TASK 3

Part1

1. How many states could has a process in Linux?

'D' = Uninterruptible Sleep

'R' = Running or Runnable

'S' = Interruptable Sleep

'T' = Stopped

'Z' = Zombie

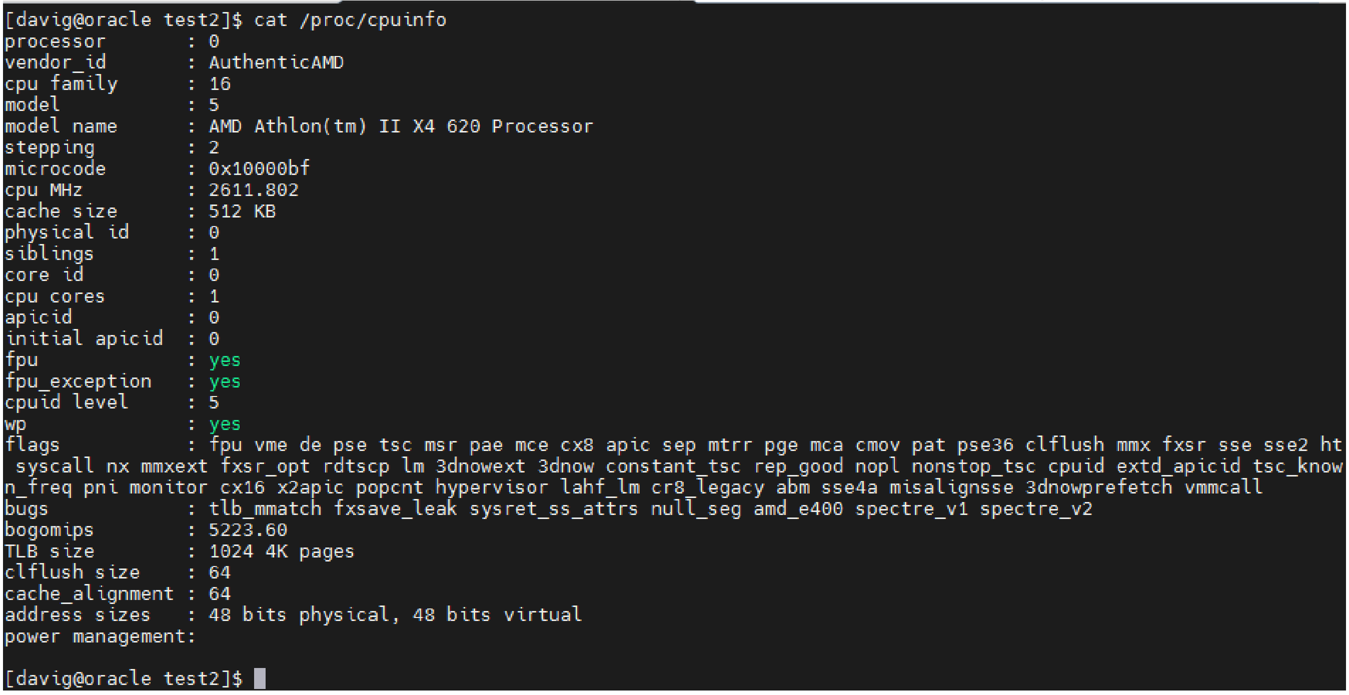
2. Examine the pstree command. Make output (highlight) the chain (ancestors) of the current process.

pstree -h

3. What is a proc file system?

Proc file system contains useful information about the processes that are currently running. The proc file system also provides communication medium between kernel space and user space.

