Homework Wan Huzaifah bin Wan Azhar

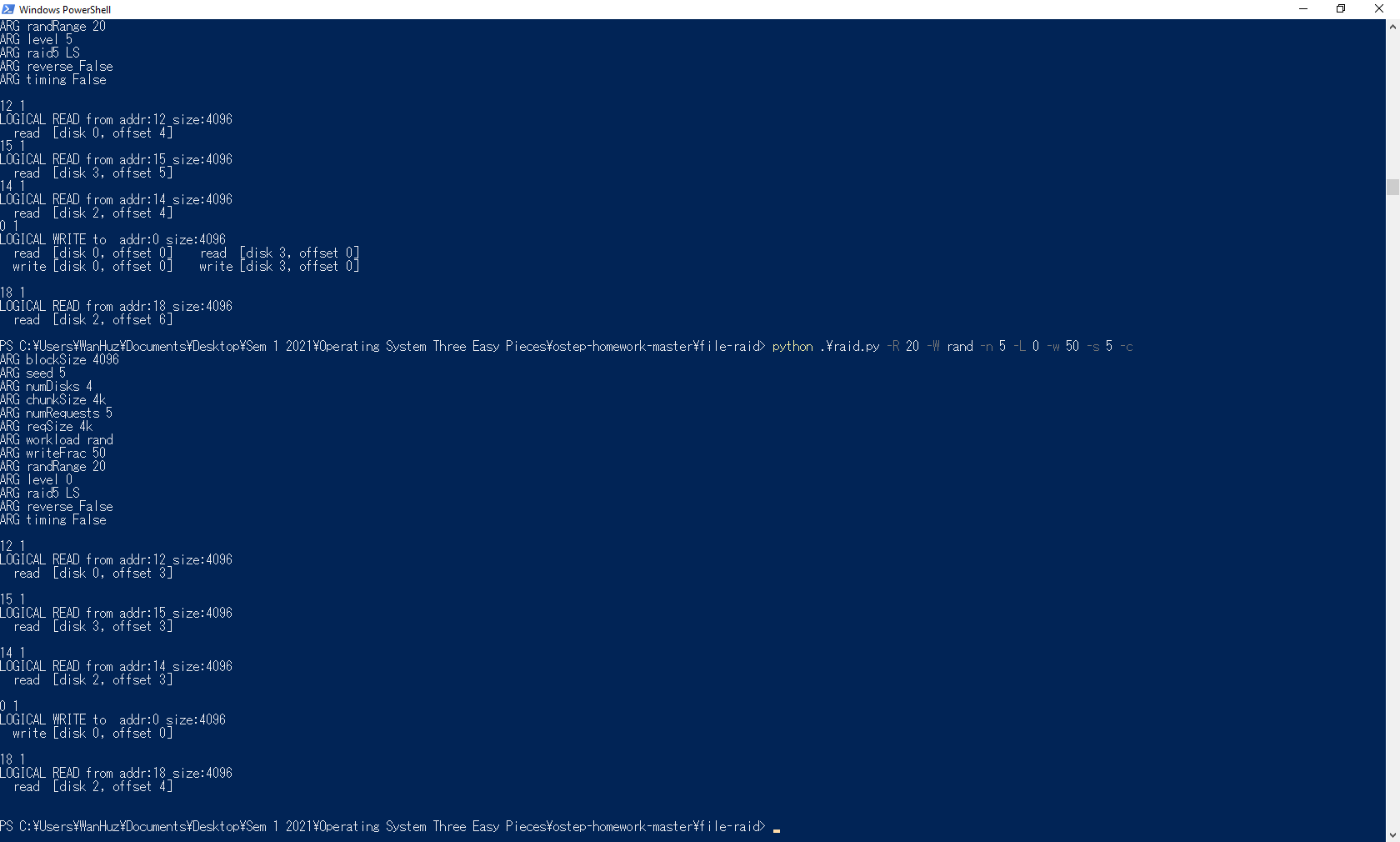
Answer:

Redundant Array of Inexpensive Disk (RAID)



Level 0 Random:

