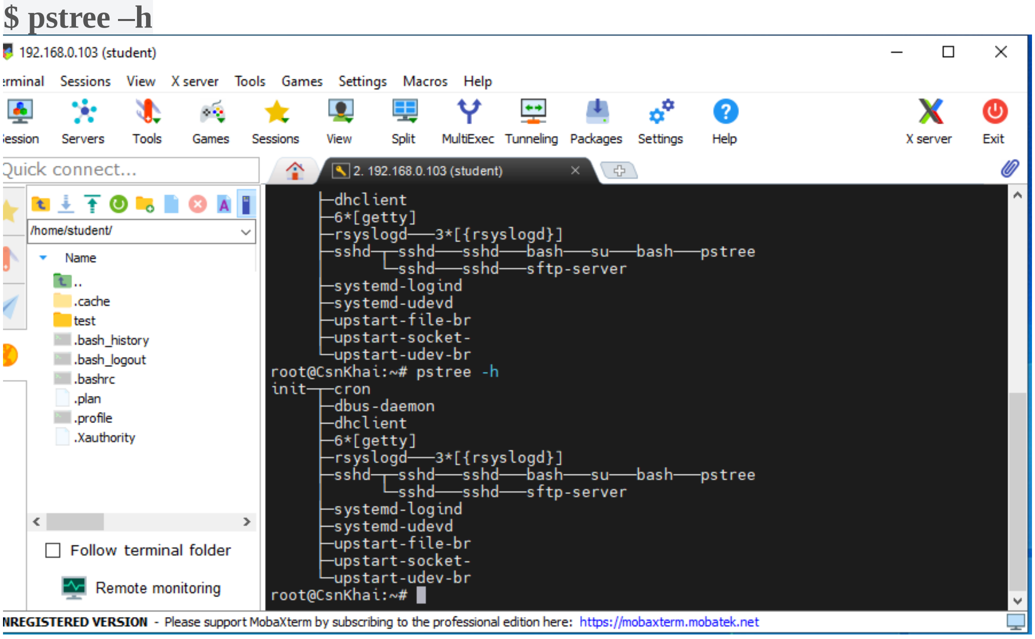
Part1

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

There are five Linux process states: running & runnable, interruptable_sleep, uninterruptable_sleep, stopped, and zombie.

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

$ pstree –h


192.168.0.103 (student)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
2. 192.168.0.103 (student)
/home/student/
Name
..
.cache
test
.bash_history
.bash_logout
.bashrc
.plan
.profile
.Xauthority
□ Follow terminal folder
Remote monitoring
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

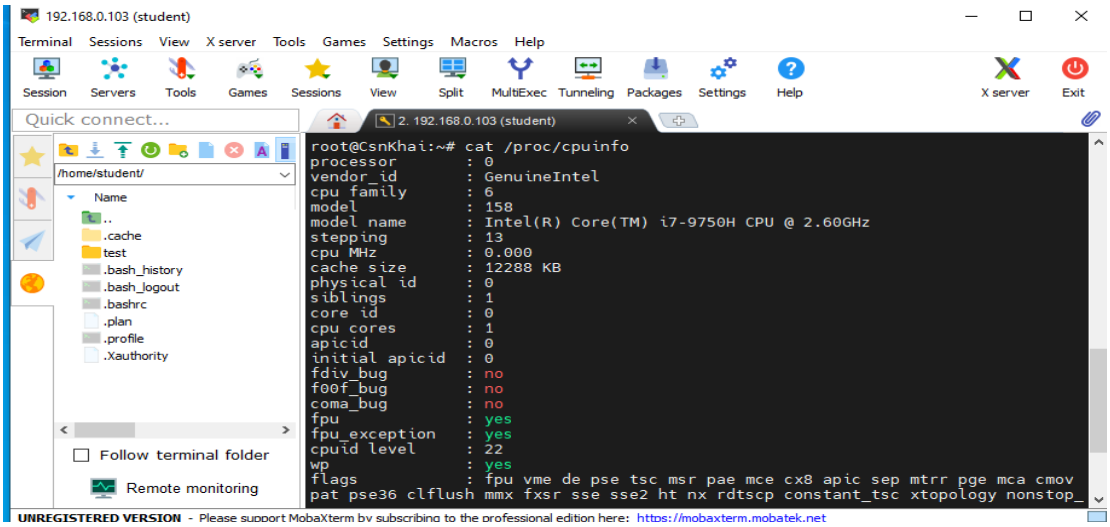
```
        ├─dhclient
        ├─6*[getty]
        ├─rsyslogd───3*[{rsyslogd}]
        ├─sshd─┬─sshd───sshd───bash───su───bash───pstree
        │      └─sshd───sshd───sftp-server
        ├─systemd-logind
        ├─systemd-udevd
        ├─upstart-file-br
        ├─upstart-socket-
        └─upstart-udev-br
root@CsnKhai:~# pstree -h
init─┬─cron
     ├─dbus-daemon
     ├─dhclient
     ├─6*[getty]
     ├─rsyslogd───3*[{rsyslogd}]
     ├─sshd─┬─sshd───sshd───bash───su───bash───pstree
     │      └─sshd───sshd───sftp-server
     ├─systemd-logind
     ├─systemd-udevd
     ├─upstart-file-br
     ├─upstart-socket-
     └─upstart-udev-br
root@CsnKhai:~#
```

3. What is a proc file system?

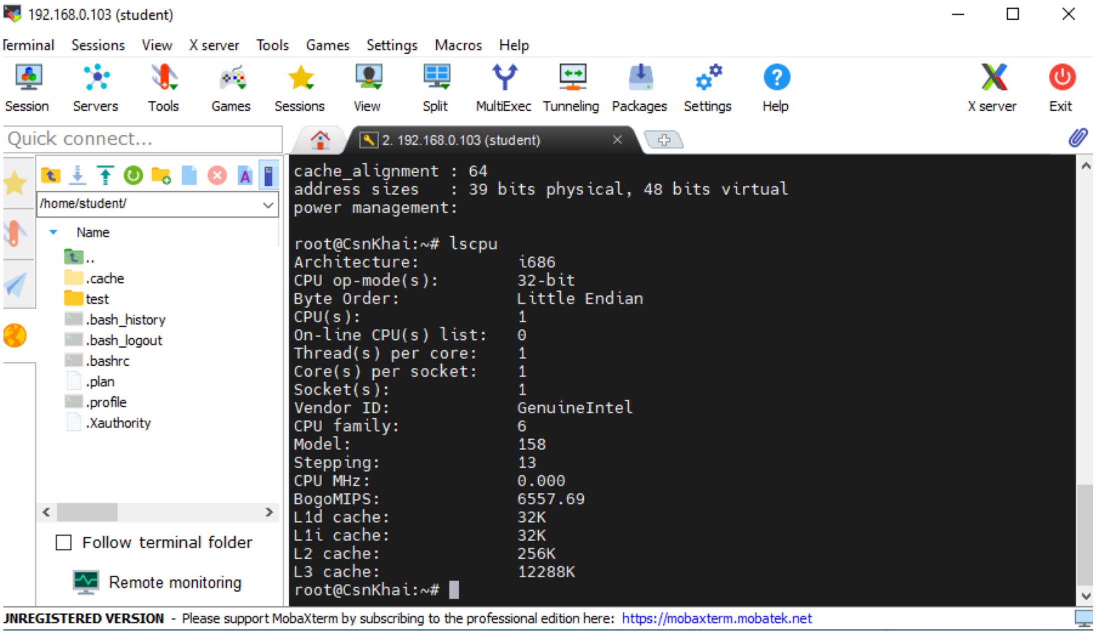
Proc file system (procfs) is a virtual file system created on the fly when the system boots and is dissolved at the time of system shutdown. It contains useful information about the processes that are currently running, it is regarded as a control and information center for the kernel.

