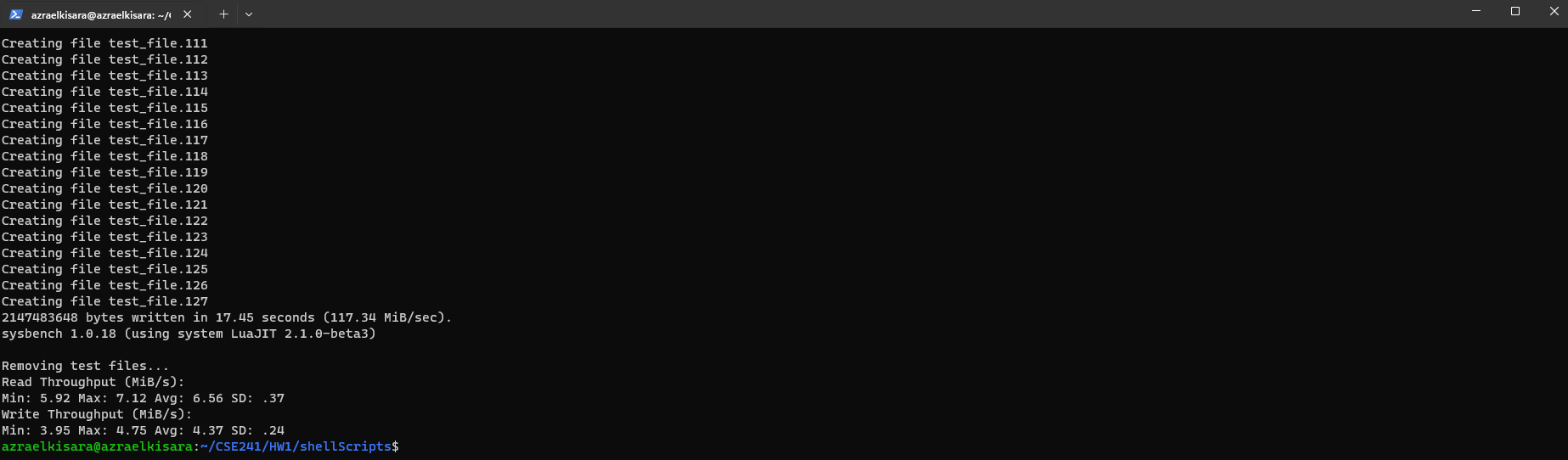
Memory test case 2



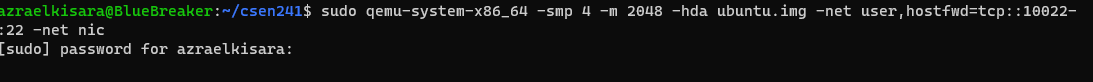
FileIO test case 1



FileIO test case 2



* 1. cpu: 4, RAM: 2048



CPU test case 1



CPU test case 2



Memory test case 1



Memory test case 2



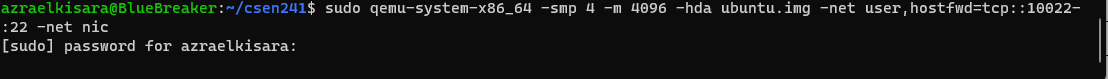
FileIO test case 1



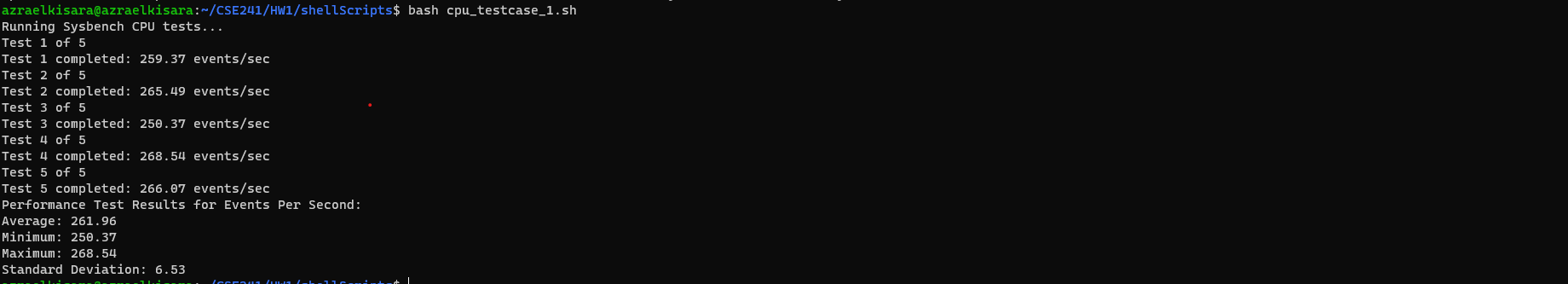
FileIO test case 2



* 1. cpu: 4, RAM: 4096



CPU test case 1



CPU test case 2



Memory test case 1



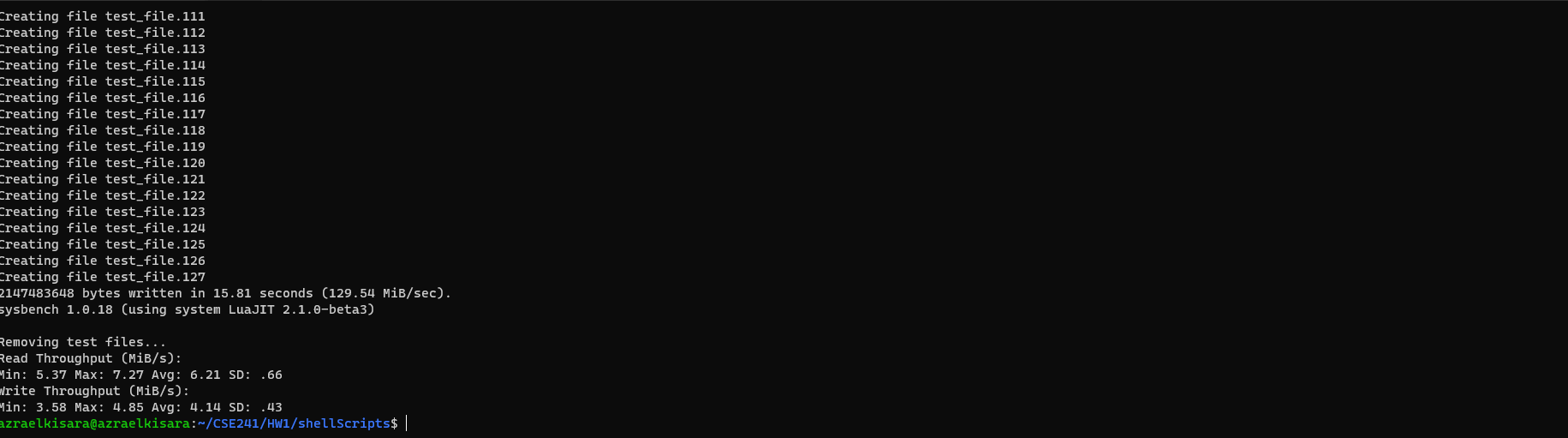
Memory test case 2



FileIO test case 1



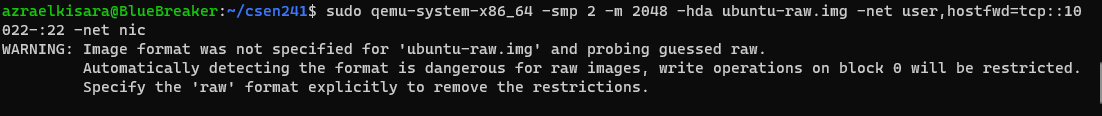
FileIO test case 2



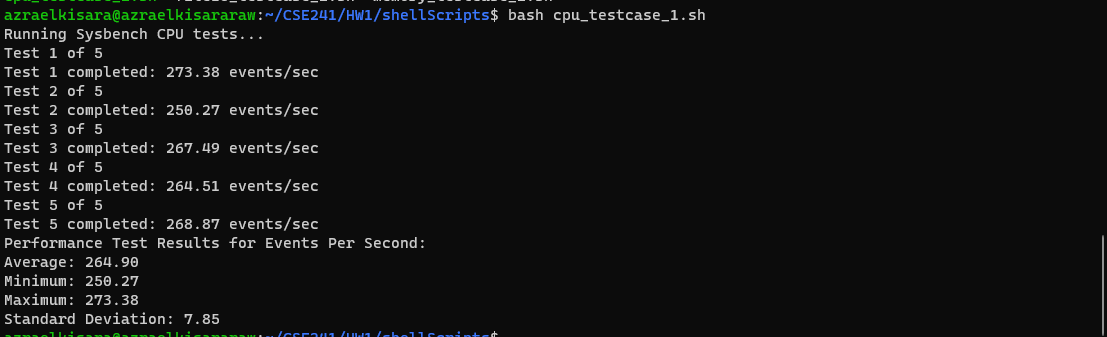
1. Experiment with raw image:

**Proof of experiment**

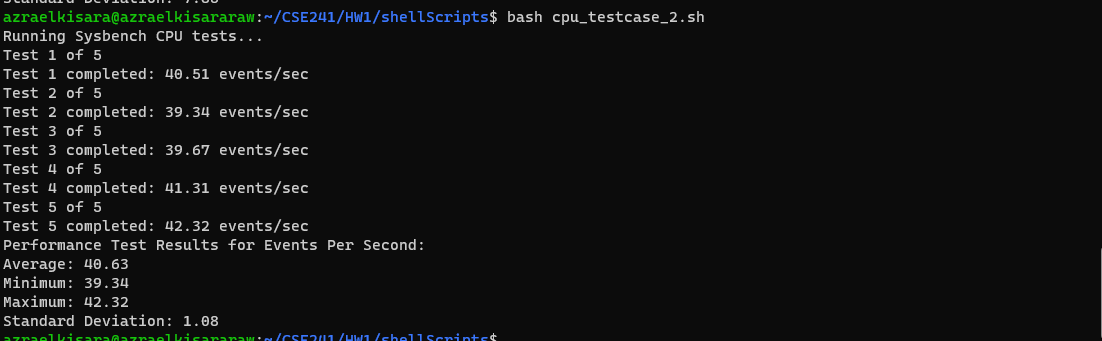
* 1. cpu: 2, RAM: 2048



CPU test case 1



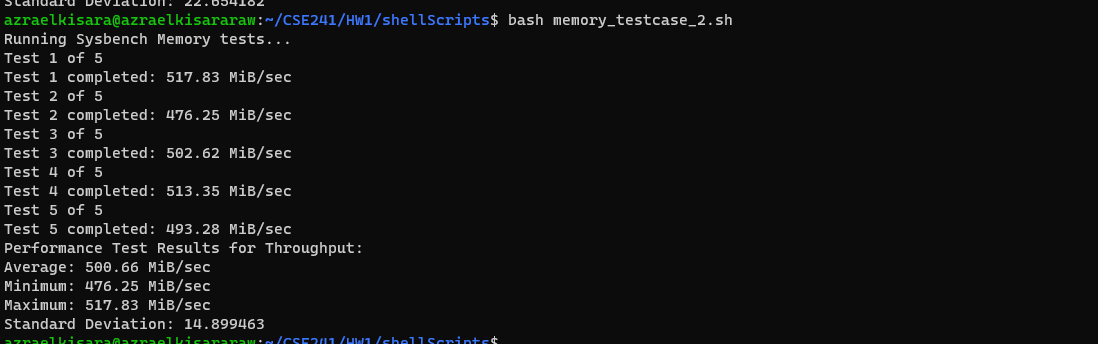
CPU test case 2



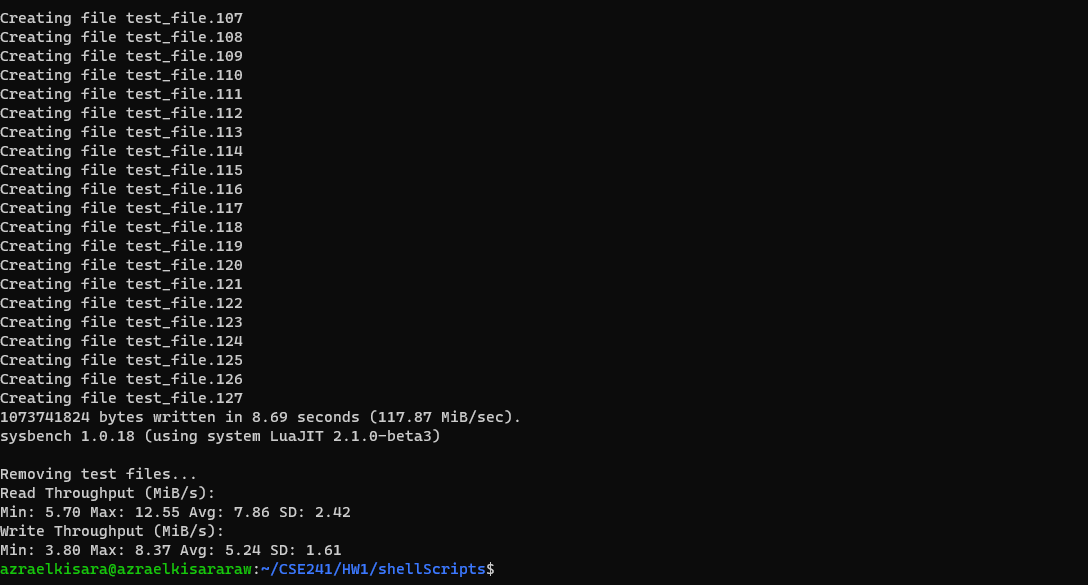
Memory test case 1



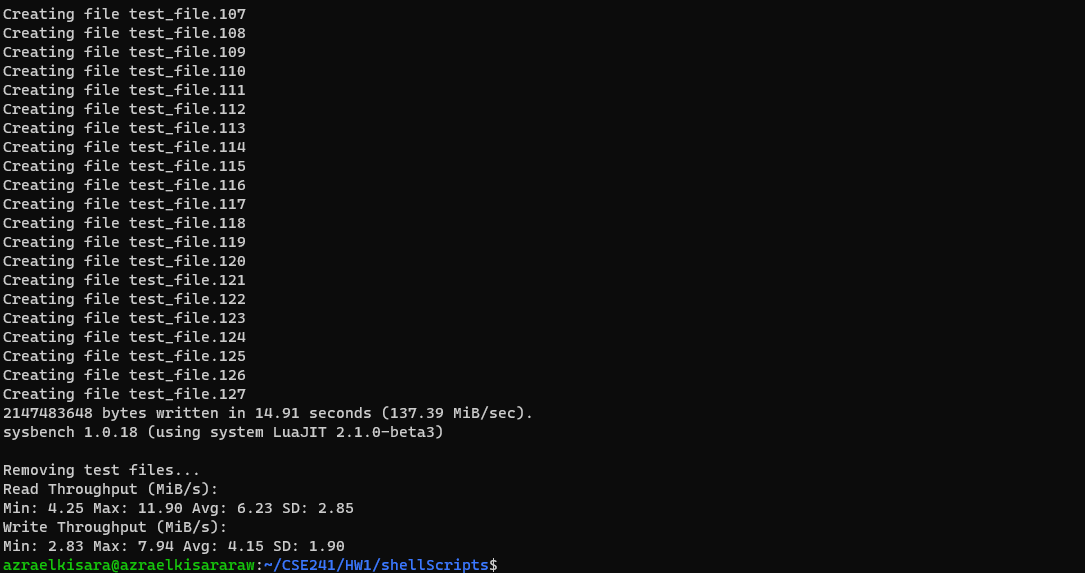
Memory test case 2



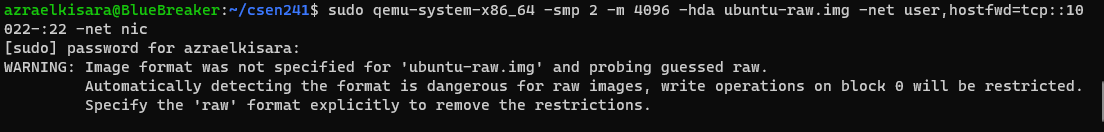
FileIO test case 1



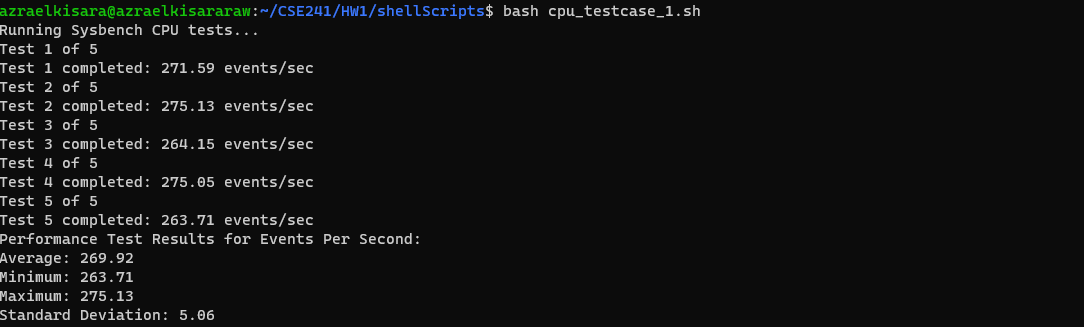
FileIO test case 2



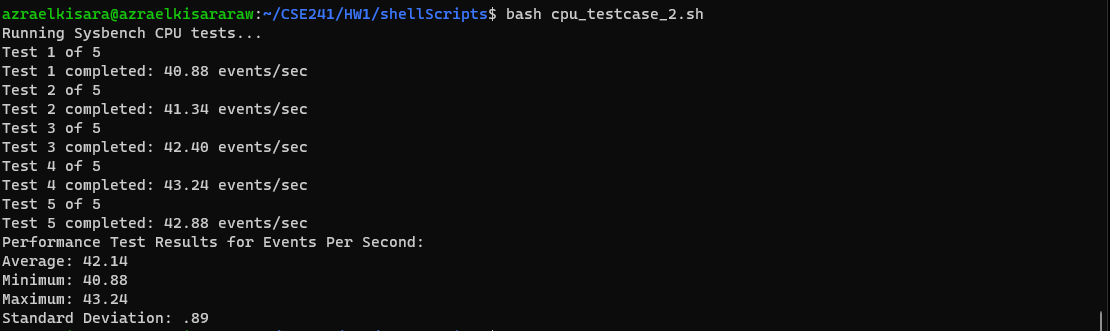
* 1. cpu: 2, RAM: 4096



CPU test case 1



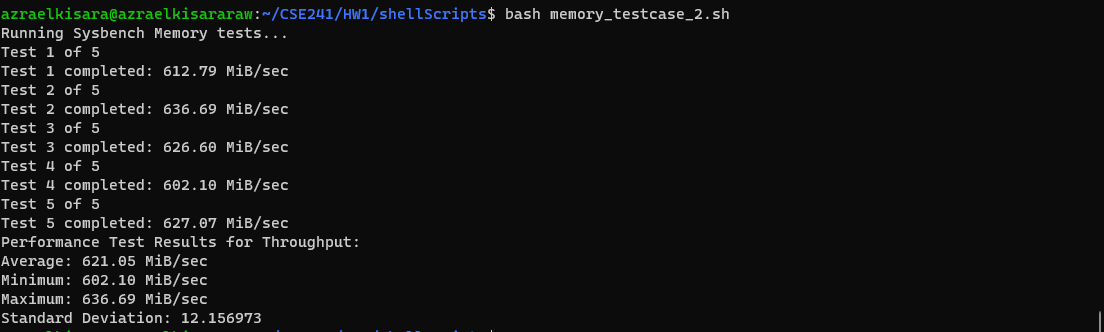
CPU test case 2



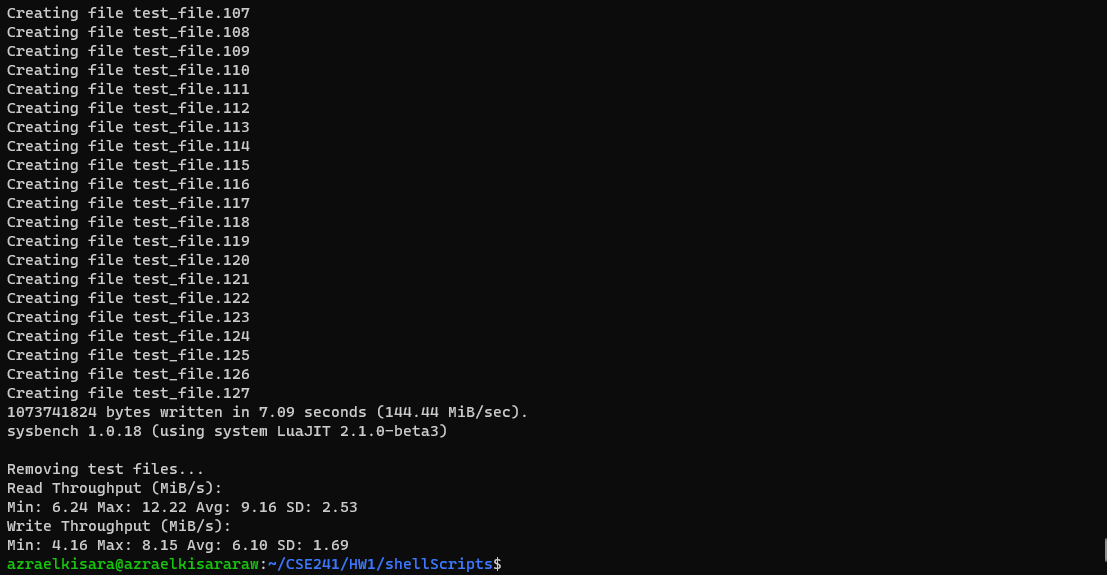
