



MTU

Ollscoil Teicneolaíochta na Mumhan
Munster Technological University

Lab 1

COMP6042_26213 Operating Systems in Practice

Student ID: R00201303

Date: 22.03.2022

Group: COMP1D-Y

Question1:

```
laiba@ubuntu: ~/A2_LAB1
laiba@ubuntu:~/Desktop$ cd ~
laiba@ubuntu:~$ cd A2_LAB1
laiba@ubuntu:~/A2_LAB1$ PWD
PWD: command not found
laiba@ubuntu:~/A2_LAB1$ pwd
/home/laiba/A2_LAB1
laiba@ubuntu:~/A2_LAB1$
```

```
laiba@ubuntu: ~/A2_LAB1
GNU nano 4.8                               Myprog1                               Modified
cd /etc
cat timezone
echo

tail -8 passwd
```

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Paste Text ^T To Spell ^_ Go To Line

```
laiba@ubuntu: ~/A2_LAB1
PWD: command not found
laiba@ubuntu:~/A2_LAB1$ pwd
/home/laiba/A2_LAB1
laiba@ubuntu:~/A2_LAB1$ nano Myprog1
laiba@ubuntu:~/A2_LAB1$ cat MyProg1
cat: MyProg1: No such file or directory
laiba@ubuntu:~/A2_LAB1$ cat Myprog1
cd /etc
cat timezone
echo

tail -8 passwd
laiba@ubuntu:~/A2_LAB1$ bash Myprog1
America/Los_Angeles

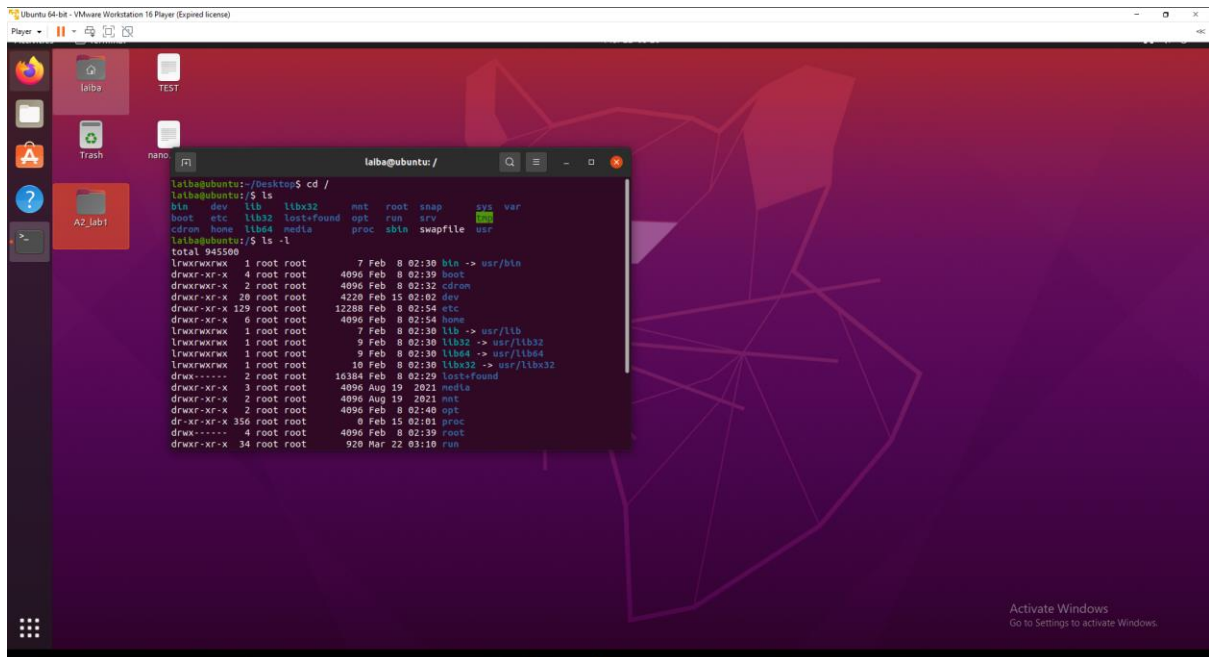
gnome-initial-setup:x:124:65534::/run/gnome-initial-setup:/bin/false
gdm:x:125:130:Gnome Display Manager:/var/lib/gdm3:/bin/false
sssd:x:126:131:SSSD system user,,,:/var/lib/sss:/usr/sbin/nologin
laiba:x:1000:1000:laiba,,,:/home/laiba:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin
user1:x:1001:1001:user1,,,:/home/user1:/bin/bash
user2:x:1002:1002:user2,,,:/home/user2:/bin/bash
user3:x:1003:1003:user3,,,:/home/user3:/bin/bash
laiba@ubuntu:~/A2_LAB1$
```