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

You can use the /proc/cpuinfo file or the lscpu command to find out all the information about the processor.


192.168.0.103 (student)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
2. 192.168.0.103 (student)
/home/student/
Name
..
.cache
test
.bash_history
.bash_logout
.bashrc
.plan
.profile
.Xauthority
□ Follow terminal folder
Remote monitoring
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

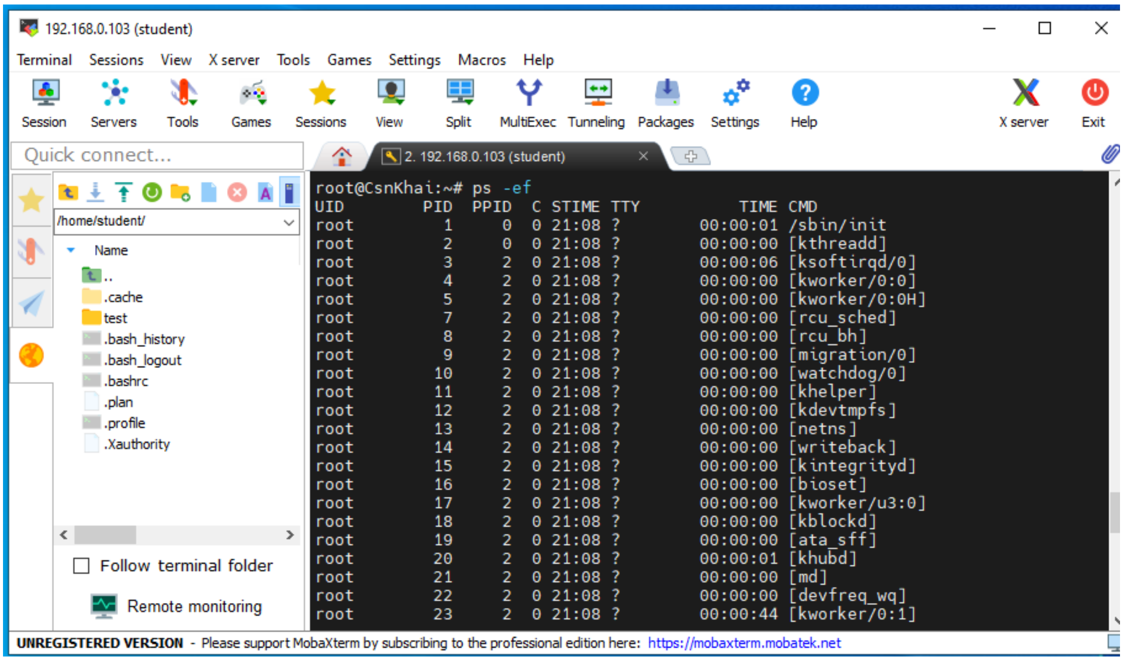
```
root@CsnKhai:~# cat /proc/cpuinfo
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 6
model           : 158
model name      : Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz
stepping        : 13
cpu MHz         : 0.000
cache size      : 12288 KB
physical id     : 0
siblings        : 1
core id         : 0
cpu cores       : 1
apicid          : 0
initial apicid  : 0
fdiv_bug        : no
f00f_bug        : no
coma_bug        : no
fpu             : yes
fpu_exception   : yes
cpuid level     : 22
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 nx rdtscp constant_tsc xtopology nonstop_
```


192.168.0.103 (student)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
2. 192.168.0.103 (student)
/home/student/
Name
..
.cache
test
.bash_history
.bash_logout
.bashrc
.plan
.profile
.Xauthority
☐ Follow terminal folder
Remote monitoring

```
cache_alignment : 64
address sizes   : 39 bits physical, 48 bits virtual
power management:

root@CsnKhai:~# lscpu
Architecture:          i686
CPU op-mode(s):        32-bit
Byte Order:            Little Endian
CPU(s):                1
On-line CPU(s) list:   0
Thread(s) per core:    1
Core(s) per socket:    1
Socket(s):             1
Vendor ID:             GenuineIntel
CPU family:            6
Model:                 158
Stepping:              13
CPU MHz:               0.000
BogoMIPS:              6557.69
L1d cache:             32K
L1i cache:             32K
L2 cache:              256K
L3 cache:              12288K
root@CsnKhai:~# 
```

JNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

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**.


192.168.0.103 (student)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
2. 192.168.0.103 (student)
/home/student/
Name
..
.cache
test
.bash_history
.bash_logout
.bashrc
.plan
.profile
.Xauthority
☐ Follow terminal folder
Remote monitoring

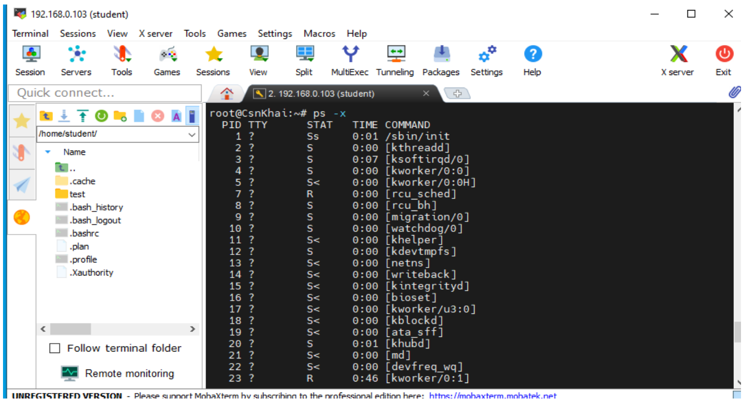
```
root@CsnKhai:~# ps -ef
UID        PID  PPID  C STIME TTY          TIME CMD
root         1     0  0 21:08 ?        00:00:01 /sbin/init
root         2     0  0 21:08 ?        00:00:00 [kthreadd]
root         3     2  0 21:08 ?        00:00:06 [ksoftirqd/0]
root         4     2  0 21:08 ?        00:00:00 [kworker/0:0]
root         5     2  0 21:08 ?        00:00:00 [kworker/0:0H]
root         7     2  0 21:08 ?        00:00:00 [rcu_sched]
root         8     2  0 21:08 ?        00:00:00 [rcu_bh]
root         9     2  0 21:08 ?        00:00:00 [migration/0]
root        10     2  0 21:08 ?        00:00:00 [watchdog/0]
root        11     2  0 21:08 ?        00:00:00 [khelper]
root        12     2  0 21:08 ?        00:00:00 [kdevtmpfs]
root        13     2  0 21:08 ?        00:00:00 [netns]
root        14     2  0 21:08 ?        00:00:00 [writeback]
root        15     2  0 21:08 ?        00:00:00 [kintegrityd]
root        16     2  0 21:08 ?        00:00:00 [bioset]
root        17     2  0 21:08 ?        00:00:00 [kworker/u3:0]
root        18     2  0 21:08 ?        00:00:00 [kblockd]
root        19     2  0 21:08 ?        00:00:00 [ata_sff]
root        20     2  0 21:08 ?        00:00:01 [khubd]
root        21     2  0 21:08 ?        00:00:00 [md]
root        22     2  0 21:08 ?        00:00:00 [devfreq_wq]
root        23     2  0 21:08 ?        00:00:44 [kworker/0:1]
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

6. How to define kernel processes and user processes?

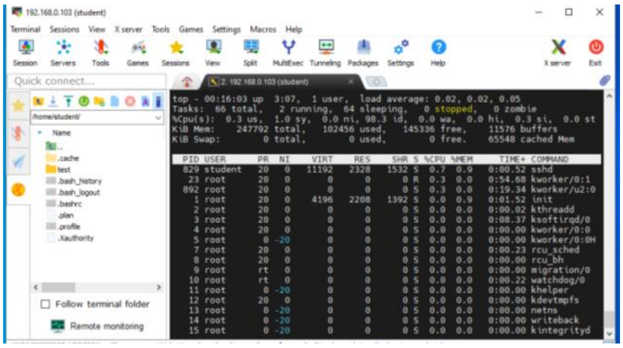
User-space processes have its own virtual address space. **Kernel processes** or threads do not have their own address space, they operate within kernel address space only. And they may be started before the kernel has started any user process.

If you want to find out exactly which tasks were opened on behalf of the current account, it will be enough to enter the line ps –x in the console