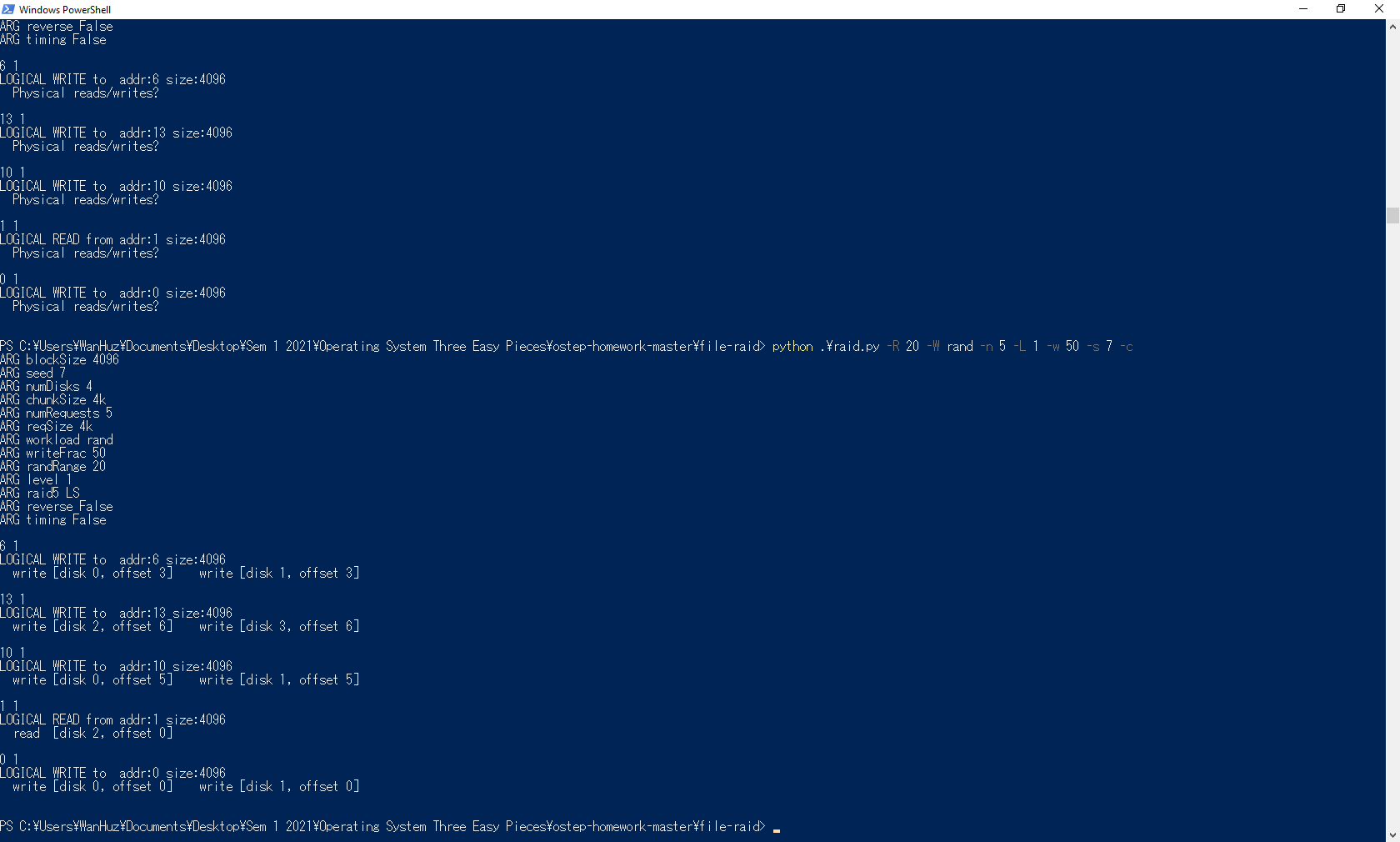
python .\raid.py -R 20 -W rand -n 5 -L 0 -w 50 -s 5 -c



1. Read [disk 0, offset 3]
2. Read [disk 3, offset 3]
3. Read [disk 2, offset 3]
4. Write [disk 0, offset 0]
5. Read [disk 2, offset 4]

Level 1 Random:

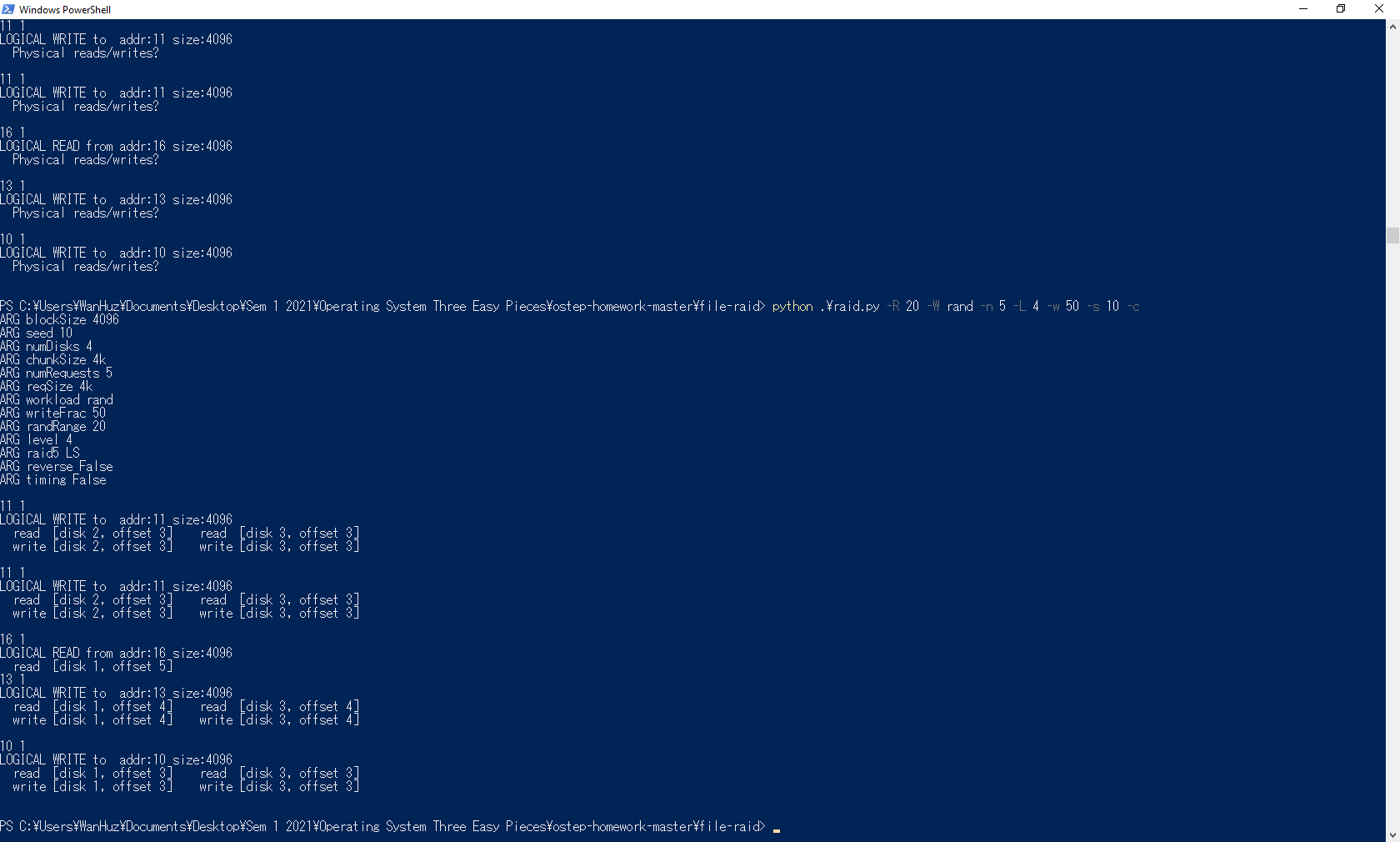
python .\raid.py -R 20 -W rand -n 5 -L 1 -w 50 -s 7 -c



1. write [disk 0, offset 3] & write [disk 1, offset 3]
2. write [disk 2, offset 6] & write [disk 3, offset 6]
3. write [disk 0, offset 5] & write [disk 1, offset 5]
4. read [disk 2, offset 0]
5. write [disk 0, offset 0] & write [disk 1, offset 0]

