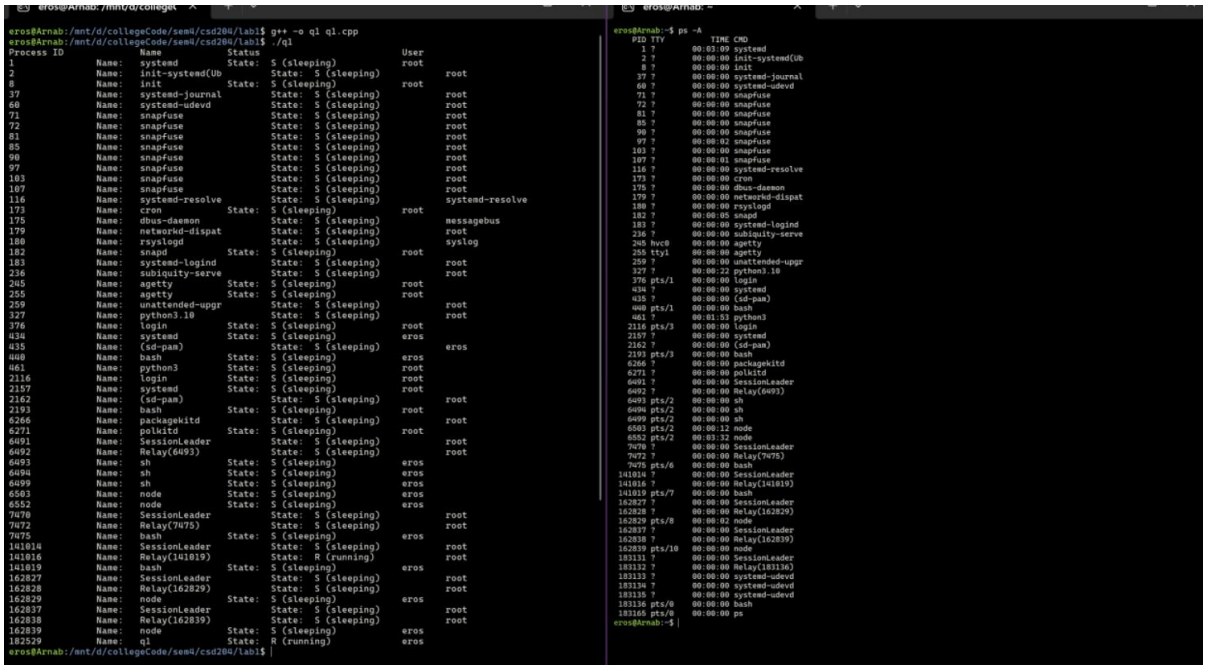
Q1.)

We observe that the written program gives us an additional output in the form of if the process is running or sleeping, and who is the user of the process (in this case, either the root or Eros). However, the “ps -A” command returns the time of the process as well. Additionally, the PID of the processes may have changed between the command and the compilation of the process.
The written program gives us a more comprehensive overview of the current programs.



Q2.)
eros@Arnab:/mnt/d/collegeCode/sem4/csd204/lab1/Assgn1-ArnabMandal$ ./q2
Config of the system:
OS: Linux
Release: 5.15.133.1-microsoft-standard-WSL2
Version: #1 SMP Thu Oct 5 21:02:42 UTC 2023
Machine: x86_64
eros@Arnab:/mnt/d/collegeCode/sem4/csd204/lab1/Assgn1-ArnabMandal$

Q3.)

a) The processor or the CPU, is the overall chip that contains cores and threads, to execute tasks using its cores. Example:
On-line CPU(s) list: 0-19
Vendor ID: GenuineIntel
Model name: 13th Gen Intel(R) Core(TM) i9-13900H
CPU family: 6

Model: 186

Thread(s) per core: 2

Core(s) per socket: 10

The cores are present in the CPU, with each core being a processing unit within the processor that handles individual tasks independently and enables parallel task execution.