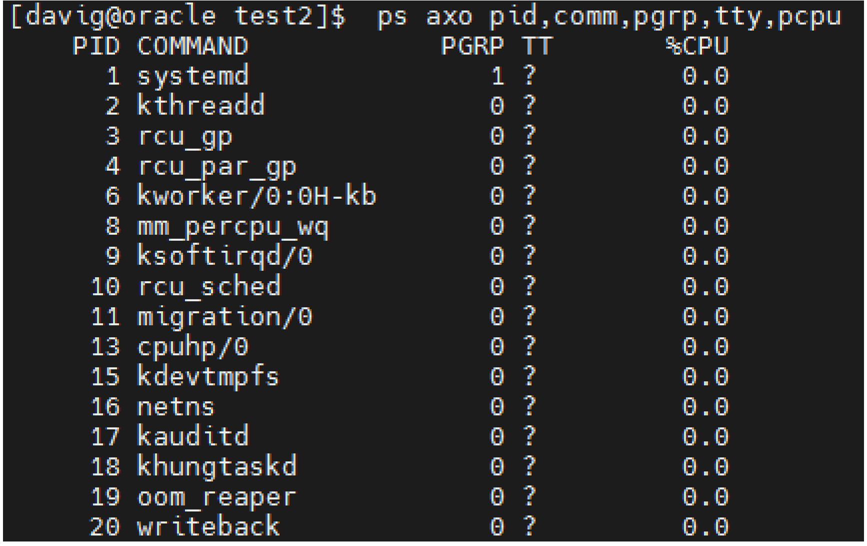
4. Print information about the processor (its type, supported technologies, etc.).



```
[davig@oracle test2]$ cat /proc/cpuinfo
processor       : 0
vendor_id       : AuthenticAMD
cpu family      : 16
model           : 5
model name      : AMD Athlon(tm) II X4 620 Processor
stepping        : 2
microcode       : 0x10000bf
cpu MHz         : 2611.802
cache size      : 512 KB
physical id     : 0
siblings        : 1
core id         : 0
cpu cores       : 1
apicid          : 0
initial apicid  : 0
fpu             : yes
fpu_exception   : yes
cpuid level     : 5
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht
 syscall nx mmxext fxsr_opt rdtscp lm 3dnowext 3dnow constant_tsc rep_good nopl nonstop_tsc cpuid extd_apicid tsc_know
n_freq pni monitor cx16 x2apic popcnt hypervisor lahf_lm cr8_legacy abm sse4a misalignsse 3dnowprefetch vmmcall
bugs            : tlb_mmatch fxsave_leak sysret_ss_attrs null_seg amd_e400 spectre_v1 spectre_v2
bogomips        : 5223.60
TLB size        : 1024 4K pages
clflush size    : 64
cache_alignment : 64
address sizes   : 48 bits physical, 48 bits virtual
power management:

[davig@oracle test2]$
```

5. Use the ps command to get information about the process. The information should be as follows: the owner of the process, the arguments with which the process was launched for execution, the group owner of this process, etc.

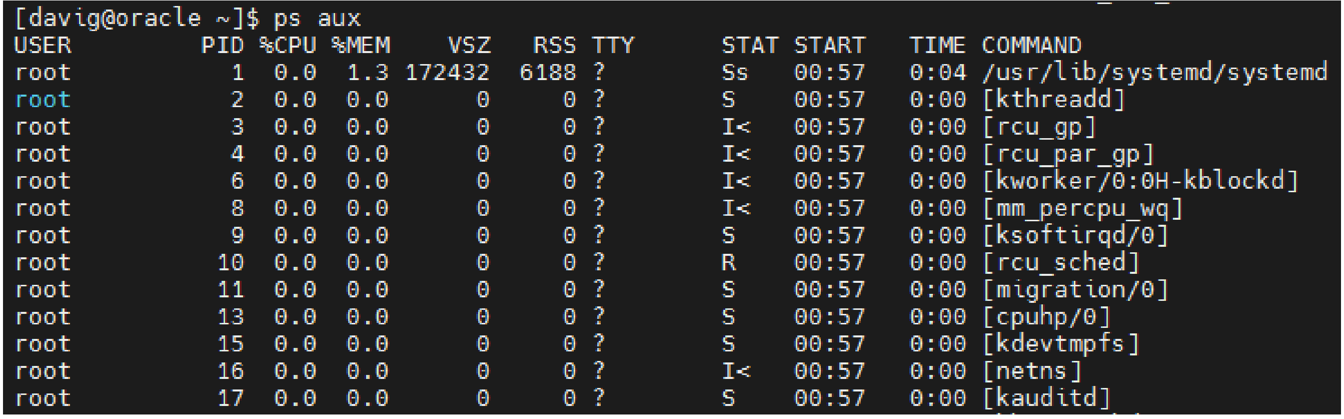


```
[davig@oracle test2]$  ps axo pid,comm,pgrp,tty,pcpu
    PID COMMAND            PGRP TT       %CPU
      1 systemd               1 ?         0.0
      2 kthreadd              0 ?         0.0
      3 rcu_gp                0 ?         0.0
      4 rcu_par_gp            0 ?         0.0
      6 kworker/0:0H-kb       0 ?         0.0
      8 mm_percpu_wq          0 ?         0.0
      9 ksoftirqd/0           0 ?         0.0
     10 rcu_sched             0 ?         0.0
     11 migration/0           0 ?         0.0
     13 cpuhp/0               0 ?         0.0
     15 kdevtmpfs             0 ?         0.0
     16 netns                 0 ?         0.0
     17 kauditd               0 ?         0.0
     18 khungtaskd            0 ?         0.0
     19 oom_reaper            0 ?         0.0
     20 writeback             0 ?         0.0
```