```
root@CsnKhai:~# ps -x
  PID TTY      STAT   TIME COMMAND
    1 ?        Ss     0:01 /sbin/init
    2 ?        S      0:00 [kthreadd]
    3 ?        S      0:07 [ksoftirqd/0]
    4 ?        S      0:00 [kworker/0:0]
    5 ?        S<     0:00 [kworker/0:0H]
    7 ?        R      0:00 [rcu_sched]
    8 ?        S      0:00 [rcu_bh]
    9 ?        S      0:00 [migration/0]
   10 ?        S      0:00 [watchdog/0]
   11 ?        S<     0:00 [khelper]
   12 ?        S      0:00 [kdevtmpfs]
   13 ?        S<     0:00 [netns]
   14 ?        S<     0:00 [writeback]
   15 ?        S<     0:00 [kintegrityd]
   16 ?        S<     0:00 [bioset]
   17 ?        S<     0:00 [kworker/u3:0]
   18 ?        S<     0:00 [kblockd]
   19 ?        S<     0:00 [ata_sff]
   20 ?        S      0:01 [khubd]
   21 ?        S<     0:00 [md]
   22 ?        S<     0:00 [devfreq_wq]
   23 ?        R      0:46 [kworker/0:1]
```

7. **Print the list of processes to the terminal. Briefly describe the statuses of the processes.**



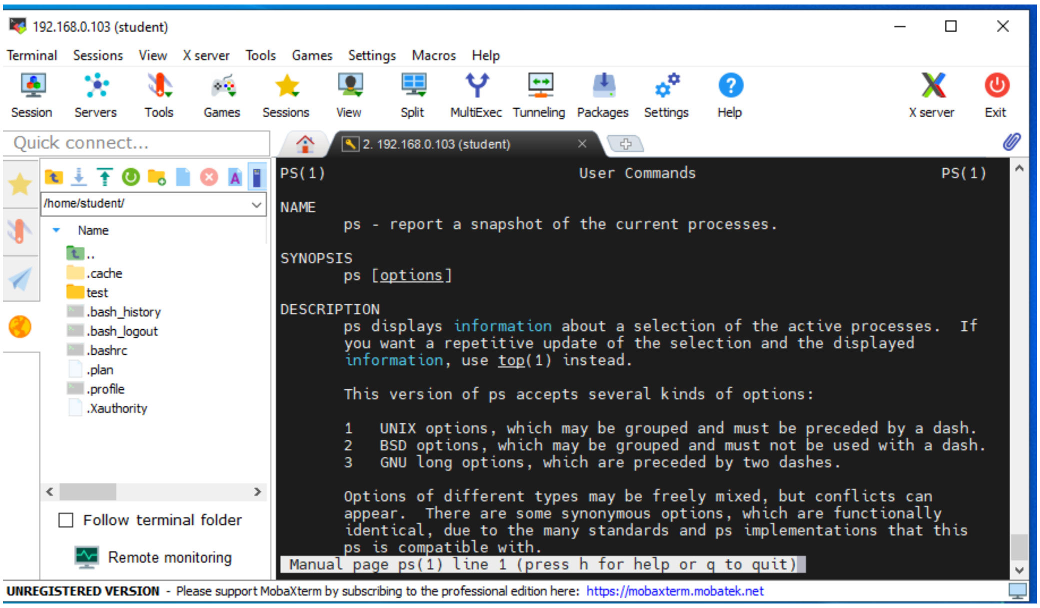
```
top - 00:16:03 up  3:07,  1 user,  load average: 0.02, 0.02, 0.05
Tasks:  66 total,   2 running,  64 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.3 us,  1.0 sy,  0.0 ni, 98.3 id,  0.0 wa,  0.0 hi,  0.3 si,  0.0 st
KiB Mem:    247792 total,   102456 used,   145336 free,    11576 buffers
KiB Swap:        0 total,        0 used,        0 free.    65548 cached Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
  829 student   20   0   11192   2328   1532 S  0.7  0.9   0:00.52 sshd
   23 root      20   0       0      0      0 R  0.3  0.0   0:54.68 kworker/0:1
  892 root      20   0       0      0      0 S  0.3  0.0   0:19.34 kworker/u2:0
    1 root      20   0    4196   2208   1392 S  0.0  0.9   0:01.52 init
    2 root      20   0       0      0      0 S  0.0  0.0   0:00.02 kthreadd
    3 root      20   0       0      0      0 S  0.0  0.0   0:08.37 ksoftirqd/0
    4 root      20   0       0      0      0 S  0.0  0.0   0:00.00 kworker/0:0
    5 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 kworker/0:0H
    7 root      20   0       0      0      0 S  0.0  0.0   0:00.23 rcu_sched
    8 root      20   0       0      0      0 S  0.0  0.0   0:00.00 rcu_bh
    9 root      rt   0       0      0      0 S  0.0  0.0   0:00.00 migration/0
   10 root      rt   0       0      0      0 S  0.0  0.0   0:00.22 watchdog/0
   11 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 khelper
   12 root      20   0       0      0      0 S  0.0  0.0   0:00.00 kdevtmpfs
   13 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 netns
   14 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 writeback
   15 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 kintegrityd
```

S is the current **status** of the process: **R** — running; S — sleeping, Z — zombie.

8. **Display only the processes of a specific user.**

$ ps –fU

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

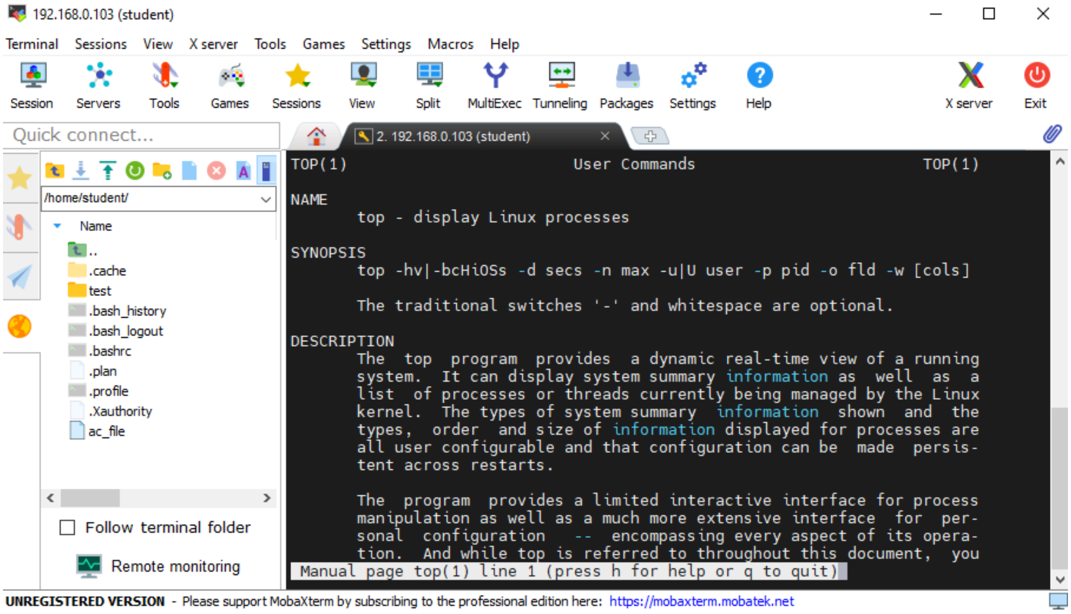


```
PS(1)                         User Commands                         PS(1)

NAME
       ps - report a snapshot of the current processes.