Level 4 Random:

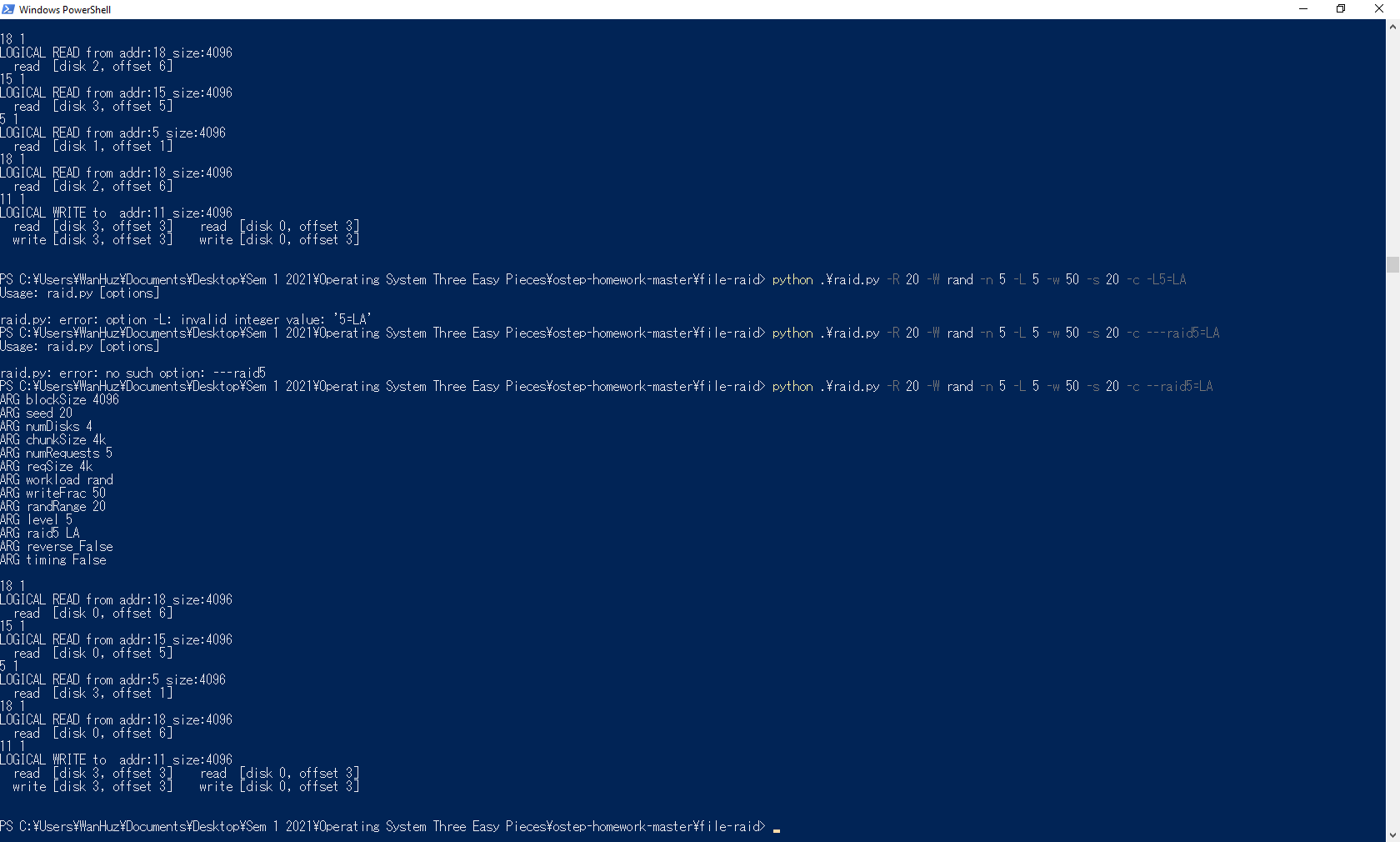
python .\raid.py -R 20 -W rand -n 5 -L 4 -w 50 -s 10 -c