6. How to define kernel processes and user processes?

When managing processes, it is easy to recognize the kernel processes because they have a name that is between square brackets.

7. Print the list of processes to the terminal. Briefly describe the statuses of the processes. What condition are they in, or can they be arriving in?



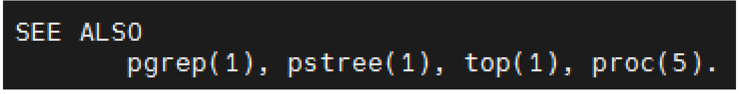
```
[davig@oracle ~]$ ps aux
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  1.3 172432  6188 ?        Ss   00:57   0:04 /usr/lib/systemd/systemd
root           2  0.0  0.0      0     0 ?        S    00:57   0:00 [kthreadd]
root           3  0.0  0.0      0     0 ?        I<   00:57   0:00 [rcu_gp]
root           4  0.0  0.0      0     0 ?        I<   00:57   0:00 [rcu_par_gp]
root           6  0.0  0.0      0     0 ?        I<   00:57   0:00 [kworker/0:0H-kblockd]
root           8  0.0  0.0      0     0 ?        I<   00:57   0:00 [mm_percpu_wq]
root           9  0.0  0.0      0     0 ?        S    00:57   0:00 [ksoftirqd/0]
root          10  0.0  0.0      0     0 ?        R    00:57   0:00 [rcu_sched]
root          11  0.0  0.0      0     0 ?        S    00:57   0:00 [migration/0]
root          13  0.0  0.0      0     0 ?        S    00:57   0:00 [cpuhp/0]
root          15  0.0  0.0      0     0 ?        S    00:57   0:00 [kdevtmpfs]
root          16  0.0  0.0      0     0 ?        I<   00:57   0:00 [netns]
root          17  0.0  0.0      0     0 ?        S    00:57   0:00 [kauditd]
```

Stats: S – sleep, I – idle, R – running.