For your sample program, above, describe the impact of lines 1, 2, and 4:

Ans:

- 1: This command `cd /etc` changes directory to the directory specified after the slash /. The /etc refers to a folder in the root called etc .
- 2: To view your current timezone you cat the file's contents.
- 4: will show the tail of the file skipping 8 first lines.

Question2:

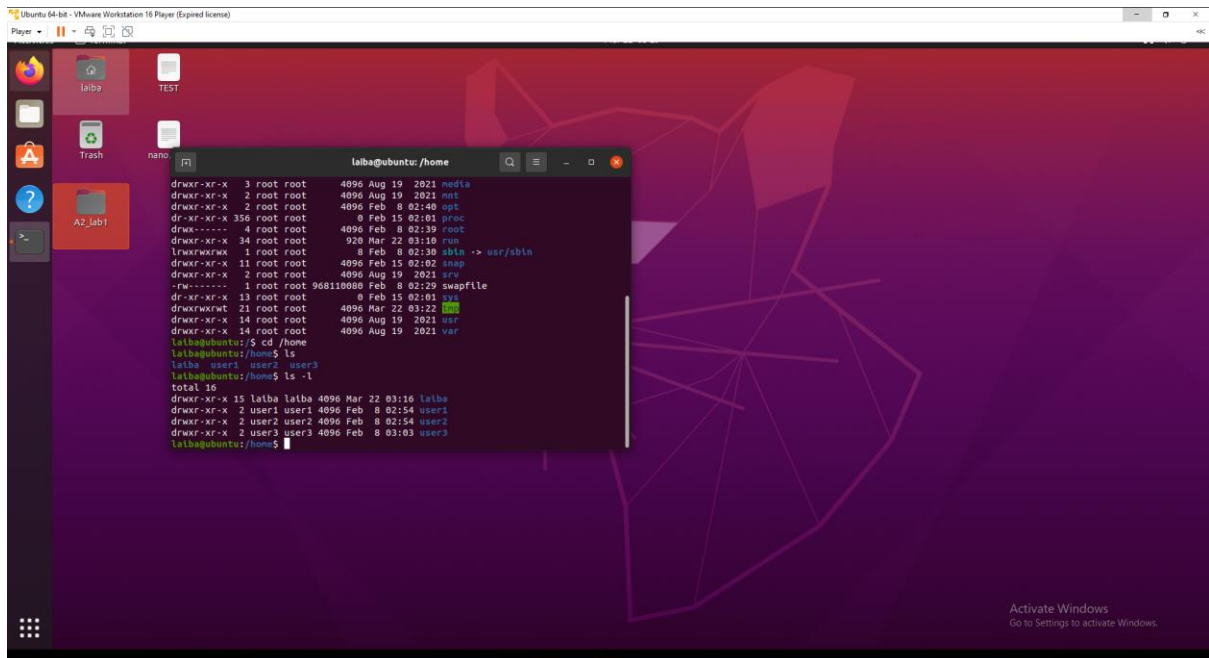


The screenshot shows a terminal window titled 'lalba@ubuntu: /' with the following output:

```
lalba@ubuntu:~/desktop$ cd /
lalba@ubuntu:/ $ ls
bin  dev  lib  lib32  mnt  root  snap  sys  var
boot  etc  lib32  lost+found  opt  run  srs
cdrom  home  lib64  media  proc /sbin  swapfile  usr
lalba@ubuntu:/ $ ls -l
total 945500
lrwxrwxrwx 1 root root      7 Feb  8 02:30 bin -> usr/bin
drwxr-xr-x 4 root root    4096 Feb  8 02:39 boot