Memory test case 1



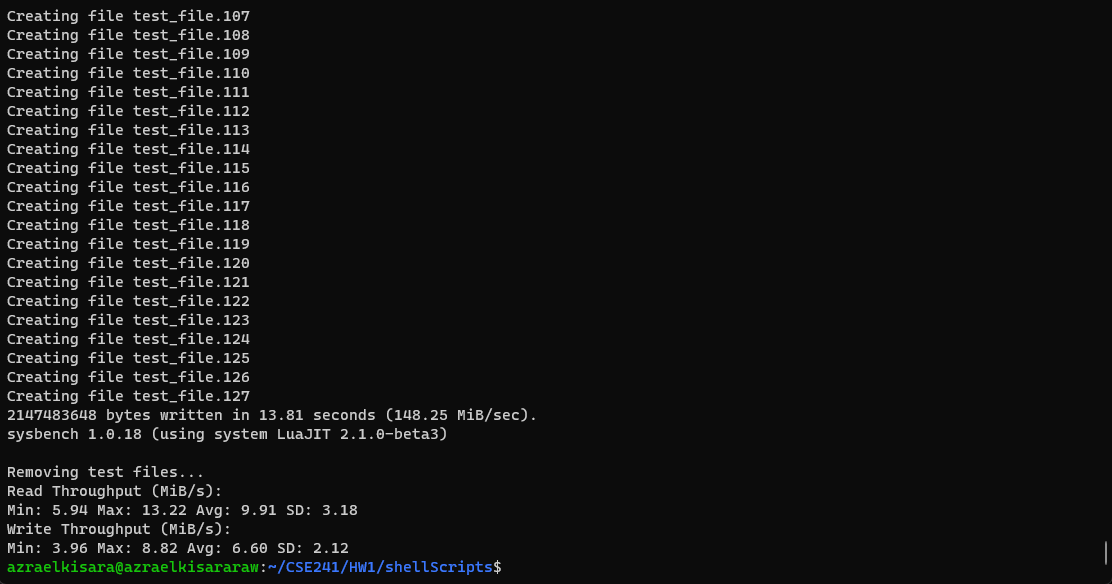
Memory test case 2



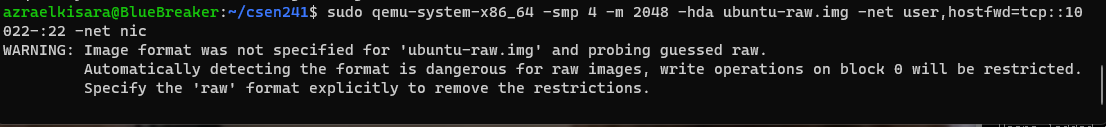
FileIO test case 1



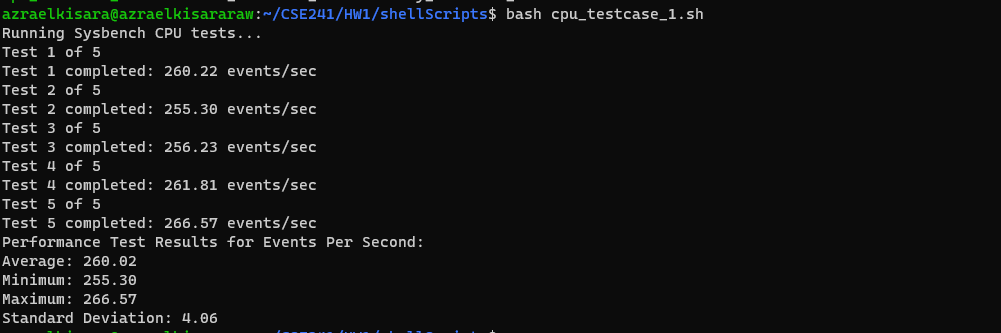
FileIO test case 2



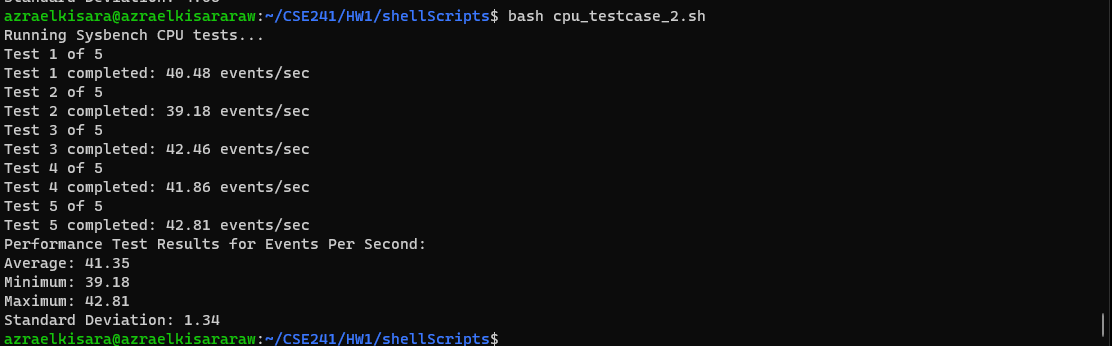
* 1. cpu: 4, RAM: 2048



CPU test case 1



CPU test case 2



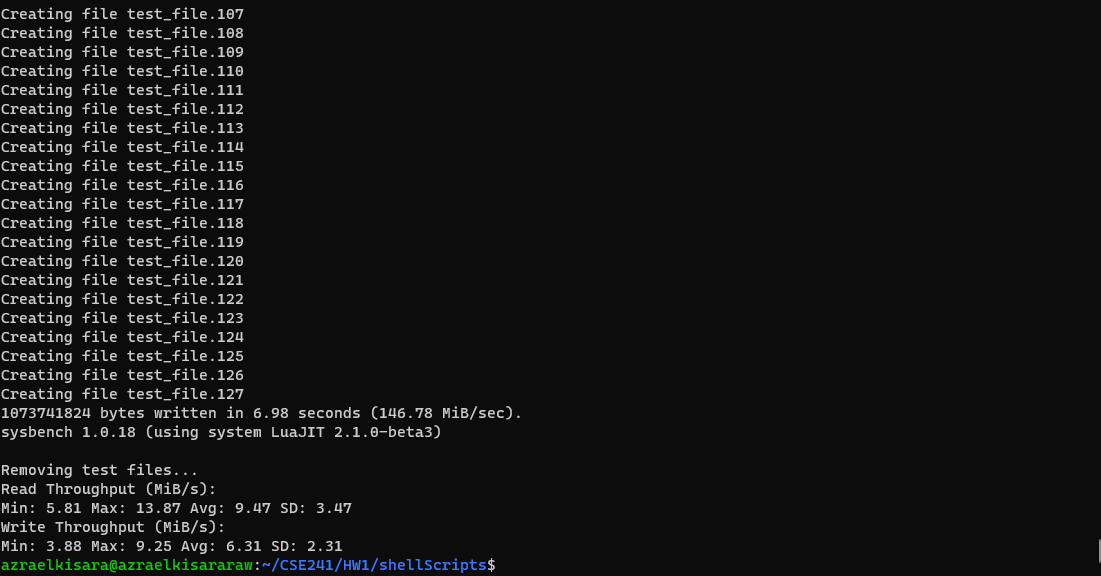
Memory test case 1



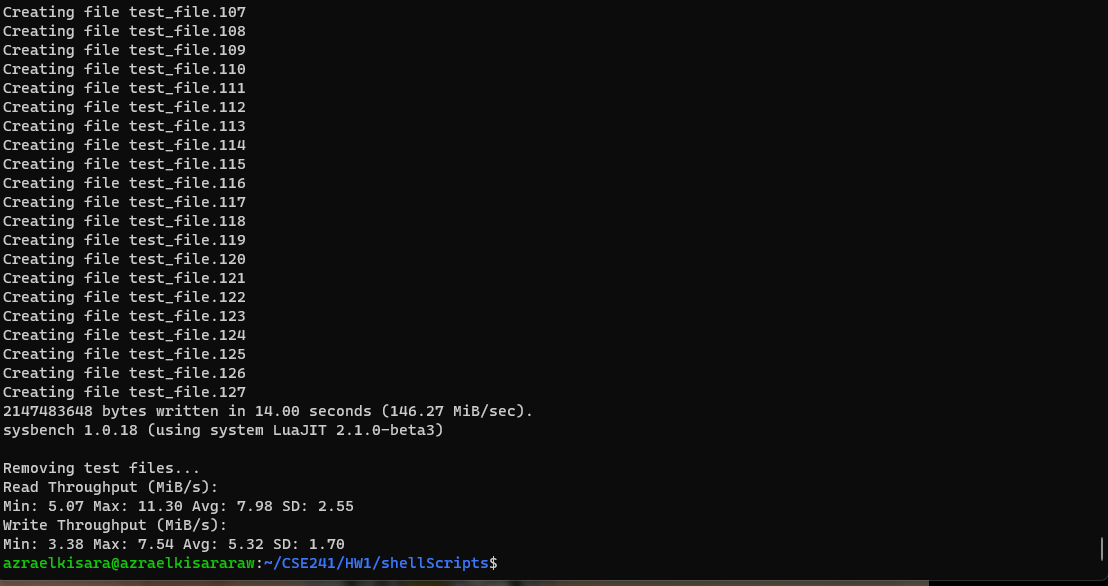
Memory test case 2



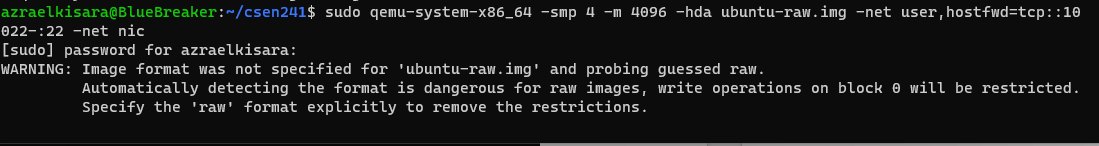
FileIO test case 1



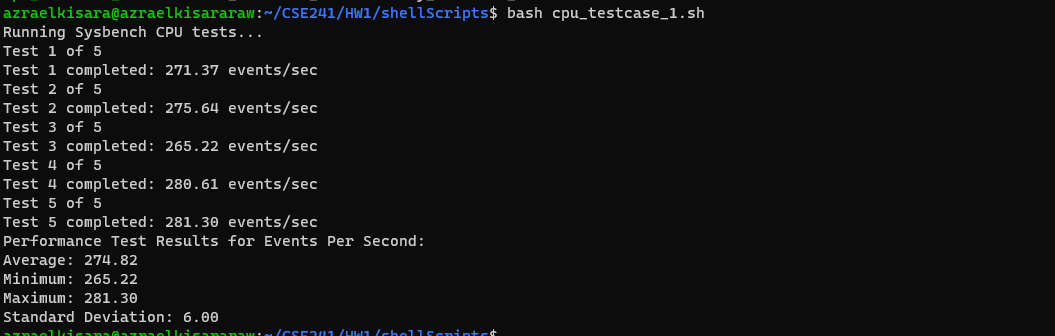
FileIO test case 2