8. Display only the processes of a specific user.

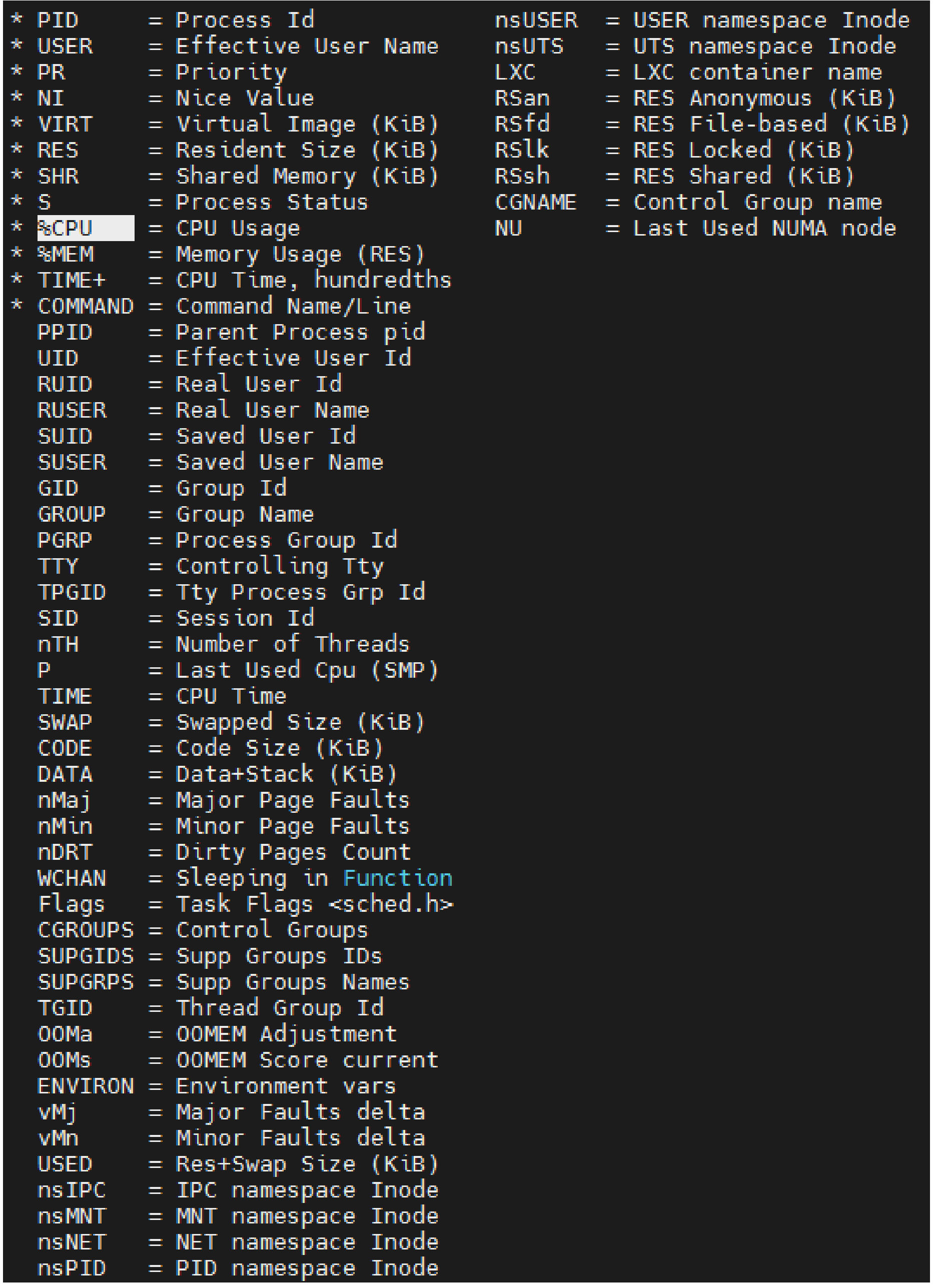
ps –u username

9. What utilities can be used to analyze existing running tasks (by analyzing the help for the ps command)?



```
SEE ALSO
       pgrep(1), pstree(1), top(1), proc(5).
```