SYNOPSIS
       ps [options]

DESCRIPTION
       ps displays information about a selection of the active processes.  If
       you want a repetitive update of the selection and the displayed
       information, use top(1) instead.

       This version of ps accepts several kinds of options:

       1   UNIX options, which may be grouped and must be preceded by a dash.
       2   BSD options, which may be grouped and must not be used with a dash.
       3   GNU long options, which are preceded by two dashes.

       Options of different types may be freely mixed, but conflicts can
       appear.  There are some synonymous options, which are functionally
       identical, due to the many standards and ps implementations that this
       ps is compatible with.
 Manual page ps(1) line 1 (press h for help or q to quit)
```

Top is a command that allows users to monitor processes and system resource usage in Linux.

10. What information does top command display?

The top (table of processes) command shows a real-time view of running processes in Linux and displays kernel-managed tasks. The command also provides a system information summary that shows resource utilization, including CPU and memory usage.

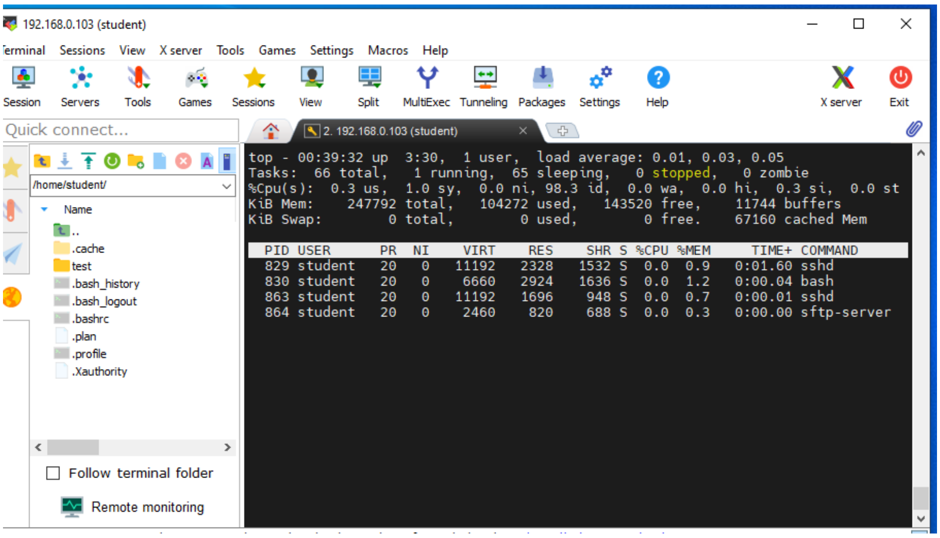


```
TOP(1)                        User Commands                        TOP(1)

NAME
       top - display Linux processes

SYNOPSIS
       top -hv|-bcHiOSs -d secs -n max -u|U user -p pid -o fld -w [cols]

       The traditional switches '-' and whitespace are optional.

DESCRIPTION
       The  top  program  provides  a dynamic real-time view of a running
       system.  It can display system summary information as  well  as  a
       list  of processes or threads currently being managed by the Linux
       kernel.  The types of system summary  information  shown  and  the
       types,  order  and size of information displayed for processes are
       all user configurable and that configuration can be  made  persis-
       tent across restarts.

       The  program  provides a limited interactive interface for process
       manipulation as well as a much more extensive interface  for  per-
       sonal  configuration   --  encompassing every aspect of its opera-
       tion.  And while top is referred to throughout this document,  you
 Manual page top(1) line 1 (press h for help or q to quit)
```

192.168.0.103 (student)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
2. 192.168.0.103 (student)
/home/student/
Name
..
.cache
test
.bash_history
.bash_logout
.bashrc
.plan
.profile
.Xauthority
ac_file
Follow terminal folder
Remote monitoring
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

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

To filter processes running as a specific user, use the u command:



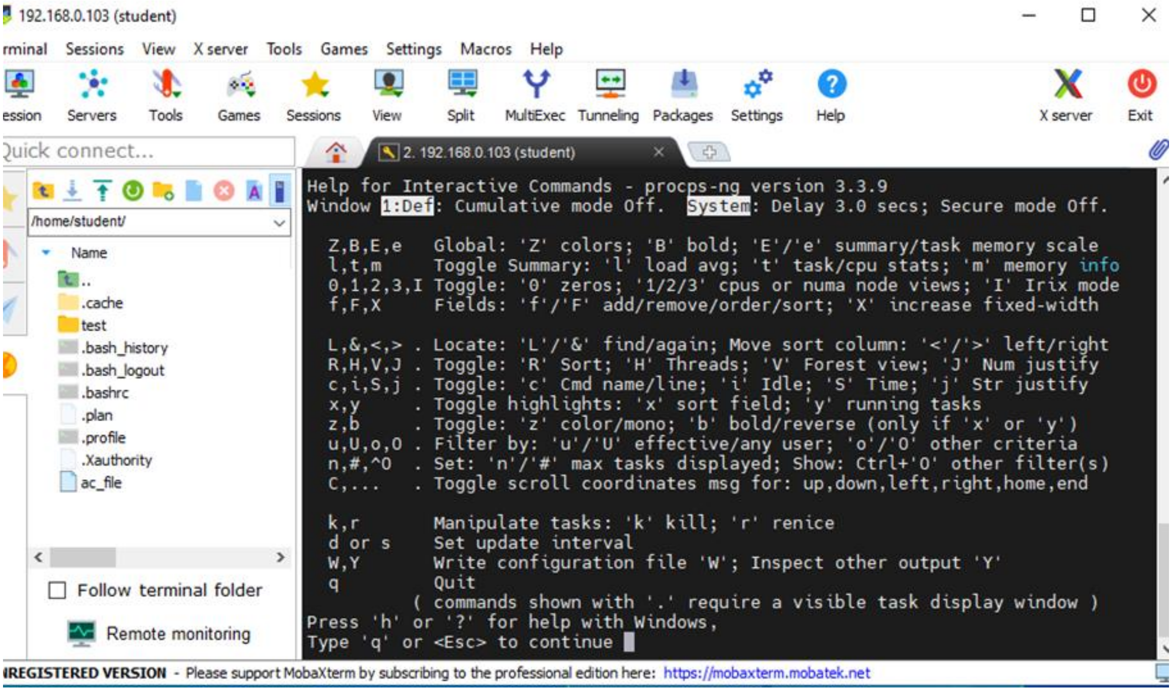
```
top - 00:39:32 up  3:30,  1 user,  load average: 0.01, 0.03, 0.05
Tasks:  66 total,   1 running,  65 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.3 us,  1.0 sy,  0.0 ni, 98.3 id,  0.0 wa,  0.0 hi,  0.3 si,  0.0 st
KiB Mem:    247792 total,   104272 used,   143520 free,    11744 buffers
KiB Swap:        0 total,        0 used,        0 free.    67160 cached Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
  829 student   20   0   11192   2328   1532 S  0.0  0.9   0:01.60 sshd
  830 student   20   0    6660   2924   1636 S  0.0  1.2   0:00.04 bash
  863 student   20   0   11192   1696    948 S  0.0  0.7   0:00.01 sshd
  864 student   20   0    2460    820    688 S  0.0  0.3   0:00.00 sftp-server
```

192.168.0.103 (student)
erminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
2. 192.168.0.103 (student)
/home/student/
Name
..
.cache
test
.bash_history
.bash_logout
.bashrc
.plan
.profile
.Xauthority
Follow terminal folder
Remote monitoring

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

Command'k'- kill completion of the process
Command 'r' – renice, change the current task priority

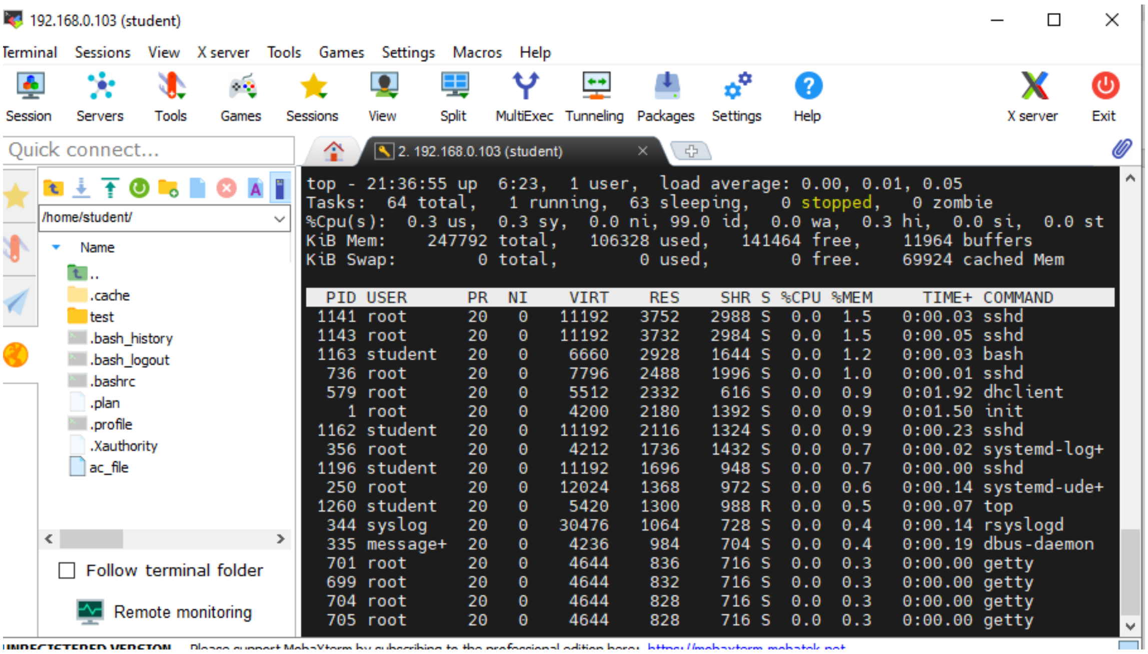
Pressing the **h** key displays command help:


192.168.0.103 (student)
rminal Sessions View X server Tools Games Settings Macros Help
ession Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
2. 192.168.0.103 (student)
/home/student/
Name
..
.cache
test
.bash_history
.bash_logout
.bashrc
.plan
.profile
.Xauthority
ac_file
☐ Follow terminal folder
Remote monitoring
IREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