b) 10 core(s) per socket * 1 socket = 10 cores

c) 0-19 procesors, a total of 20 CPU(s)

d) 2995.211 Mhz

e) Architecture: x86_64

CPU op-mode(s): 32-bit, 64-bit

Address sizes: 46 bits physical, 48 bits virtual

Byte Order: Little Endian

f) MemTotal: 7944564 kB

g) MemFree: 6390432 kB

h) Context switches: 1496748

Forks: 13641

Q4.)

a) 1602 is the PID

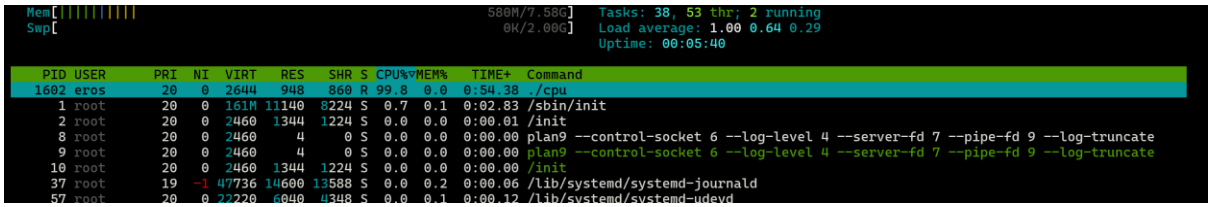
b) PID: 1602 CMD

%CPU: 102

%MEM: 0.0

CMD: ./cpu

Or



c)

```
eros@Arnab:~$ ps -o pid,state,cmd -C cpu
    PID S CMD
   1602 R ./cpu
eros@Arnab:~$
```

It is in the running state

```
top - 13:42:33 up 6 min,  1 user,  load average: 1.00, 0.72, 0.34
Tasks:   1 total,   1 running,   0 sleeping,   0 stopped,   0 zombie
%Cpu(s):  5.0 us,  0.0 sy,  0.0 ni, 95.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :   7758.4 total,   6740.4 free,    587.2 used,    430.8 buff/cache
MiB Swap:   2048.0 total,   2048.0 free,      0.0 used.   6941.6 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
   1602 eros      20   0    2644    948    860 R 100.0   0.0   2:04.11 cpu
```

Q5.)

a) 16558

b) 15522

Parent: 15520

ParentL15519

Parent: 2

Parent: 1

Parent: 0

c) If the file descriptor is 0, its either disconnected or redirected to /dev/null to be in the background

If it is 1, it points to /tmp/tmp.txt where output will be captured

If it is 2, it stays connected to terminal to be debugged

d) eros@Arnab:/mnt/d/collegeCode/sem4/csd204/lab1/Assgn1-ArnabMandal$ ls -l /proc/16433/fd

total 0

lrwx------ 1 eros eros 64 Jan 26 09:44 0 -> /dev/pts/0

lrwx------ 1 eros eros 64 Jan 26 09:44 1 -> /dev/pts/0

lrwx------ 1 eros eros 64 Jan 26 09:44 2 -> /dev/pts/0

eros@Arnab:/mnt/d/collegeCode/sem4/csd204/lab1/Assgn1-ArnabMandal$

Basically when the shell creates a pipe, the shell redirects the standard output of the first process aka cpu-print into the pipe, and redirects the second process aka grep hello to read from the pipe, enabling processes to communicate with each other.

e) Cd and history are build in commands

Ls and ps are exernal executables

Q6.)

```
   PID    VSZ   RSS CMD
  1534   6560  4732 ./mem1
    PID    VSZ   RSS CMD
  1794   6560  4884 ./mem2
```

Expected was that mem1 would have higher RSS due to array being actively accessed. The

difference is due to lazy allocation, i.e. the system allots ram before it even used.
Q7.) One of the programs randomly opens files among the ones created, while the other repeatedly opens foo0 In a loop. The disk utilization is highest during the first run, but then it reduces because it is cached into memory. Disk1 has a higher disk throughput while disk has random access causing frequent disk seeks.