1. read [disk 2, offset 3], write [disk 2, offset 3], read [disk 3, offset 3], write [disk 3, offset 3]
2. read [disk 2, offset 3], write [disk 2, offset 3], read [disk 3, offset 3], write [disk 3, offset 3]
3. read [disk 1, offset 5]
4. read [disk 1, offset 4], write [disk 1, offset 4], read [disk 3, offset 4], write [disk 3, offset 4]
5. read [disk 1, offset 3], write [disk 1, offset 3], read [disk 3, offset 3], write [disk 3, offset 3]

Level 5 Asymmetrical Random:

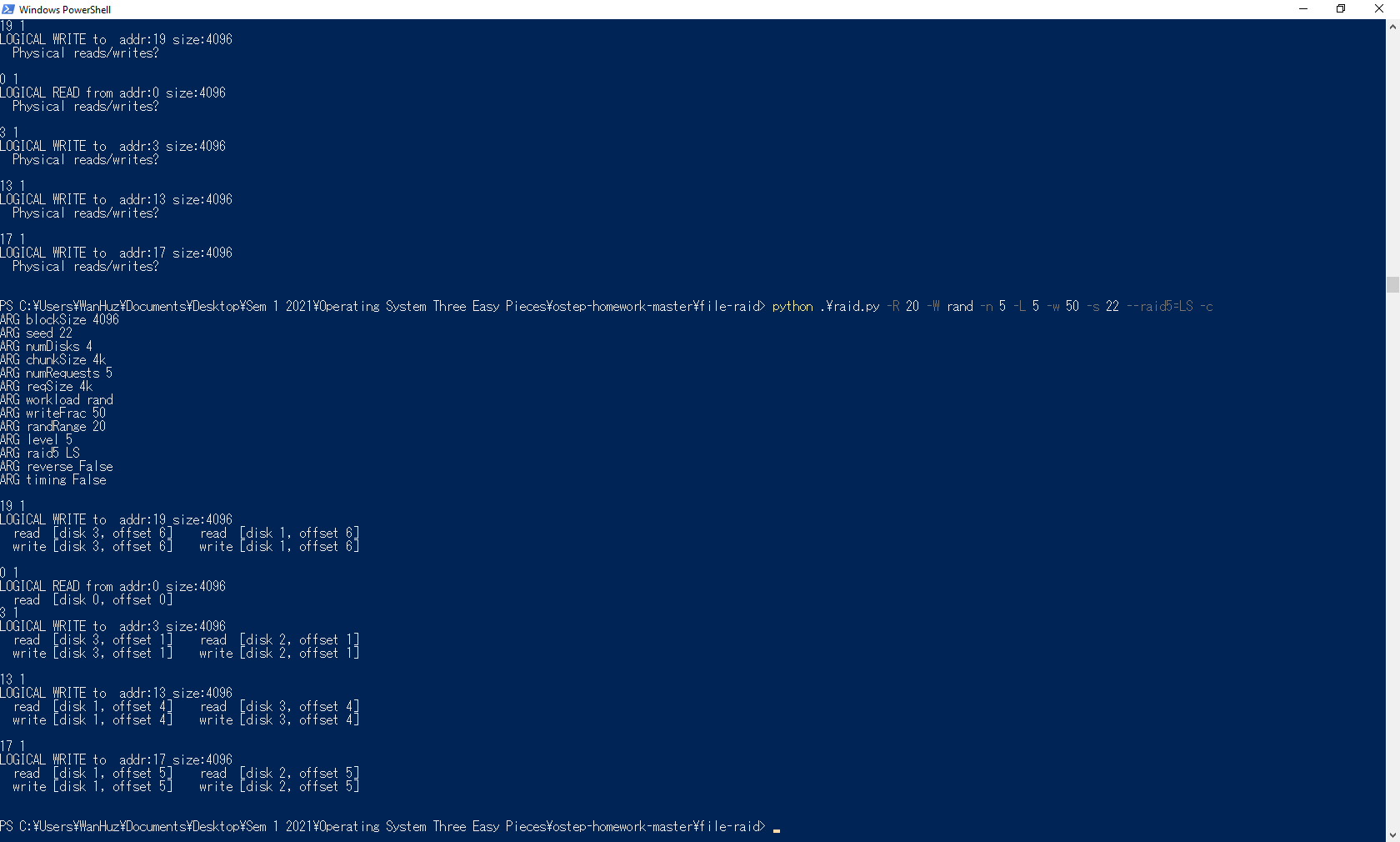
python .\raid.py -R 20 -W rand -n 5 -L 5 -w 50 -s 20 -c --raid5=LA