```
Help for Interactive Commands - procps-ng version 3.3.9
Window 1:Def: Cumulative mode Off.  System: Delay 3.0 secs; Secure mode Off.

  Z,B,E,e   Global: 'Z' colors; 'B' bold; 'E'/'e' summary/task memory scale
  l,t,m     Toggle Summary: 'l' load avg; 't' task/cpu stats; 'm' memory info
  0,1,2,3,I Toggle: '0' zeros; '1/2/3' cpus or numa node views; 'I' Irix mode
  f,F,X     Fields: 'f'/'F' add/remove/order/sort; 'X' increase fixed-width

  L,&,<,> . Locate: 'L'/'&' find/again; Move sort column: '<'/'>' left/right
  R,H,V,J . Toggle: 'R' Sort; 'H' Threads; 'V' Forest view; 'J' Num justify
  c,i,S,j . Toggle: 'c' Cmd name/line; 'i' Idle; 'S' Time; 'j' Str justify
  x,y     . Toggle highlights: 'x' sort field; 'y' running tasks
  z,b     . Toggle: 'z' color/mono; 'b' bold/reverse (only if 'x' or 'y')
  u,U,o,O . Filter by: 'u'/'U' effective/any user; 'o'/'O' other criteria
  n,#,^O  . Set: 'n'/'#' max tasks displayed; Show: Ctrl+'O' other filter(s)
  C,...   . Toggle scroll coordinates msg for: up,down,left,right,home,end

  k,r       Manipulate tasks: 'k' kill; 'r' renice
  d or s    Set update interval
  W,Y       Write configuration file 'W'; Inspect other output 'Y'
  q         Quit
          ( commands shown with '.' require a visible task display window )
Press 'h' or '?' for help with Windows,
Type 'q' or <Esc> to continue
```

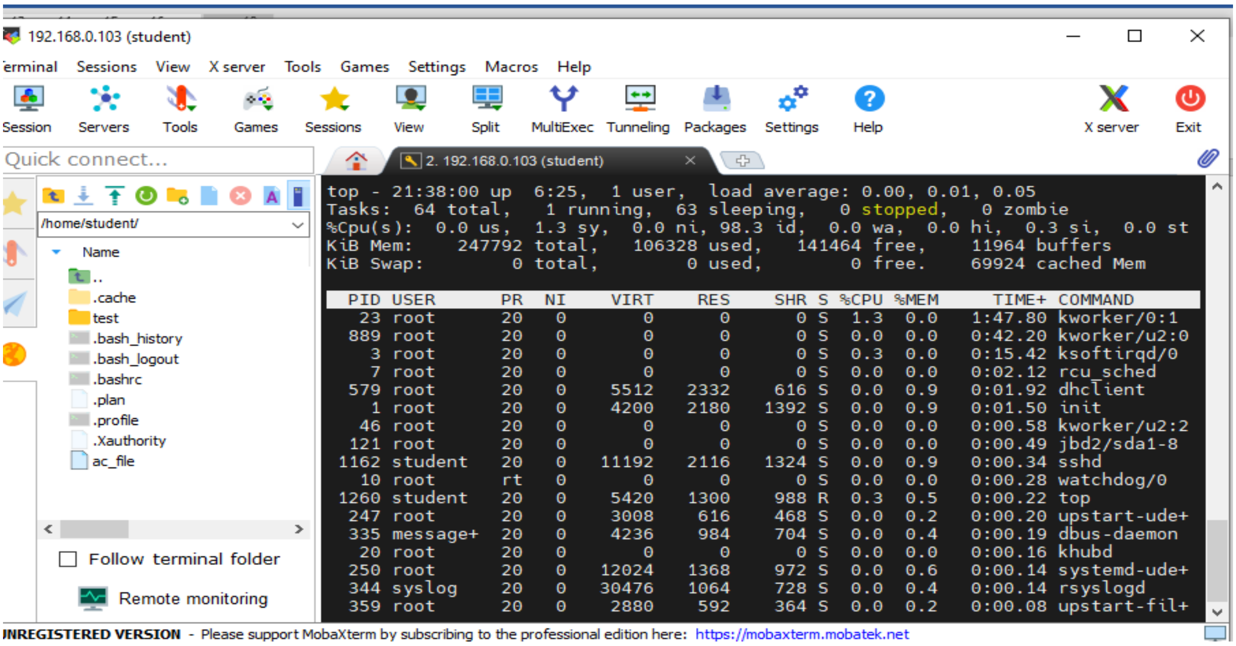
13. Sort the contents of the processes window using various parameters (for example, the amount of processor time taken up, etc.)

Shift+M Sort by memory (%MEM):


192.168.0.103 (student)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
2. 192.168.0.103 (student)
/home/student/
Name
..
.cache
test
.bash_history
.bash_logout
.bashrc
.plan
.profile
.Xauthority
ac_file
☐ Follow terminal folder
Remote monitoring