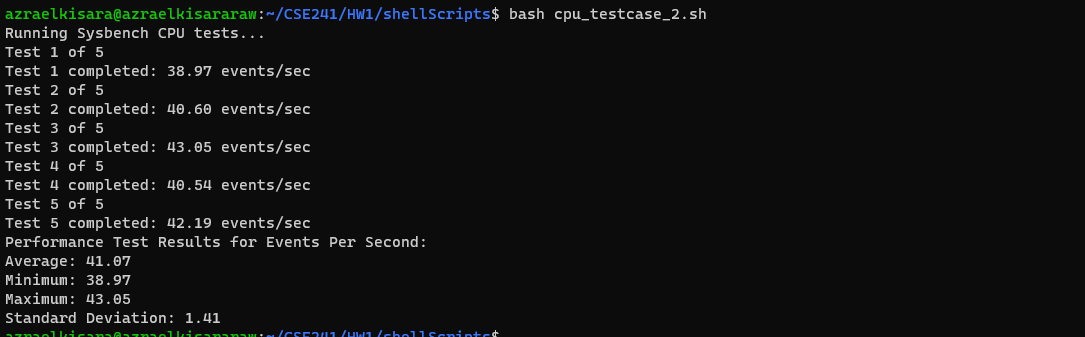
* 1. cpu: 4, RAM: 4096



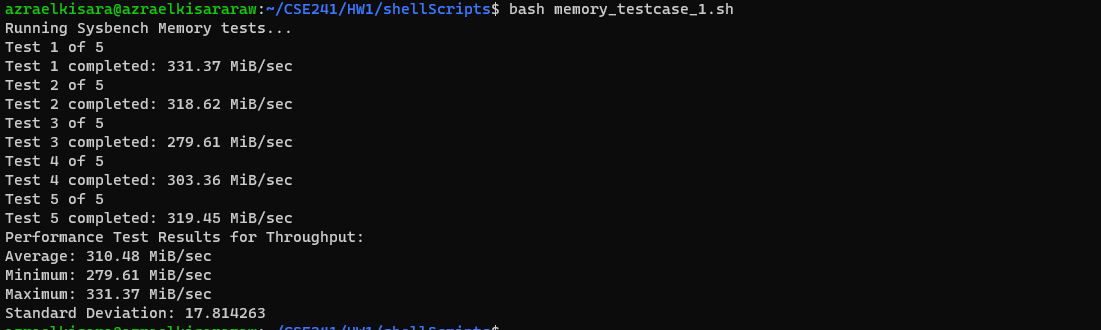
CPU test case 1



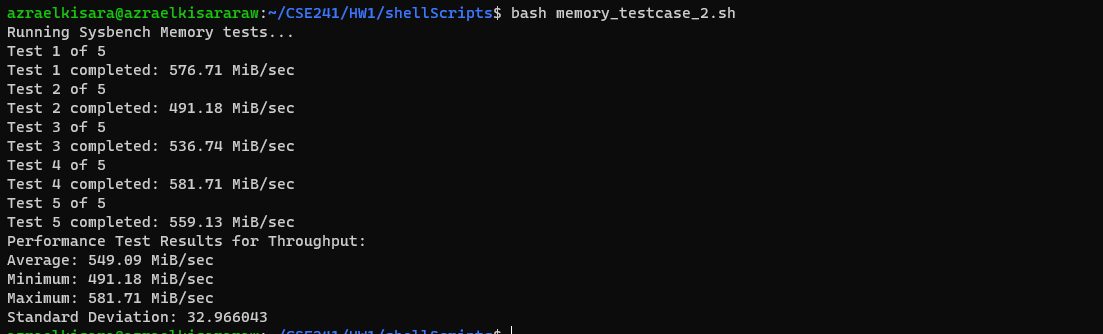
CPU test case 2



Memory test case 1



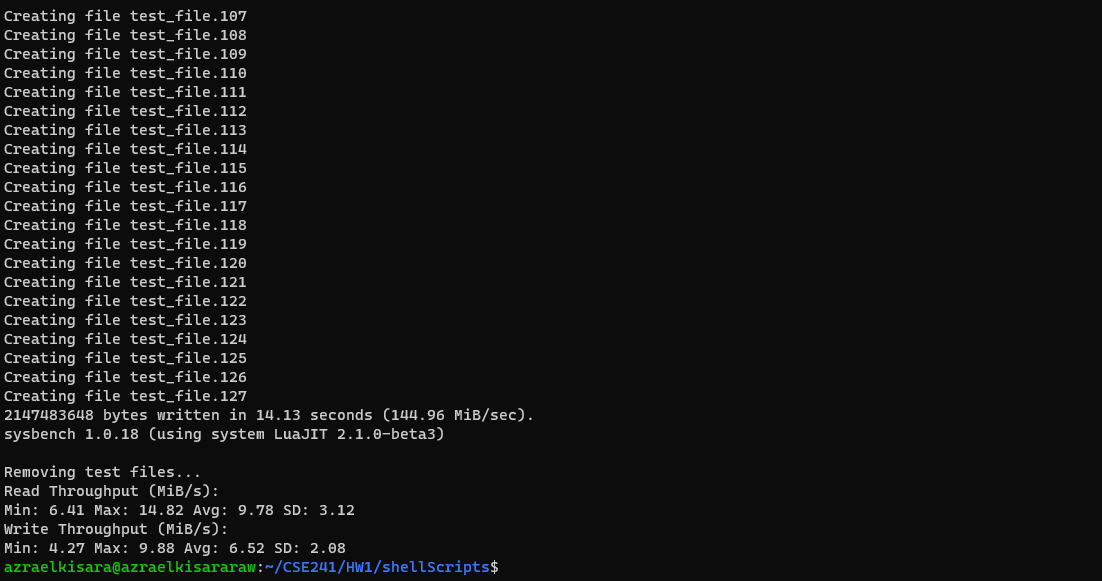
Memory test case 2



FileIO test case 1



FileIO test case 2



**Docker Experiment**

1. Start Container

Run the following command to start container from image

**sudo docker run --cpus <cpu percentage> --memory <memory limit>m -it <image> /bin/bash**

My host has 24 cpus, to limit to 2 cpus, cpus = “0.09”. to limit to 4 cpus cpus = “0.17”