drwxr-xr-x 2 root root    4096 Feb  8 02:32 cdrom
drwxr-xr-x 20 root root  4220 Feb 15 02:02 dev
drwxr-xr-x 129 root root 12288 Feb  8 02:54 etc
drwxr-xr-x 6 root root    4096 Feb  8 02:54 home
lrwxrwxrwx 1 root root      7 Feb  8 02:30 lib -> usr/lib
lrwxrwxrwx 1 root root      9 Feb  8 02:30 lib32 -> usr/lib32
lrwxrwxrwx 1 root root      9 Feb  8 02:30 lib64 -> usr/lib64
lrwxrwxrwx 1 root root     10 Feb  8 02:30 lib32 -> usr/lib32
drwx----- 2 root root 16384 Feb  8 02:29 lost+found
drwxr-xr-x 3 root root    4096 Aug 19 2021 media
drwxr-xr-x 2 root root    4096 Aug 19 2021 mnt
drwxr-xr-x 2 root root    4096 Feb  8 02:40 opt
drwxr-xr-x 356 root root    0 Feb 15 02:03 proc
drwx----- 4 root root    4096 Feb  8 02:39 root
drwxr-xr-x 34 root root    920 Mar 22 03:10 run
```

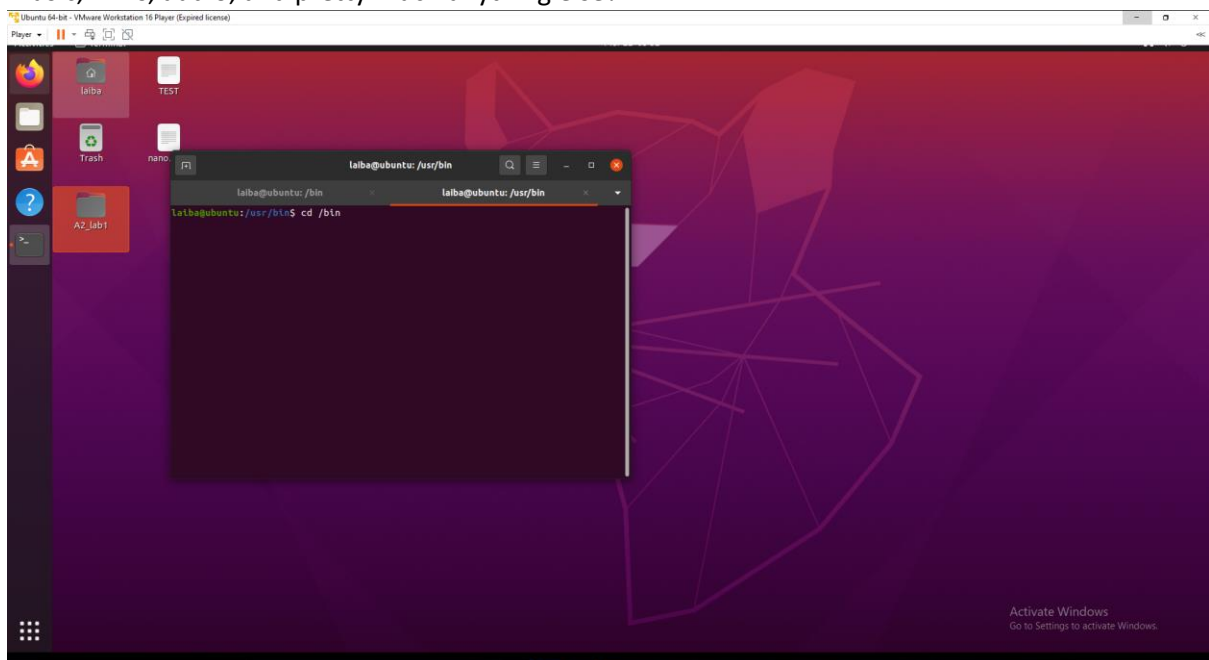
What is the purpose of the swapfile?