1. read [disk 0, offset 6]
2. read [disk 0, offset 5]
3. read [disk 3, offset 1]
4. read [disk 0, offset 6]
5. read [disk 3, offset 3], write [disk 3, offset 3], read [disk 0, offset 3], write [disk 0, offset 3]

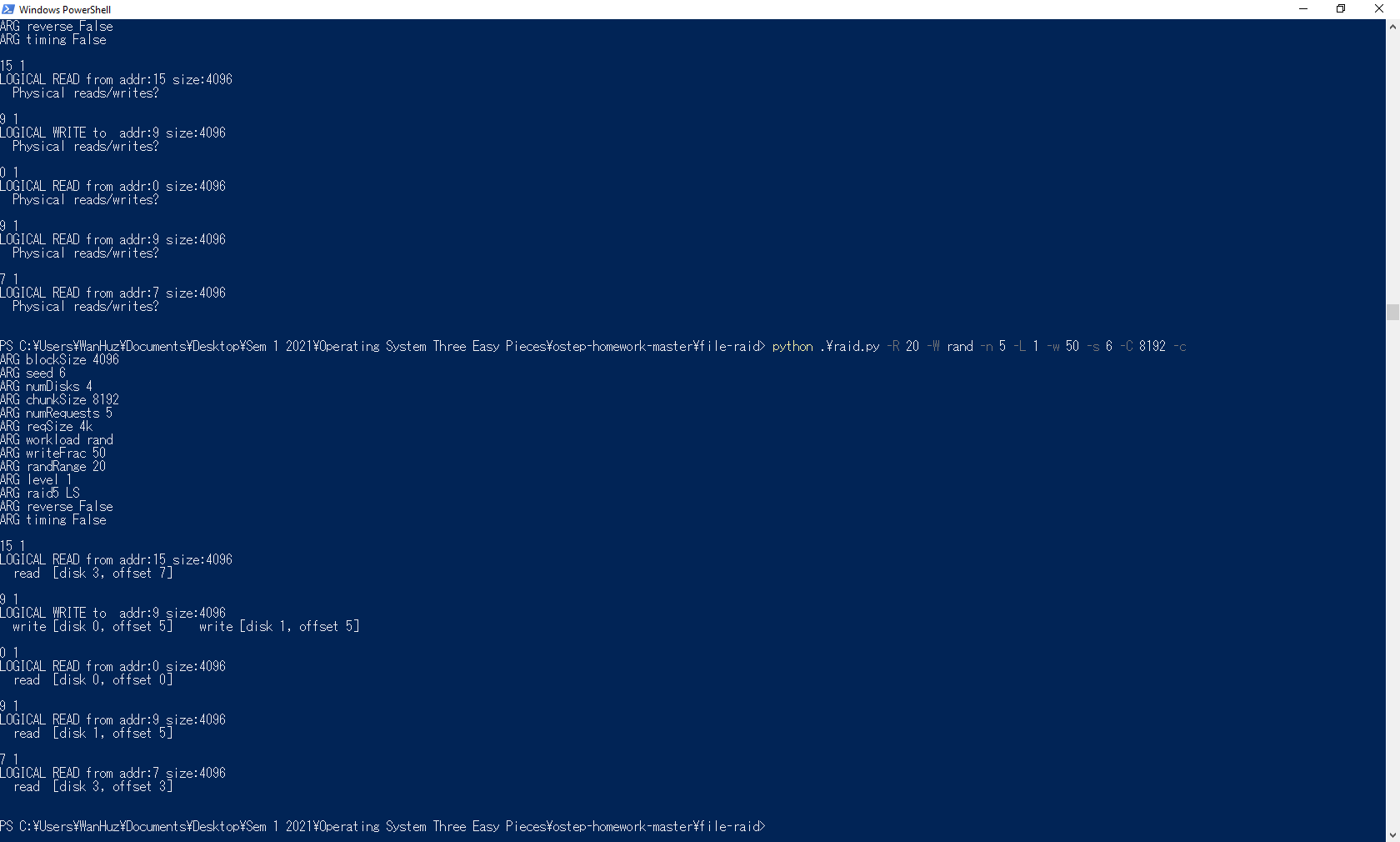
Level 5 Symmetrical Random:

python .\raid.py -R 20 -W rand -n 5 -L 5 -w 50 -s 22 --raid5=LS -c



1. read [disk 3, offset 6], write [disk 3, offset 6], read [disk 1, offset 6], write [disk 1, offset 6]
2. read [disk 0, offset 0]
3. read [disk 3, offset 1], write [disk 3, offset 1], read [disk 2, offset 1], write [disk 2, offset 1]
4. read [disk 1, offset 4], write [disk 1, offset 4], read [disk 3, offset 4], write [disk 3, offset 4]
5. read [disk 1, offset 5], write [disk 1, offset 5], read [disk 2, offset 5], write [disk 2, offset 5]
6. Level 1 only with 8192 Chunk

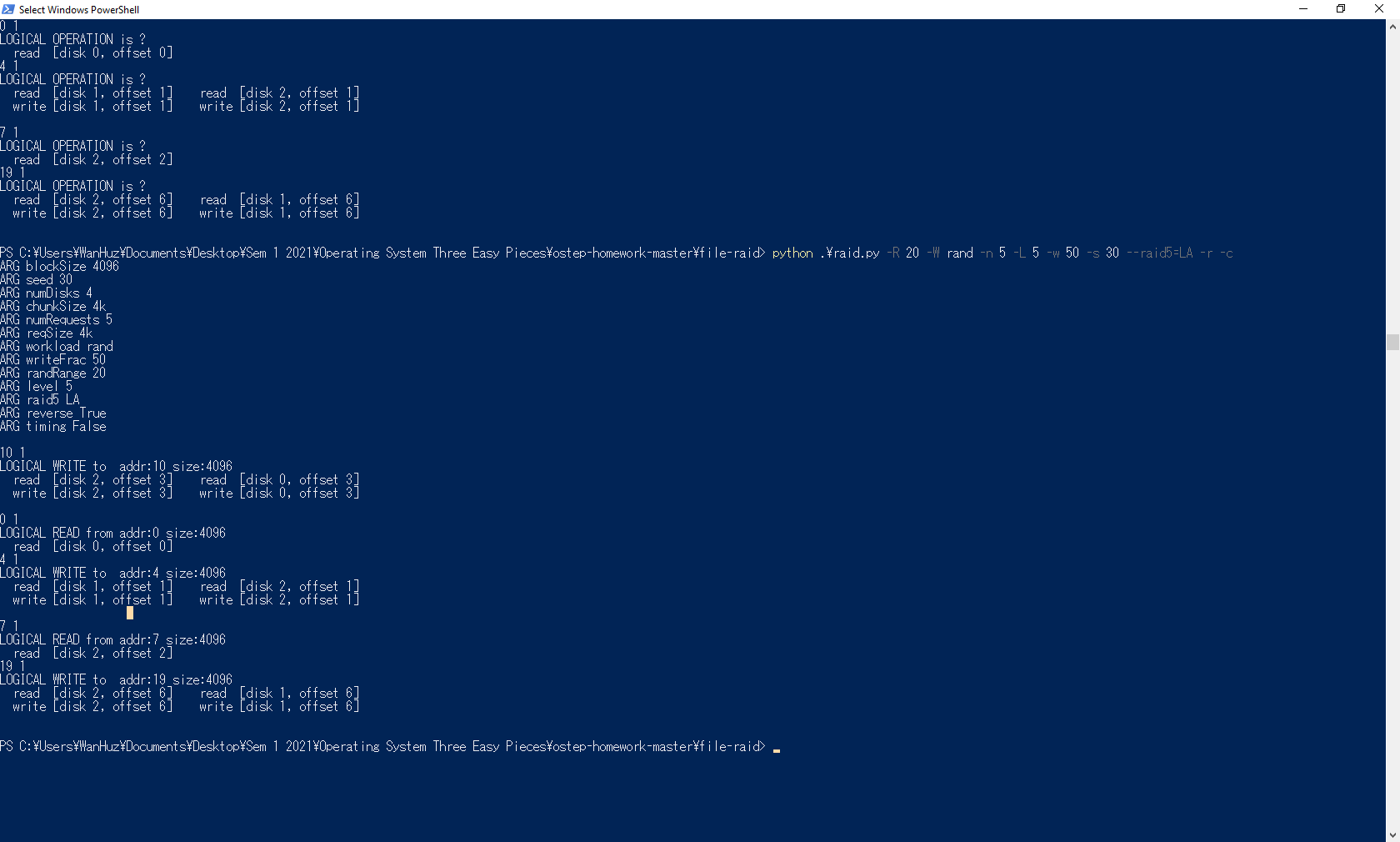
python .\raid.py -R 20 -W rand -n 5 -L 1 -w 50 -s 6 -C 8192