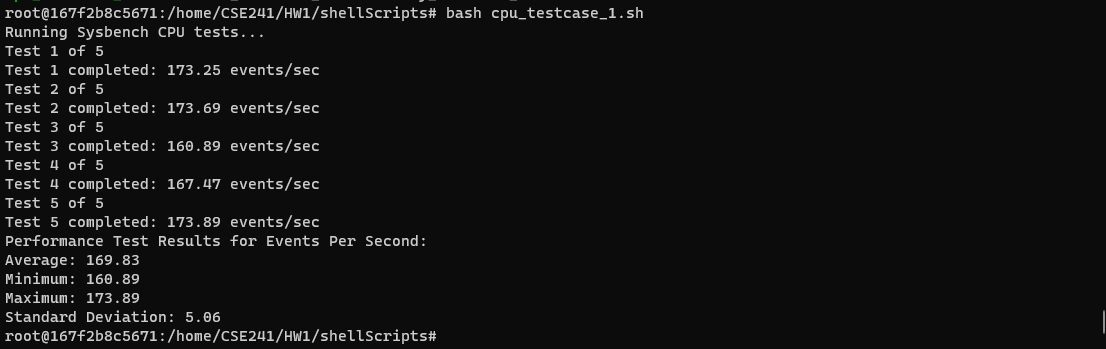
1. Experiment with docker image:

**Proof of experiment**

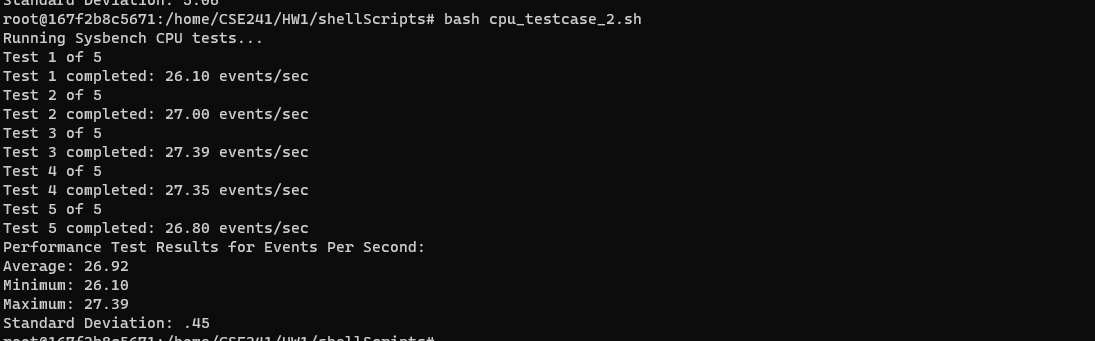
* 1. cpu: 2, RAM: 2048



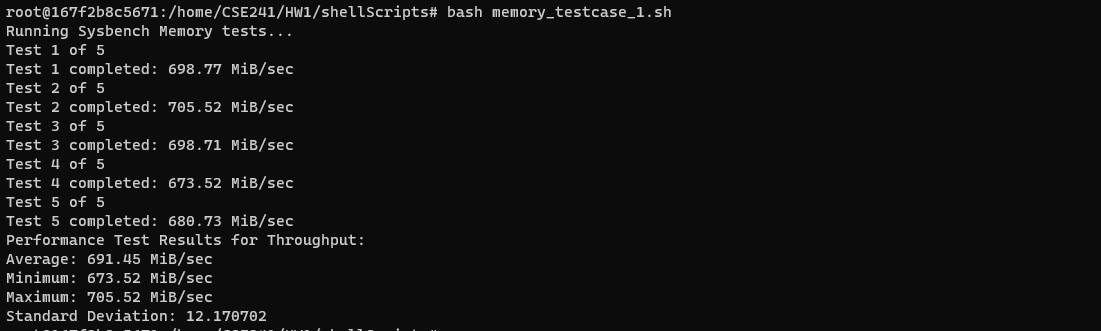
CPU test case 1



CPU test case 2



Memory test case 1



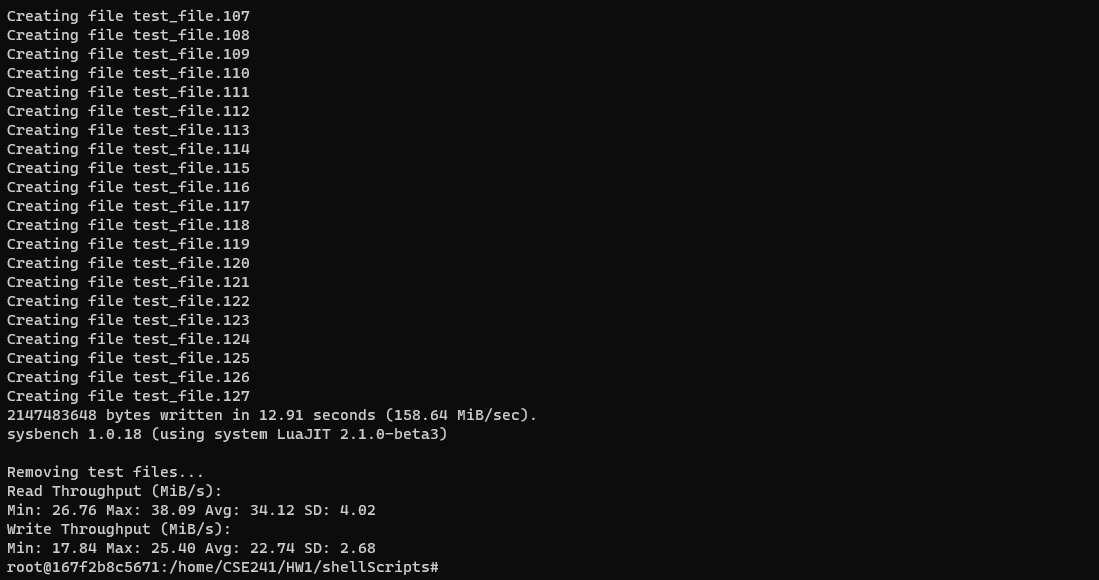
Memory test case 2



FileIO test case 1



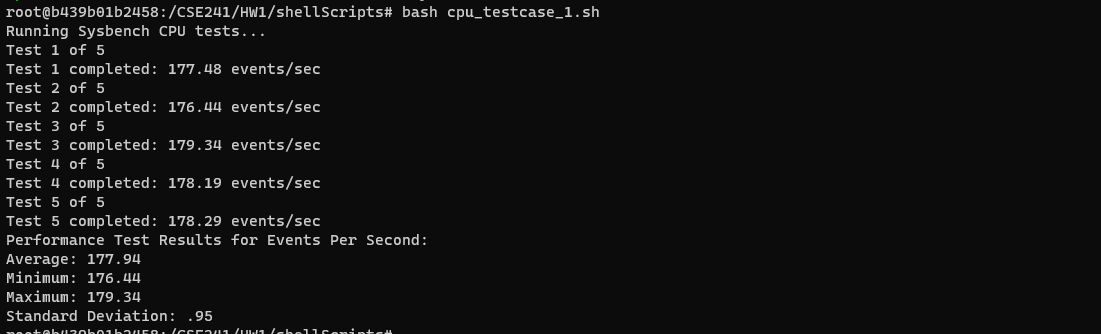
FileIO test case 2



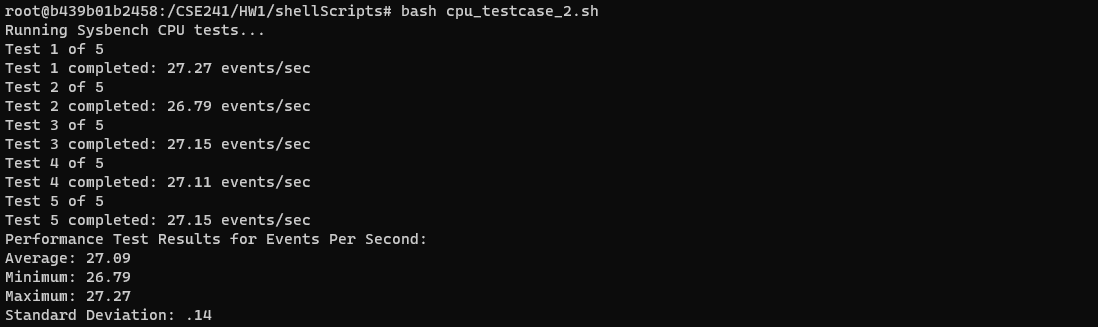
* 1. cpu: 2, RAM: 4096



CPU test case 1



CPU test case 2



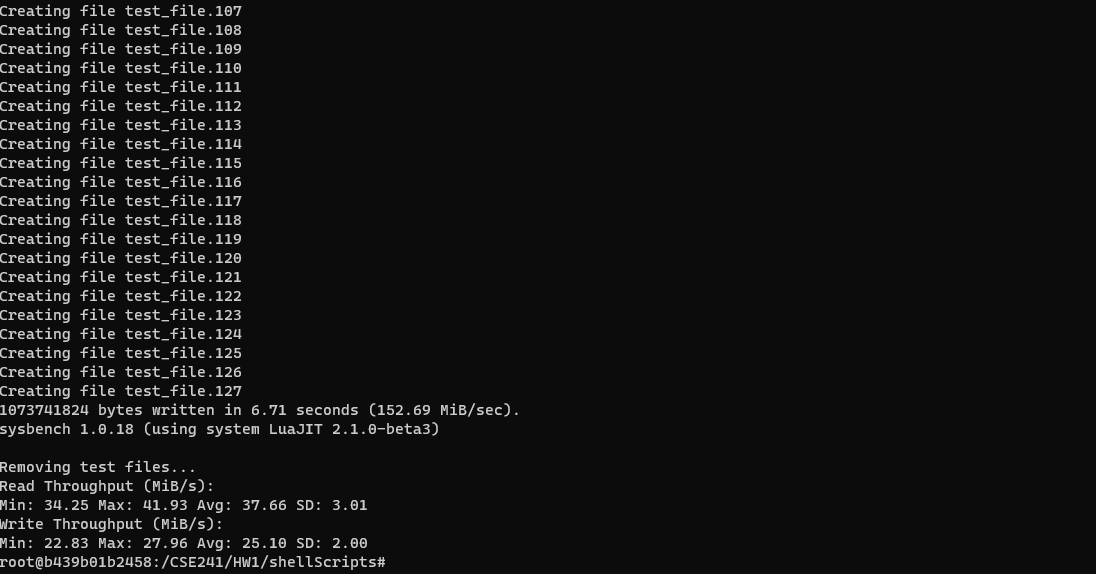
Memory test case 1



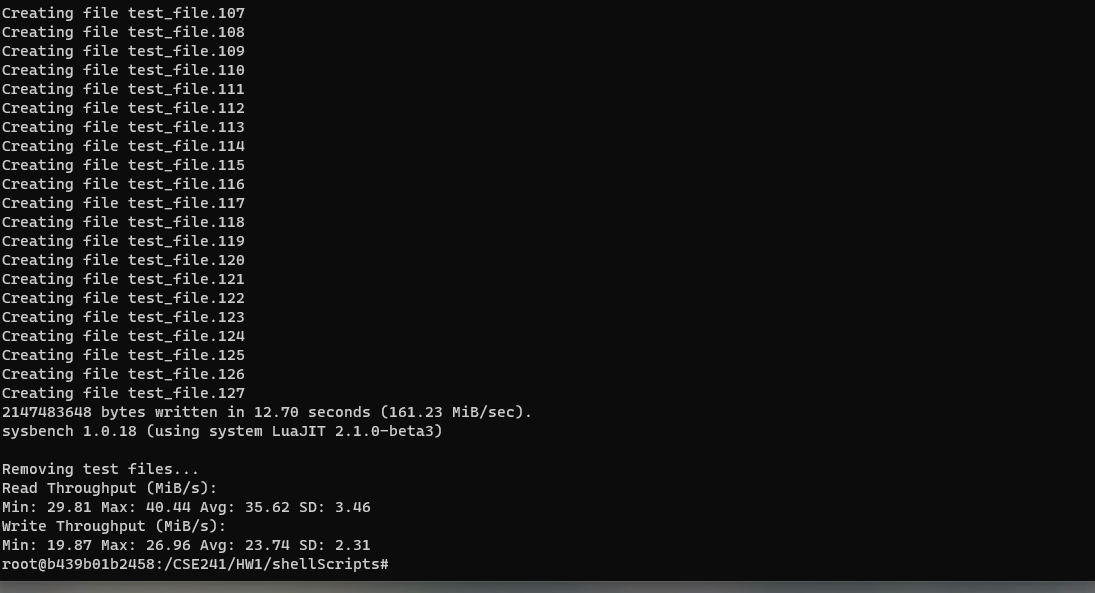
Memory test case 2



FileIO test case 1



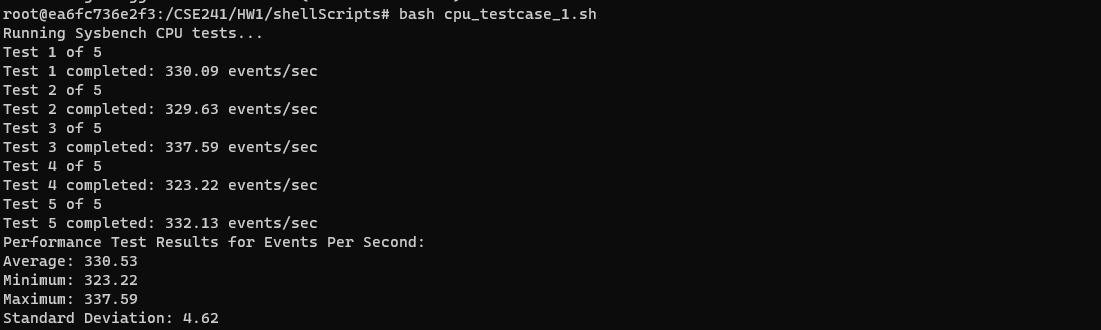
FileIO test case 2



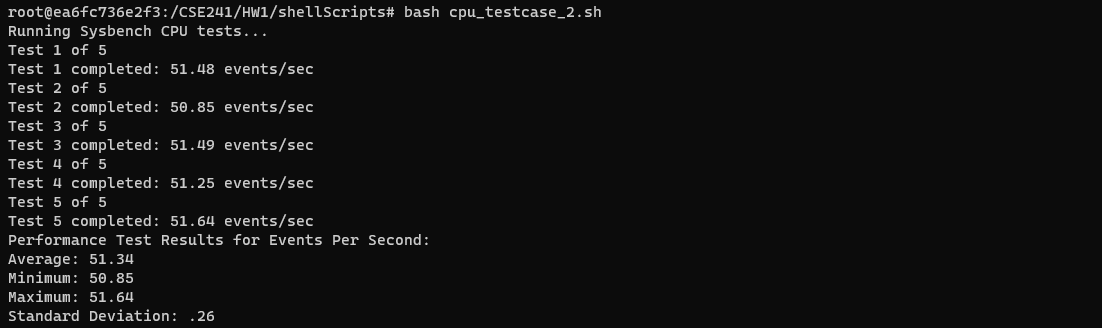
* 1. cpu: 4, RAM: 2048



CPU test case 1



CPU test case 2



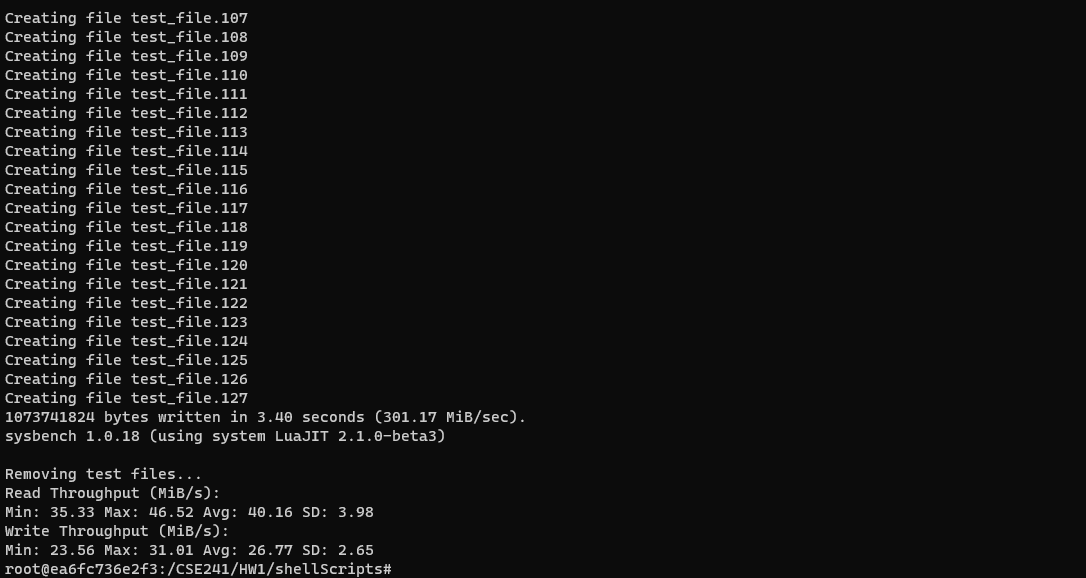
Memory test case 1



Memory test case 2



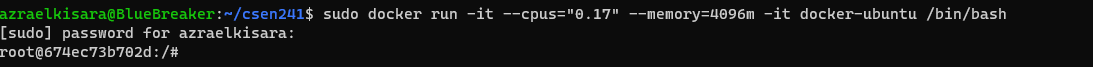
FileIO test case 1



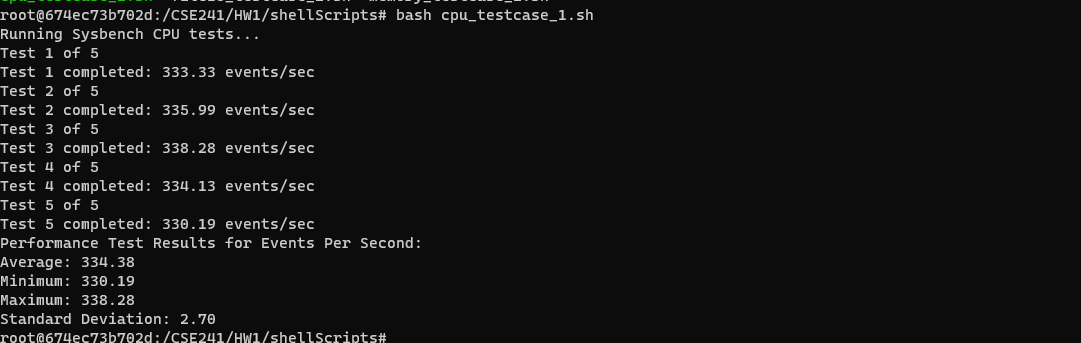
FileIO test case 2



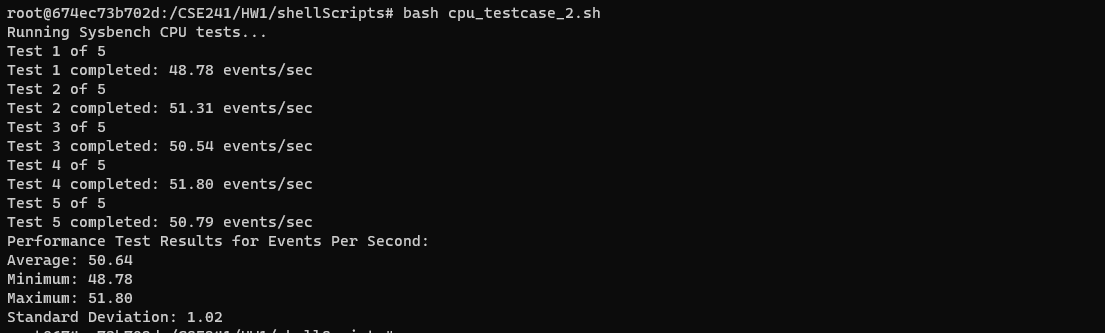
* 1. cpu: 4, RAM: 4096



CPU test case 1



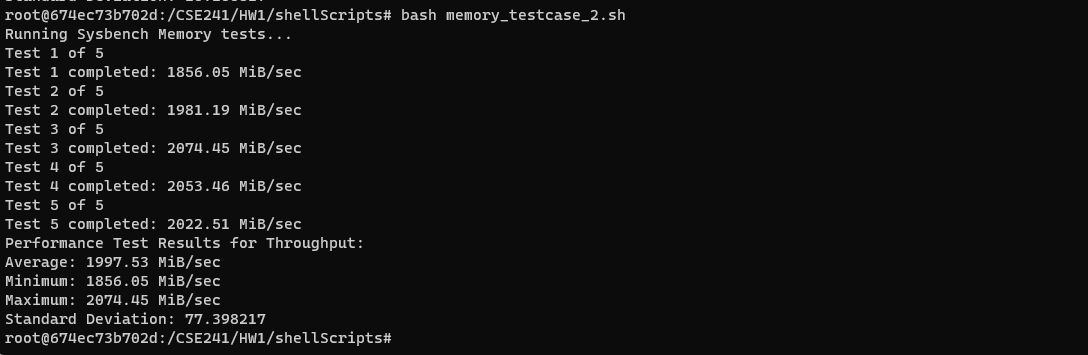
CPU test case 2



Memory test case 1



Memory test case 2



FileIO test case 1



FileIO test case 2



**Measurements**

**CPU Test (events/sec)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Image** | **CPU** | **RAM** | **Parameter** | **Value** | **min** | **max** | **avg** | **sd** |
| qcow2 | 2 | 2048 | cpu-max-prime | 20000 | 236.52 | 266.75 | 252.27 | 9.59 |
| qcow2 | 2 | 2048 | cpu-max-prime | 80000 | 38.49 | 40.56 | 39.66 | 0.68 |
| qcow2 | 2 | 4096 | cpu-max-prime | 20000 | 235.19 | 256.56 | 243.60 | 9.93 |
| qcow2 | 2 | 4096 | cpu-max-prime | 80000 | 40.36 | 41.93 | 41.19 | 0.63 |
| qcow2 | 4 | 2048 | cpu-max-prime | 20000 | 244.78 | 279.35 | 265.75 | 11.63 |
| qcow2 | 4 | 2048 | cpu-max-prime | 80000 | 38.14 | 41.07 | 39.71 | 1.17 |
| qcow2 | 4 | 4096 | cpu-max-prime | 20000 | 250.37 | 268.54 | 261.96 | 6.53 |
| qcow2 | 4 | 4096 | cpu-max-prime | 80000 | 39.81 | 42.09 | 41.13 | 0.91 |
| raw | 2 | 2048 | cpu-max-prime | 20000 | 250.27 | 273.38 | 264.90 | 7.85 |
| raw | 2 | 2048 | cpu-max-prime | 80000 | 39.34 | 42.32 | 40.63 | 1.08 |
| raw | 2 | 4096 | cpu-max-prime | 20000 | 263.71 | 275.13 | 269.92 | 5.06 |
| raw | 2 | 4096 | cpu-max-prime | 80000 | 40.88 | 43.24 | 42.14 | 0.89 |
| raw | 4 | 2048 | cpu-max-prime | 20000 | 255.30 | 266.57 | 260.02 | 4.06 |
| raw | 4 | 2048 | cpu-max-prime | 80000 | 39.18 | 42.81 | 41.35 | 1.34 |
| raw | 4 | 4096 | cpu-max-prime | 20000 | 265.22 | 281.30 | 274.82 | 6.00 |
| raw | 4 | 4096 | cpu-max-prime | 80000 | 38.97 | 43.05 | 41.07 | 1.41 |
| docker | 2 | 2048 | cpu-max-prime | 20000 | 160.89 | 173.89 | 169.83 | 5.06 |