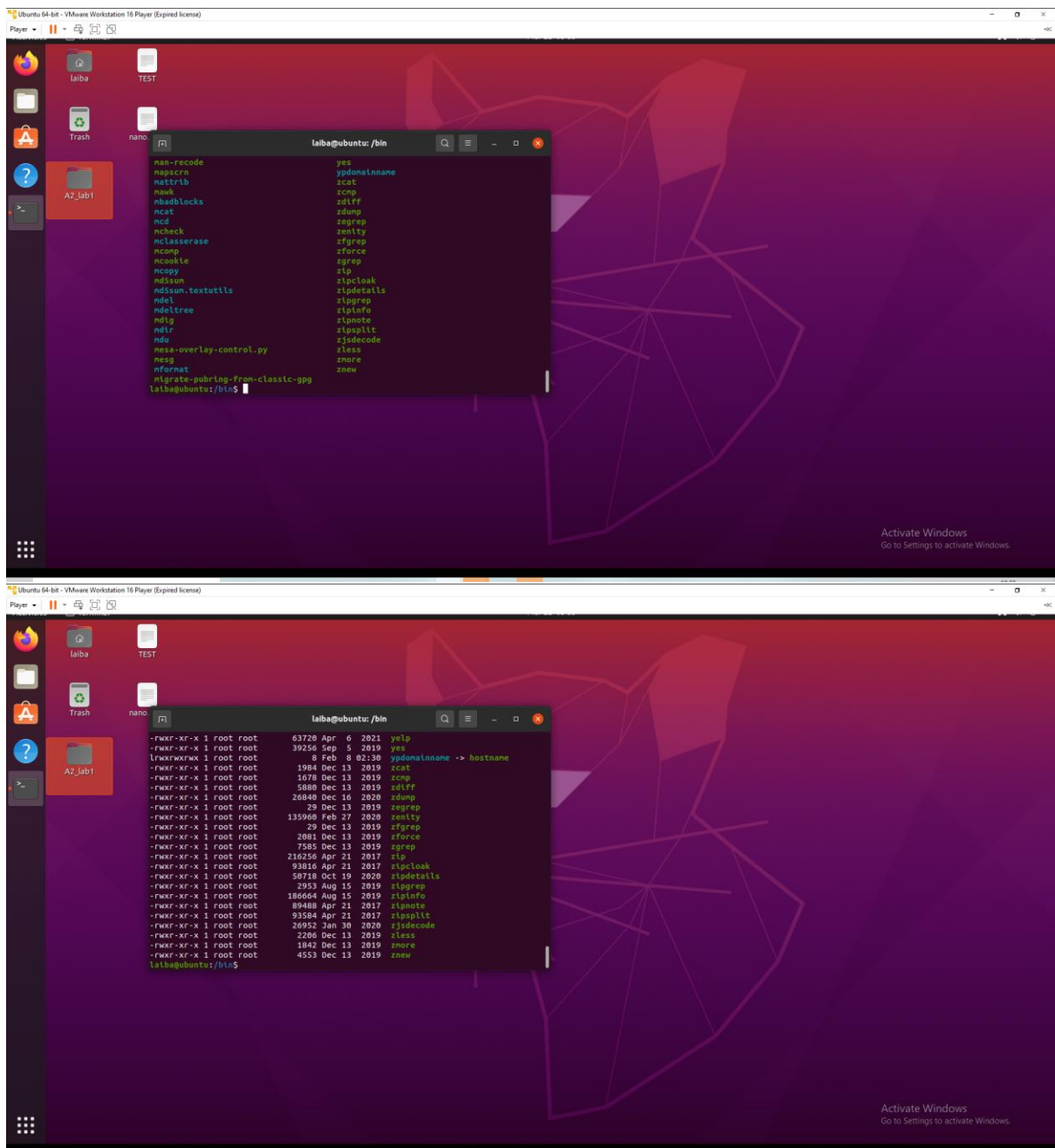
Ans: A swap file allows an operating system to mimic more memory by using hard drive space. When the system runs out of memory, it transfers a piece of RAM that is being used by an idle application onto the hard drive to make room for other processes. Ubuntu automatically creates a swap file of 2GB in size



What is the purpose of this directory?