```
top - 21:36:55 up  6:23,  1 user,  load average: 0.00, 0.01, 0.05
Tasks:  64 total,   1 running,  63 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.3 us,  0.3 sy,  0.0 ni, 99.0 id,  0.0 wa,  0.3 hi,  0.0 si,  0.0 st
KiB Mem:    247792 total,   106328 used,   141464 free,    11964 buffers
KiB Swap:        0 total,        0 used,        0 free.    69924 cached Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 1141 root      20   0   11192   3752   2988 S  0.0  1.5   0:00.03 sshd
 1143 root      20   0   11192   3732   2984 S  0.0  1.5   0:00.05 sshd
 1163 student   20   0    6660   2928   1644 S  0.0  1.2   0:00.03 bash
  736 root      20   0    7796   2488   1996 S  0.0  1.0   0:00.01 sshd
  579 root      20   0    5512   2332    616 S  0.0  0.9   0:01.92 dhclient
    1 root      20   0    4200   2180   1392 S  0.0  0.9   0:01.50 init
 1162 student   20   0   11192   2116   1324 S  0.0  0.9   0:00.23 sshd
  356 root      20   0    4212   1736   1432 S  0.0  0.7   0:00.02 systemd-log+
 1196 student   20   0   11192   1696    948 S  0.0  0.7   0:00.00 sshd
  250 root      20   0   12024   1368    972 S  0.0  0.6   0:00.14 systemd-ude+
 1260 student   20   0    5420   1300    988 R  0.0  0.5   0:00.07 top
  344 syslog    20   0   30476   1064    728 S  0.0  0.4   0:00.14 rsyslogd
  335 message+  20   0    4236    984    704 S  0.0  0.4   0:00.19 dbus-daemon
  701 root      20   0    4644    836    716 S  0.0  0.3   0:00.00 getty
  699 root      20   0    4644    832    716 S  0.0  0.3   0:00.00 getty
  704 root      20   0    4644    828    716 S  0.0  0.3   0:00.00 getty
  705 root      20   0    4644    828    716 S  0.0  0.3   0:00.00 getty
```

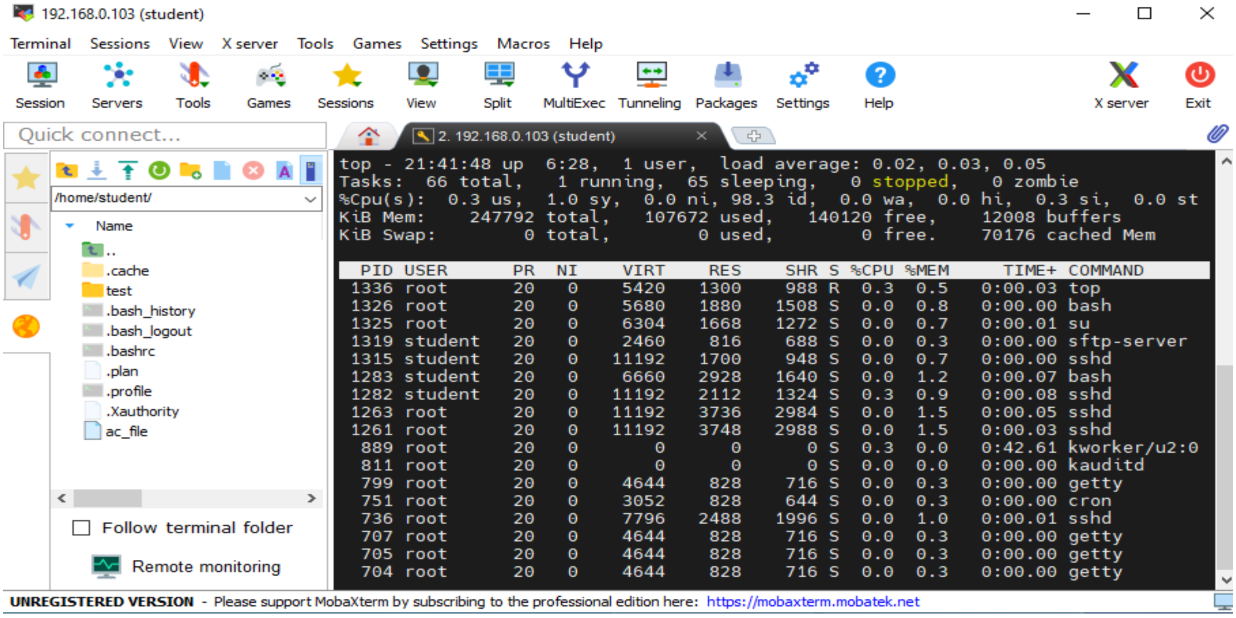
Shift+T Sort by time:


192.168.0.103 (student)
erminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
2. 192.168.0.103 (student)
/home/student/
Name
..
.cache
test
.bash_history
.bash_logout
.bashrc
.plan
.profile
.Xauthority
ac_file
□ Follow terminal folder
Remote monitoring

```
top - 21:38:00 up  6:25,  1 user,  load average: 0.00, 0.01, 0.05
Tasks:  64 total,   1 running,  63 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.0 us,  1.3 sy,  0.0 ni, 98.3 id,  0.0 wa,  0.0 hi,  0.3 si,  0.0 st
KiB Mem:    247792 total,   106328 used,   141464 free,    11964 buffers
KiB Swap:        0 total,        0 used,        0 free.    69924 cached Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
   23 root      20   0       0      0      0 S  1.3  0.0   1:47.80 kworker/0:1
  889 root      20   0       0      0      0 S  0.0  0.0   0:42.20 kworker/u2:0
    3 root      20   0       0      0      0 S  0.3  0.0   0:15.42 ksoftirqd/0
    7 root      20   0       0      0      0 S  0.0  0.0   0:02.12 rcu_sched
  579 root      20   0    5512   2332    616 S  0.0  0.9   0:01.92 dhclient
    1 root      20   0    4200   2180   1392 S  0.0  0.9   0:01.50 init
   46 root      20   0       0      0      0 S  0.0  0.0   0:00.58 kworker/u2:2
  121 root      20   0       0      0      0 S  0.0  0.0   0:00.49 jbd2/sda1-8
 1162 student   20   0   11192   2116   1324 S  0.0  0.9   0:00.34 sshd
   10 root      rt   0       0      0      0 S  0.0  0.0   0:00.28 watchdog/0
 1260 student   20   0    5420   1300    988 R  0.3  0.5   0:00.22 top
  247 root      20   0    3008    616    468 S  0.0  0.2   0:00.20 upstart-ude+
  335 message+  20   0    4236    984    704 S  0.0  0.4   0:00.19 dbus-daemon
   20 root      20   0       0      0      0 S  0.0  0.0   0:00.16 khubd
  250 root      20   0   12024   1368    972 S  0.0  0.6   0:00.14 systemd-ude+
  344 syslog    20   0   30476   1064    728 S  0.0  0.4   0:00.14 rsyslogd
  359 root      20   0    2880    592    364 S  0.0  0.2   0:00.08 upstart-fil+
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