| docker | 2 | 2048 | cpu-max-prime | 80000 | 26.10 | 27.39 | 26.92 | 0.45 |
| docker | 2 | 4096 | cpu-max-prime | 20000 | 176.44 | 179.34 | 177.94 | 0.95 |
| docker | 2 | 4096 | cpu-max-prime | 80000 | 26.79 | 27.27 | 27.09 | 0.14 |
| docker | 4 | 2048 | cpu-max-prime | 20000 | 323.22 | 337.59 | 330.53 | 4.62 |
| docker | 4 | 2048 | cpu-max-prime | 80000 | 50.85 | 51.64 | 51.34 | 0.26 |
| docker | 4 | 4096 | cpu-max-prime | 20000 | 330.19 | 338.28 | 334.38 | 2.70 |
| docker | 4 | 4096 | cpu-max-prime | 80000 | 48.78 | 51.80 | 50.64 | 1.02 |

**Memory Test (MiB/sec)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Image** | **CPU** | **RAM** | **Parameter** | **Value** | **min** | **max** | **avg** | **sd** |
| qcow2 | 2 | 2048 | memory\_block\_size | 1K | 287.49 | 331.24 | 318.46 | 16 |
| qcow2 | 2 | 2048 | memory\_block\_size | 2K | 545.17 | 638.13 | 597.46 | 32.94 |
| qcow2 | 2 | 4096 | memory\_block\_size | 1K | 224.46 | 310.09 | 280.21 | 30.40 |
| qcow2 | 2 | 4096 | memory\_block\_size | 2K | 578.93 | 623.20 | 596.58 | 14.84 |
| qcow2 | 4 | 2048 | memory\_block\_size | 1K | 262.69 | 305.88 | 288.26 | 15.59 |
| qcow2 | 4 | 2048 | memory\_block\_size | 2K | 544.14 | 619.51 | 596.51 | 26.71 |
| qcow2 | 4 | 4096 | memory\_block\_size | 1K | 256.58 | 304.58 | 279.20 | 16.47 |
| qcow2 | 4 | 4096 | memory\_block\_size | 2K | 489.43 | 581.06 | 548.51 | 33.44 |
| raw | 2 | 2048 | memory\_block\_size | 1K | 264.43 | 325.59 | 304.61 | 22.65 |
| raw | 2 | 2048 | memory\_block\_size | 2K | 476.25 | 517.83 | 500.66 | 14.89 |
| raw | 2 | 4096 | memory\_block\_size | 1K | 264.86 | 317.01 | 298.58 | 17.90 |
| raw | 2 | 4096 | memory\_block\_size | 2K | 602.10 | 636.69 | 621.05 | 12.15 |
| raw | 4 | 2048 | memory\_block\_size | 1K | 294.55 | 326.31 | 311.46 | 10.17 |
| raw | 4 | 2048 | memory\_block\_size | 2K | 512.68 | 611.46 | 574.17 | 41.79 |
| raw | 4 | 4096 | memory\_block\_size | 1K | 279.61 | 331.37 | 310.48 | 17.81 |
| raw | 4 | 4096 | memory\_block\_size | 2K | 491.18 | 581.71 | 549.09 | 32.96 |
| docker | 2 | 2048 | memory\_block\_size | 1K | 673.52 | 705.52 | 691.45 | 12.17 |
| docker | 2 | 2048 | memory\_block\_size | 2K | 1079.60 | 1108.23 | 1089.03 | 10.29 |
| docker | 2 | 4096 | memory\_block\_size | 1K | 677.46 | 691.94 | 685.97 | 5.08 |
| docker | 2 | 4096 | memory\_block\_size | 2K | 1090.06 | 1125.34 | 1111.77 | 13.17 |
| docker | 4 | 2048 | memory\_block\_size | 1K | 1278.02 | 1311.72 | 1292.68 | 11.41 |
| docker | 4 | 2048 | memory\_block\_size | 2K | 2031.13 | 2124.34 | 2081.09 | 31.48 |
| docker | 4 | 4096 | memory\_block\_size | 1K | 1256.07 | 1296.96 | 1273.12 | 16.17 |
| docker | 4 | 4096 | memory\_block\_size | 2K | 1856.05 | 2074.45 | 1997.53 | 77.39 |