10. What information does top command display?



```
* PID     = Process Id              nsUSER  = USER namespace Inode
* USER    = Effective User Name     nsUTS   = UTS namespace Inode
* PR      = Priority                LXC     = LXC container name
* NI      = Nice Value              RSan    = RES Anonymous (KiB)
* VIRT    = Virtual Image (KiB)     RSfd    = RES File-based (KiB)
* RES     = Resident Size (KiB)     RSlk    = RES Locked (KiB)
* SHR     = Shared Memory (KiB)     RSsh    = RES Shared (KiB)
* S       = Process Status          CGNAME  = Control Group name
* %CPU    = CPU Usage               NU      = Last Used NUMA node
* %MEM    = Memory Usage (RES)
* TIME+   = CPU Time, hundredths
* COMMAND = Command Name/Line
  PPID    = Parent Process pid
  UID     = Effective User Id
  RUID    = Real User Id
  RUSER   = Real User Name
  SUID    = Saved User Id
  SUSER   = Saved User Name
  GID     = Group Id
  GROUP   = Group Name
  PGRP    = Process Group Id
  TTY     = Controlling Tty
  TPGID   = Tty Process Grp Id
  SID     = Session Id
  nTH     = Number of Threads
  P       = Last Used Cpu (SMP)
  TIME    = CPU Time
  SWAP    = Swapped Size (KiB)
  CODE    = Code Size (KiB)
  DATA    = Data+Stack (KiB)
  nMaj    = Major Page Faults
  nMin    = Minor Page Faults
  nDRT    = Dirty Pages Count
  WCHAN   = Sleeping in Function
  Flags   = Task Flags <sched.h>
  CGROUPS = Control Groups
  SUPGIDS = Supp Groups IDs
  SUPGRPS = Supp Groups Names
  TGID    = Thread Group Id
  OOMa    = OOMEM Adjustment
  OOMs    = OOMEM Score current
  ENVIRON = Environment vars
  vMj     = Major Faults delta
  vMn     = Minor Faults delta
  USED    = Res+Swap Size (KiB)
  nsIPC   = IPC namespace Inode
  nsMNT   = MNT namespace Inode
  nsNET   = NET namespace Inode
  nsPID   = PID namespace Inode
```

12. Display the processes of the specific user using the top command.

top –u username

13. What interactive commands can be used to control the top command? Give a couple of examples.

k – kill a task, q –quit, r –renice a task.

14. Sort the contents of the processes window using various parameters (for example, the amount of processor time taken up, etc.)

T – sort by TIME+, M – sort by %MEM, N – sort by PID.

15. Concept of priority, what commands are used to set priority?

nice

renice

16. Can I change the priority of a process using the top command? If so, how?

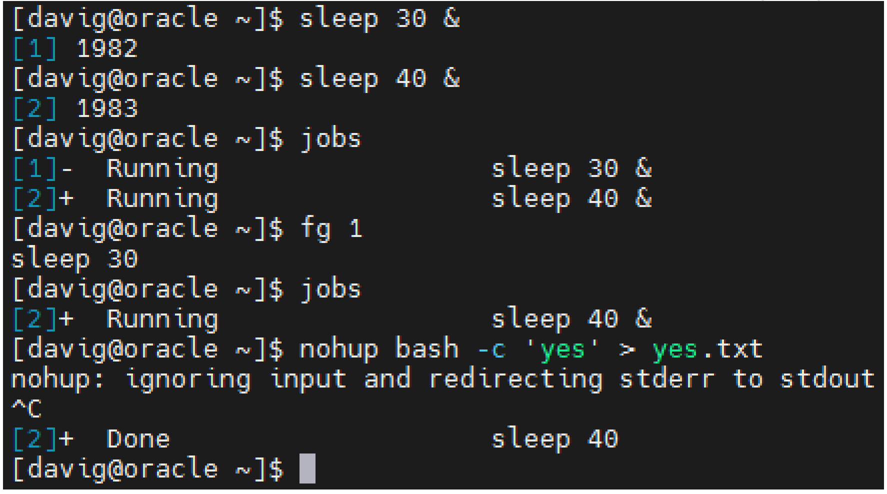
Type r, enter PID, enter value.

17. Examine the kill command. How to send with the kill command process control signal? Give an example of commonly used signals.

kill –s signalname PID

24 – SIGSTOP, 26 – SIGCOUNT, 9 – SIGKILL.

18. Commands jobs, fg, bg, nohup. What are they for? Use the sleep, yes command to demonstrate the process control mechanism with fg, bg.



```
[davig@oracle ~]$ sleep 30 &
[1] 1982
[davig@oracle ~]$ sleep 40 &
[2] 1983
[davig@oracle ~]$ jobs
[1]-  Running                 sleep 30 &
[2]+  Running                 sleep 40 &
[davig@oracle ~]$ fg 1
sleep 30
[davig@oracle ~]$ jobs
[2]+  Running                 sleep 40 &
[davig@oracle ~]$ nohup bash -c 'yes' > yes.txt
nohup: ignoring input and redirecting stderr to stdout
^C
[2]+  Done                    sleep 40
[davig@oracle ~]$
```