Shift+N Sort by PID:

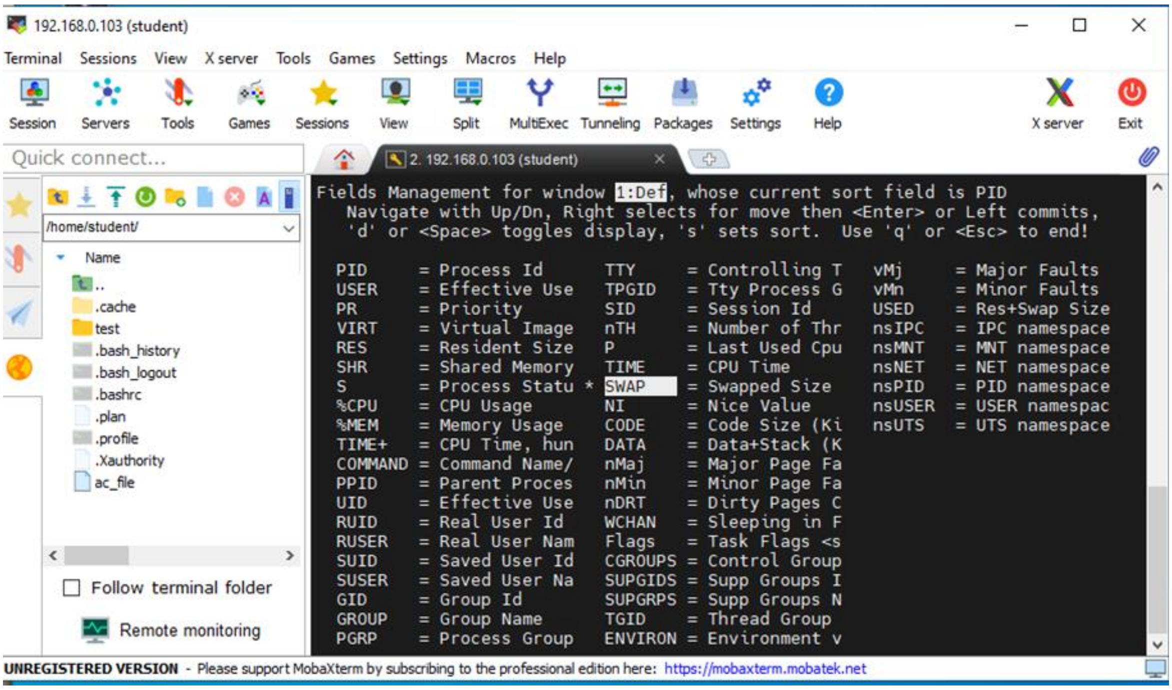

192.168.0.103 (student)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
2. 192.168.0.103 (student)
/home/student/
Name
..
.cache
test
.bash_history
.bash_logout
.bashrc
.plan
.profile
.Xauthority
ac_file
□ Follow terminal folder
Remote monitoring

```
top - 21:41:48 up  6:28,  1 user,  load average: 0.02, 0.03, 0.05
Tasks:  66 total,   1 running,  65 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.3 us,  1.0 sy,  0.0 ni, 98.3 id,  0.0 wa,  0.0 hi,  0.3 si,  0.0 st
KiB Mem:    247792 total,   107672 used,   140120 free,    12008 buffers
KiB Swap:        0 total,        0 used,        0 free.    70176 cached Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 1336 root      20   0    5420   1300    988 R  0.3  0.5   0:00.03 top
 1326 root      20   0    5680   1880   1508 S  0.0  0.8   0:00.00 bash
 1325 root      20   0    6304   1668   1272 S  0.0  0.7   0:00.01 su
 1319 student   20   0    2460    816    688 S  0.0  0.3   0:00.00 sftp-server
 1315 student   20   0   11192   1700    948 S  0.0  0.7   0:00.00 sshd
 1283 student   20   0    6660   2928   1640 S  0.0  1.2   0:00.07 bash
 1282 student   20   0   11192   2112   1324 S  0.3  0.9   0:00.08 sshd
 1263 root      20   0   11192   3736   2984 S  0.0  1.5   0:00.05 sshd
 1261 root      20   0   11192   3748   2988 S  0.0  1.5   0:00.03 sshd
  889 root      20   0       0      0      0 S  0.3  0.0   0:42.61 kworker/u2:0
  811 root      20   0       0      0      0 S  0.0  0.0   0:00.00 kauditd
  799 root      20   0    4644    828    716 S  0.0  0.3   0:00.00 getty
  751 root      20   0    3052    828    644 S  0.0  0.3   0:00.00 cron
  736 root      20   0    7796   2488   1996 S  0.0  1.0   0:00.01 sshd
  707 root      20   0    4644    828    716 S  0.0  0.3   0:00.00 getty
  705 root      20   0    4644    828    716 S  0.0  0.3   0:00.00 getty
  704 root      20   0    4644    828    716 S  0.0  0.3   0:00.00 getty
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

Sort by SWAP:

Shift+F
Choose SWAP
+d
+s


192.168.0.103 (student)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
2. 192.168.0.103 (student)
/home/student/
Name
..
.cache
test
.bash_history
.bash_logout
.bashrc
.plan
.profile
.Xauthority
ac_file
Follow terminal folder
Remote monitoring
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

```
Fields Management for window 1:Def, whose current sort field is PID
   Navigate with Up/Dn, Right selects for move then <Enter> or Left commits,
   'd' or <Space> toggles display, 's' sets sort.  Use 'q' or <Esc> to end!

  PID     = Process Id      TTY     = Controlling T   vMj     = Major Faults
  USER    = Effective Use   TPGID   = Tty Process G   vMn     = Minor Faults
  PR      = Priority        SID     = Session Id      USED    = Res+Swap Size
  VIRT    = Virtual Image   nTH     = Number of Thr   nsIPC   = IPC namespace
  RES     = Resident Size   P       = Last Used Cpu   nsMNT   = MNT namespace
  SHR     = Shared Memory   TIME    = CPU Time        nsNET   = NET namespace
  S       = Process Statu * SWAP    = Swapped Size    nsPID   = PID namespace
  %CPU    = CPU Usage       NI      = Nice Value      nsUSER  = USER namespac
  %MEM    = Memory Usage    CODE    = Code Size (Ki   nsUTS   = UTS namespace
  TIME+   = CPU Time, hun   DATA    = Data+Stack (K
  COMMAND = Command Name/   nMaj    = Major Page Fa
  PPID    = Parent Proces   nMin    = Minor Page Fa
  UID     = Effective Use   nDRT    = Dirty Pages C
  RUID    = Real User Id    WCHAN   = Sleeping in F
  RUSER   = Real User Nam   Flags   = Task Flags <s
  SUID    = Saved User Id   CGROUPS = Control Group
  SUSER   = Saved User Na   SUPGIDS = Supp Groups I
  GID     = Group Id        SUPGRPS = Supp Groups N
  GROUP   = Group Name      TGID    = Thread Group
  PGRP    = Process Group   ENVIRON = Environment v
```

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

To change the priority of processes, the **nice and renice** commands are used. When a process starts, its priority is set by the value of the parent process, for example, an xterm terminal or a shell command.

The nice command allows you to start a process with a priority equal to the sum of the parent (for example, 8) and the number specified as an option of the nice command:

nice -< number > command

Example:

$ nice -4 mc - will start mc with priority = 8+4=12.

The **renice** command is used to change the priority of an already running process:

$ renice < number > -p PID

Example**:**

$ renice 4 -p 11597 – will set the priority value for process mc (PID=11597) to 4.

After execution, renice will issue the following result: 11597: old priority 12, new priority 4 - old priority 12, new priority 4.

Negative values and 0 can be set only by the root user.

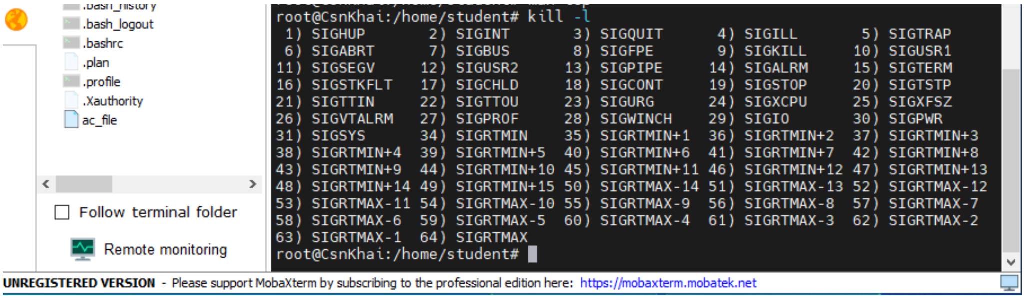
15. Can I change the priority of a process using the top command? If so, how?

In order to change the current task priority, you need to use **the -r (renice) command** with top. The PID will be queried, and then the new priority value (displayed in the NI column). The range of priority values is from minus 20 (the highest) to plus 19. Only the root user can set negative values and 0.

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

list of signal numbers and names:

$ kill -l



