p1

```
vagrant@ubuntu2210:~$ mkdir midterm
vagrant@ubuntu2210:~$ ls
''$'\033\033'   Documents   midterm   week3
vagrant@ubuntu2210:~$ cd midterm
vagrant@ubuntu2210:~/midterm$ touch p1-file
vagrant@ubuntu2210:~/midterm$ ls
p1-file
vagrant@ubuntu2210:~/midterm$ ls -lrt
total 0
-rw-rw-r-- 1 vagrant vagrant 0 Feb 15 19:04 p1-file
```

```
vagrant@ubuntu2210:~/midterm$ chmod a-x p1-file
vagrant@ubuntu2210:~/midterm$ ls -lrt
total 0
-rw-r--r-- 1 vagrant vagrant 0 Feb 15 19:04 p1-file
vagrant@ubuntu2210:~/midterm$ chmod 744 p1-file
vagrant@ubuntu2210:~/midterm$ ls -lrt
total 0
-rwxr--r-- 1 vagrant vagrant 0 Feb 15 19:04 p1-file
```

p2.

Using only the ps utility, sort all processes by resident set size. Display the command and resident set size.

```
vagrant@ubuntu2210:~/midterm$ ps -e -o comm,rss
COMMAND             RSS
systemd           12308
kthreadd              0
rcu_gp                0
rcu_par_gp            0
slub_flushwq          0
netns                 0
kworker/0:0H-ev       0
mm_percpu_wq          0
rcu_tasks_kthre       0
```

```
STANDARD FORMAT SPECIFIERS
       Here are the different keywords that may be used to control the output format (e.g., with option -o) or to
       sort the selected processes with the GNU-style --sort option.

       For example: ps -eo pid,user,args --sort user
```

```
AIX FORMAT DESCRIPTORS
       This ps supports AIX format descriptors, which work somewhat like the formatting codes of printf(1) and
       printf(3).  For example, the normal default output can be produced with this: ps -eo "%p %y %x %c".  The
       NORMAL codes are described in the next section.

       CODE   NORMAL   HEADER
       %C     pcpu     %CPU
       %G     group    GROUP
       %P     ppid     PPID
       %U     user     USER
       %a     args     COMMAND
       %c     comm     COMMAND
       %g     rgroup   RGROUP
       %n     nice     NI
       %p     pid      PID
       %r     pgid     PGID
       %t     etime    ELAPSED
       %u     ruser    RUSER
       %x     time     TIME
       %y     tty      TTY
       %z     vsz      VSZ
```

```
       -o format
              User-defined format.  format is a single argument in the form of a blank-separated or comma-separated
              list, which offers a way to specify individual output columns.  The recognized keywords are described
              in the STANDARD FORMAT SPECIFIERS section below.  Headers may be renamed (ps -o pid,ruser=RealUser -o
              comm=Command) as desired.  If all column headers are empty (ps -o pid= -o comm=) then the header line
              will not be output.  Column width will increase as needed for wide headers; this may be used to widen
              up columns such as WCHAN (ps -o pid,wchan=WIDE-WCHAN-COLUMN -o comm).  Explicit width control (ps opid,
              wchan:42,cmd) is offered too.  The behavior of ps -o pid=X,comm=Y varies with personality; output may
              be one column named "X,comm=Y" or two columns named "X" and "Y".  Use multiple -o options when in
              doubt.  Use the PS_FORMAT environment variable to specify a default as desired; DefSysV and DefBSD are
              macros that may be used to choose the default UNIX or BSD columns.
```

p3.

Create a new user. Your new user should have the following:

a regular home directory in /home

```
vagrant@ubuntu2210:~/midterm$ ls
p1-file
vagrant@ubuntu2210:~/midterm$ sudo adduser newuser
Adding user `newuser' ...
Adding new group `newuser' (1003) ...
Adding new user `newuser' (1003) with group `newuser
Creating home directory `/home/newuser' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for newuser
Enter the new value, or press ENTER for the default
        Full Name []: midterm
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
vagrant@ubuntu2210:~/midterm$ ls /home
newuser  vagrant
vagrant@ubuntu2210:~/midterm$
```

use bash as their login shell

```
vagrant@ubuntu2210:~/midterm$ grep newuser /etc/passwd
newuser:x:1003:1003:midterm,,,:/home/newuser:/bin/bash
vagrant@ubuntu2210:~/midterm$
```

setpasswd or change passwd if assigned:

```
vagrant@ubuntu2210:~/midterm$ sudo passwd newuser
New password:
Retype new password:
passwd: password updated successfully
```

Create group:

```
vagrant@ubuntu2210:~/midterm$ sudo groupadd midterm
vagrant@ubuntu2210:~/midterm$ grep midterm /etc/group
midterm:x:1004:
vagrant@ubuntu2210:~/midterm$
```

```
GROUPADD(8)

NAME
       groupadd - create a new group

SYNOPSIS
       groupadd [options] group

DESCRIPTION
```

```
    -U, --users
        A list of usernames to add as members of the group.

        The default behavior (if the -g, -N, and -U options are not specified) is defined by the USERGROUPS_ENAB
        variable in /etc/login.defs.
```

change ownership:~~~~~~~~

```
CHGRP(1)                                          User Comman

NAME
       chgrp - change group ownership

SYNOPSIS
       chgrp [OPTION]... GROUP FILE...
       chgrp [OPTION]... --reference=RFILE FILE...
```

```
EXAMPLES
       chgrp staff /u
              Change the group of /u to "staff".

       chgrp -hR staff /u
              Change the group of /u and subfiles to "staff".
```

Add your new user to the midterm group:

p4.
Write a command that uses grep to find and print all the regular users on your system.
Only use grep. Assume that you don’t know how many regular users there are.

```
vagrant@ubuntu2210:~/midterm$ grep -E ':[1-9][0-9]{3}:' /etc/passwd
vagrant:x:1000:1000::/home/vagrant:/bin/bash
stanley:x:1001:1001::/home/stanley:/bin/sh
/etc/skel/stanley:x:1002:1002::/home//etc/skel/stanley:/bin/sh
newuser:x:1003:1003:midterm,,,:/home/newuser:/bin/bash
vagrant@ubuntu2210:~/midterm$
```

p5.
Write a command, using find , that will find and count all the files in /etc .
Hide all the error messages, “Permission denied”, by sending them to a file that doesn't store any data.

```
vagrant@ubuntu2210:~/midterm$ find /etc -type f 2>/dev/null | wc -l
706
```

p6.
Run a command that will display the version of the kernel that your VM is using.

```
vagrant@ubuntu2210:~/midterm$ cat /proc/version
Linux version 5.19.0-29-generic (buildd@lcy02-amd64-117) (x86_64-linux-gnu-gcc-12 (Ubuntu 12.2.0-3ubuntu1) 12.2.0, GNU l
d (GNU Binutils for Ubuntu) 2.39) #30-Ubuntu SMP PREEMPT_DYNAMIC Wed Jan 4 12:14:09 UTC 2023
vagrant@ubuntu2210:~/midterm$ uname -r
5.19.0-29-generic
vagrant@ubuntu2210:~/midterm$ hostnamectl | grep Kernel
         Kernel: Linux 5.19.0-29-generic
vagrant@ubuntu2210:~/midterm$
```

p7