**File I/O Test (MiB/Sec) - Random Read**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Image** | **CPU** | **RAM** | **Parameter** | **Value** | **min** | **max** | **avg** | **sd** |
| qcow2 | 2 | 2048 | file-total size | 1G | 2.83 | 9.47 | 4.84 | 2.44 |
| qcow2 | 2 | 2048 | file-total size | 2G | 4.93 | 11.65 | 7.51 | 2.23 |
| qcow2 | 2 | 4096 | file-total size | 1G | 6.12 | 6.67 | 6.33 | 0.17 |
| qcow2 | 2 | 4096 | file-total size | 2G | 5.92 | 7.12 | 6.56 | 0.37 |
| qcow2 | 4 | 2048 | file-total size | 1G | 6.21 | 10.57 | 7.18 | 1.69 |
| qcow2 | 4 | 2048 | file-total size | 2G | 5.16 | 6.22 | 5.59 | 0.33 |
| qcow2 | 4 | 4096 | file-total size | 1G | 4.92 | 8.19 | 6.31 | 1.09 |
| qcow2 | 4 | 4096 | file-total size | 2G | 5.37 | 7.27 | 6.21 | 0.66 |
| raw | 2 | 2048 | file-total size | 1G | 5.70 | 12.55 | 7.86 | 2.42 |
| raw | 2 | 2048 | file-total size | 2G | 4.25 | 11.90 | 6.23 | 2.85 |
| raw | 2 | 4096 | file-total size | 1G | 6.24 | 12.22 | 9.16 | 2.53 |
| raw | 2 | 4096 | file-total size | 2G | 5.94 | 13.22 | 9.91 | 3.18 |
| raw | 4 | 2048 | file-total size | 1G | 5.81 | 13.87 | 9.47 | 3.47 |
| raw | 4 | 2048 | file-total size | 2G | 5.07 | 11.30 | 7.98 | 2.55 |
| raw | 4 | 4096 | file-total size | 1G | 6.23 | 12.07 | 8.38 | 2.37 |
| raw | 4 | 4096 | file-total size | 2G | 6.41 | 14.82 | 9.78 | 3.12 |
| docker | 2 | 2048 | file-total size | 1G | 28.54 | 38.27 | 33.72 | 3.82 |
| docker | 2 | 2048 | file-total size | 2G | 26.76 | 38.09 | 34.12 | 4.02 |
| docker | 2 | 4096 | file-total size | 1G | 34.25 | 41.93 | 37.66 | 3.01 |
| docker | 2 | 4096 | file-total size | 2G | 29.81 | 40.44 | 35.62 | 3.46 |
| docker | 4 | 2048 | file-total size | 1G | 35.33 | 46.52 | 40.16 | 3.98 |
| docker | 4 | 2048 | file-total size | 2G | 34.92 | 41.37 | 38.31 | 2.24 |
| docker | 4 | 4096 | file-total size | 1G | 12.21 | 16.00 | 14.09 | 1.57 |
| docker | 4 | 4096 | file-total size | 2G | 11.73 | 24.52 | 16.63 | 5.48 |

**File I/O Test (MiB/Sec) - Random Write**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Image** | **CPU** | **RAM** | **Parameter** | **Value** | **min** | **max** | **avg** | **sd** |
| qcow2 | 2 | 2048 | file-total size | 1G | 1.89 | 6.31 | 3.23 | 1.62 |
| qcow2 | 2 | 2048 | file-total size | 2G | 3.29 | 7.77 | 5.01 | 1.48 |
| qcow2 | 2 | 4096 | file-total size | 1G | 4.08 | 4.45 | 4.22 | 0.1 |
| qcow2 | 2 | 4096 | file-total size | 2G | 3.95 | 4.75 | 4.37 | 0.24 |
| qcow2 | 4 | 2048 | file-total size | 1G | 4.14 | 7.05 | 4.79 | 1.12 |
| qcow2 | 4 | 2048 | file-total size | 2G | 3.44 | 4.14 | 3.72 | 0.20 |
| qcow2 | 4 | 4096 | file-total size | 1G | 3.28 | 5.46 | 4.20 | 0.72 |
| qcow2 | 4 | 4096 | file-total size | 2G | 3.58 | 4.85 | 4.14 | 0.43 |
| raw | 2 | 2048 | file-total size | 1G | 3.80 | 8.37 | 5.24 | 1.61 |
| raw | 2 | 2048 | file-total size | 2G | 2.83 | 7.94 | 4.15 | 1.90 |
| raw | 2 | 4096 | file-total size | 1G | 4.16 | 8.15 | 6.10 | 1.69 |
| raw | 2 | 4096 | file-total size | 2G | 3.96 | 8.82 | 6.60 | 2.12 |
| raw | 4 | 2048 | file-total size | 1G | 3.88 | 9.25 | 6.31 | 2.31 |
| raw | 4 | 2048 | file-total size | 2G | 3.38 | 7.54 | 5.32 | 1.70 |
| raw | 4 | 4096 | file-total size | 1G | 4.16 | 8.05 | 5.59 | 1.58 |
| raw | 4 | 4096 | file-total size | 2G | 4.27 | 9.88 | 6.52 | 2.08 |
| docker | 2 | 2048 | file-total size | 1G | 19.03 | 25.52 | 22.48 | 2.55 |
| docker | 2 | 2048 | file-total size | 2G | 17.84 | 25.40 | 22.74 | 2.68 |
| docker | 2 | 4096 | file-total size | 1G | 22.83 | 27.96 | 25.10 | 2.00 |
| docker | 2 | 4096 | file-total size | 2G | 19.87 | 26.96 | 23.74 | 2.31 |
| docker | 4 | 2048 | file-total size | 1G | 23.56 | 31.01 | 26.77 | 2.65 |
| docker | 4 | 2048 | file-total size | 2G | 23.28 | 27.58 | 25.54 | 1.50 |
| docker | 4 | 4096 | file-total size | 1G | 8.14 | 10.67 | 9.40 | 1.04 |
| docker | 4 | 4096 | file-total size | 2G | 7.82 | 16.34 | 11.08 | 3.65 |

**Performance Analysis**

**CPU test**

The metric I use to assess CPU performance is throughput (events/s). The two test cases I run are cpu-max-prime 20000 and cpu-max-prime 80000. Essentially, the test translates to: looping over numbers and checking if they are divisible only by themselves. The higher the number, the lower the throughput.

For qemu images, I expect the performance of the raw image to be better than that of the qcow2 image due to the additional overhead associated with qcow2, which is consistent with the data I observed. We do see that the performance of the raw image is slightly better than that of qcow2. The change in RAM doesn’t significantly impact performance, which is expected. Interestingly, for qemu images, changes in CPU also don’t affect performance much, whereas for Docker, we observe that performance nearly doubles if the CPU is doubled.

My analysis of this observation is that I set the CPU as a percentage for Docker, so Docker limits the hardware based on this percentage, while qemu limits the hardware based on CPU numbers. For a powerful, modern CPU, the sysbench CPU test workload is not heavy enough to create a noticeable difference between different CPU numbers. To discern a difference, I should modify the thread numbers.

**Memory test**

The metric I use to assess memory performance is throughput (MiB/sec). The two test cases I run are with memory-block-sizes of 1KB and 2KB. The benchmark application allocates a memory buffer and then reads or writes from it, each time for the size of a pointer, and each execution is for the total size of the memory volume. The default operation is writing.

We notice that for all images, doubling the memory-block-size nearly doubles the memory write speed, which is in line with our expectations. Changes in the hardware CPU and RAM storage should not affect the memory write performance, since the memory buffer test size is much smaller than the RAM storage limit. However, we do expect different images to exhibit varying memory speeds due to overheads. I anticipate Docker to be the fastest, followed by raw, and then qcow2 being the slowest. From my data, I observe that Docker performs the best, as expected, while the performances of raw and qcow2 are quite close. The memory test shows a large standard deviation; the overhead difference between raw and qcw2 doesn’t seem significant.

**File I/O test**

The metric I use to assess file I/O performance is throughput (MiB/sec). The two test cases I run involve file-total-sizes of 1GB and 2GB, with operations comprising a combination of random reads and writes. The tests are conducted at a ratio of 80% reads to 20% writes.

From my observations, changes in the hardware CPU and RAM do not significantly affect the I/O performance, similar to the file total size. However, we do observe a performance difference between different images. As expected, the Docker image exhibits the best I/O performance, followed by the raw image, and then the qcow2 image.