1. read [disk 3, offset 7]
2. write [disk 0, offset 5], write [disk 1, offset 5]
3. read [disk 0, offset 0]
4. read [disk 0, offset 5]
5. read [disk 3, offset 3]

* Changing chunk size will double the offset

1. Level 5 (Asymm.) in Reverse



1. Write to addr 10
2. Read from addr 0
3. Write to addr 4
4. Read from addr 7
5. Write to addr 19

* When the size of the request increases, the workload becomes more sequential than random, which will increase the performance.
* Request size of 16k will make I/O most efficient for RAID4 and RAID5
  + This is because each write request require a minimum 2 I/O operation (to calculate/update parity)
  + With size of 16k, more write can be done (as it will be sequential) and only minimum 2 I/O operation are required to update parity.



python .\raid.py -R 20 -n 100 -L 0 -w 0 -c -t -W rand

python .\raid.py -R 20 -n 100 -L 1 -w 0 -c -t -W rand

python .\raid.py -R 20 -n 100 -L 4 -w 0 -c -t -W rand

python .\raid.py -R 20 -n 100 -L 5 -w 0 -c -t -W rand

Output:

Total time: 16.5 for Raid-0

Total time: 22.899… for Raid-1

Total time: 18.199… for Raid-4

Total time: 18.6 for Raid-5



* For every RAID level, each increase of disk by power of two decrease the total amount of time by 3 / 4 or around 75%
* In general, the performance of each RAID level increased when the number of disks increasing.
* This is because more disk means that the operating system can take advantage of parallelism to do more I/O operation concurrently.



python .\raid.py -R 20 -n 100 -L 0 -w 100 -c -t -W rand

python .\raid.py -R 20 -n 100 -L 1 -w 100 -c -t -W rand

python .\raid.py -R 20 -n 100 -L 4 -w 100 -c -t -W rand

python .\raid.py -R 20 -n 100 -L 5 -w 100 -c -t -W rand

* For RAID-0, each increase in disk will decrease the total time to finish by 75% or same as read.
* For RAID-1 and RAID-4, however, every increase in disk will decrease the total time to finish by log2, as in for the first increase in disk, there will be rapid decrease in total time, however, subsequent increase in disk will slowly decrease the total time to finish.
* For RAID-5, the increase of the disk by power of two drastically decrease the total time by half, however, subsequent increase of disk only decreases the total time only a little.
* The estimate time it took to finish workload of 100 random write will be around the range of 10.0 to 25.0 depending on disk number, RAID-Level and et cetera.



* Performance of RAID level of 100 workloads sequential reads:
  + RAID-0: 12.4999… The fastest
  + RAID-1: 14.8999… The slowest
  + RAID-4: 13.3999…
  + RAID-5: 13.2999…
* Performance of RAID level of 100 workloads sequential writes:
  + RAID-0: 12.4999… The fastest
  + RAID-1: 14.999… The slowest
  + RAID-4: 13.3999…
  + RAID-5: 13.3999…
* Overall, performance difference of 100 sequential reads and writes only differs slightly by RAID level.
* This truly shows how sequential workload matters in increasing performance.
* Performance of RAID level of 100 workloads sequential reads of 16k size:
  + RAID-0: 20.00
  + RAID-1: 29.8999…
  + RAID-4: 23.4
  + RAID-5: 23.3
* When using RAID-4 or RAID-5, the best workload size 12k as the RAID can perform full-stripe to increase performance.