```
root@CsnKhai:/home/student# kill -l
 1) SIGHUP       2) SIGINT       3) SIGQUIT      4) SIGILL       5) SIGTRAP
 6) SIGABRT      7) SIGBUS       8) SIGFPE       9) SIGKILL     10) SIGUSR1
11) SIGSEGV     12) SIGUSR2     13) SIGPIPE     14) SIGALRM     15) SIGTERM
16) SIGSTKFLT   17) SIGCHLD     18) SIGCONT     19) SIGSTOP     20) SIGTSTP
21) SIGTTIN     22) SIGTTOU     23) SIGURG      24) SIGXCPU     25) SIGXFSZ
26) SIGVTALRM   27) SIGPROF     28) SIGWINCH    29) SIGIO       30) SIGPWR
31) SIGSYS      34) SIGRTMIN    35) SIGRTMIN+1  36) SIGRTMIN+2  37) SIGRTMIN+3
38) SIGRTMIN+4  39) SIGRTMIN+5  40) SIGRTMIN+6  41) SIGRTMIN+7  42) SIGRTMIN+8
43) SIGRTMIN+9  44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12 47) SIGRTMIN+13
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14 51) SIGRTMAX-13 52) SIGRTMAX-12
53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9  56) SIGRTMAX-8  57) SIGRTMAX-7
58) SIGRTMAX-6  59) SIGRTMAX-5  60) SIGRTMAX-4  61) SIGRTMAX-3  62) SIGRTMAX-2
63) SIGRTMAX-1  64) SIGRTMAX
root@CsnKhai:/home/student#
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

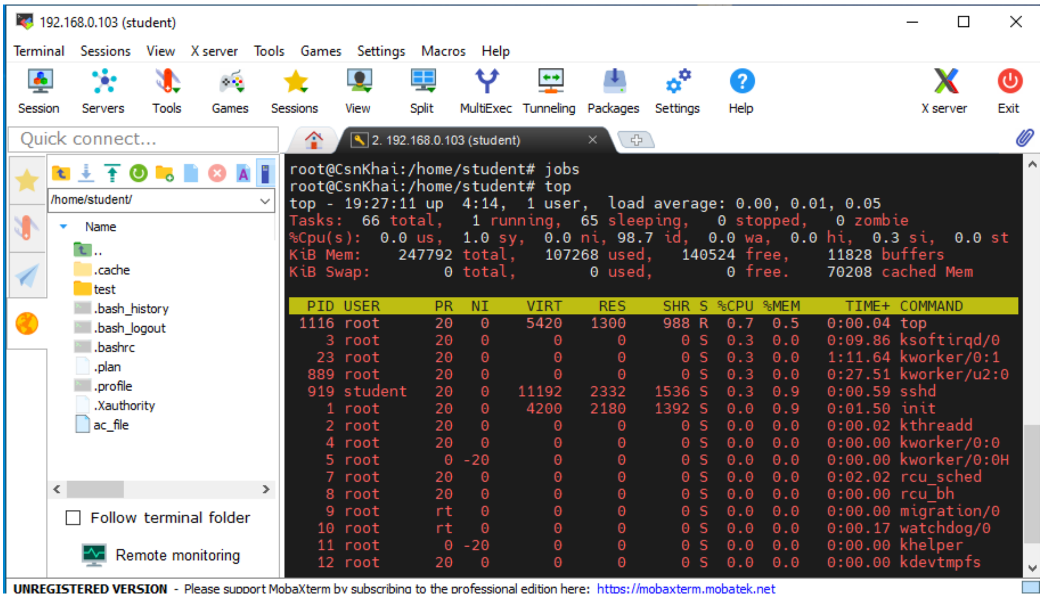
To send a specific signal to a given process:

$ kill <signal number> the PID number of the selected process

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

CTRL+Z - Stops and sends commands to the back**.**

Jobs shows all stopped commands.



```
root@CsnKhai:/home/student# jobs
root@CsnKhai:/home/student# top
top - 19:27:11 up  4:14,  1 user,  load average: 0.00, 0.01, 0.05
Tasks:  66 total,   1 running,  65 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.0 us,  1.0 sy,  0.0 ni, 98.7 id,  0.0 wa,  0.0 hi,  0.3 si,  0.0 st
KiB Mem:    247792 total,   107268 used,   140524 free,    11828 buffers
KiB Swap:        0 total,        0 used,        0 free.    70208 cached Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 1116 root      20   0    5420   1300    988 R  0.7  0.5   0:00.04 top
    3 root      20   0       0      0      0 S  0.3  0.0   0:09.86 ksoftirqd/0
   23 root      20   0       0      0      0 S  0.3  0.0   1:11.64 kworker/0:1
  889 root      20   0       0      0      0 S  0.3  0.0   0:27.51 kworker/u2:0
  919 student   20   0   11192   2332   1536 S  0.3  0.9   0:00.59 sshd
    1 root      20   0    4200   2180   1392 S  0.0  0.9   0:01.50 init
    2 root      20   0       0      0      0 S  0.0  0.0   0:00.02 kthreadd
    4 root      20   0       0      0      0 S  0.0  0.0   0:00.00 kworker/0:0
    5 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 kworker/0:0H
    7 root      20   0       0      0      0 S  0.0  0.0   0:02.02 rcu_sched
    8 root      20   0       0      0      0 S  0.0  0.0   0:00.00 rcu_bh
    9 root      rt   0       0      0      0 S  0.0  0.0   0:00.00 migration/0
   10 root      rt   0       0      0      0 S  0.0  0.0   0:00.17 watchdog/0
   11 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 khelper
   12 root      20   0       0      0      0 S  0.0  0.0   0:00.00 kdevtmpfs
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

In order to output the command from the background process, is used the **fg** (**foreground)** command. Which can either be passed the task number as an argument, or run without arguments. In the latter case, the task marked with a + sign in the jobs list will be displayed – the last task sent to the background.

The **bg (background)** command is used to keep processes running in the background.

Nohup keeps processes running on Linux systems even after exiting a shell or terminal. It prevents processes or jobs from receiving a SIGHUP (Signal Hang UP) signal.