Ans: This is the location of your computer's desktop. Where can you keep your papers, photos, music, films, audio, and pretty much anything else?





What is the purpose of this directory?

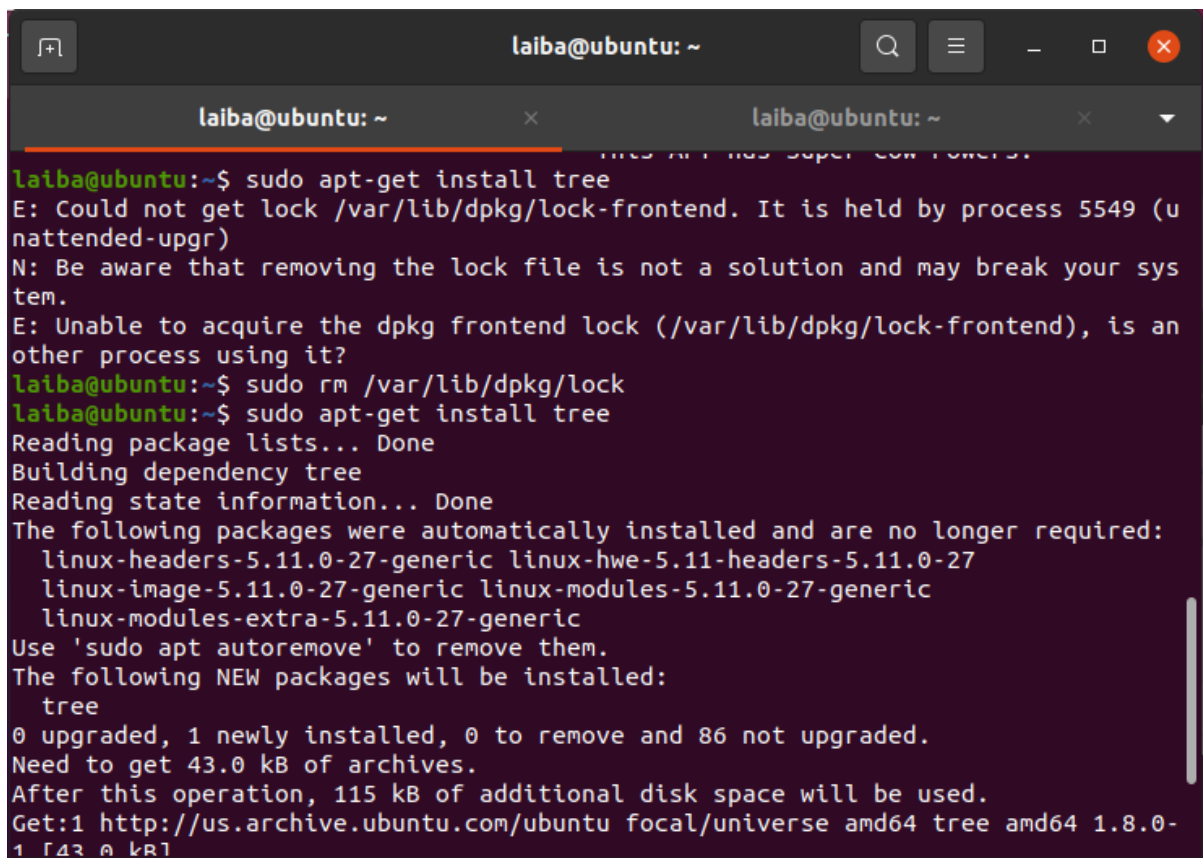
Ans: includes the executable (ready-to-run) applications that must be accessible in order for a system to boot (i.e., start) and be repaired.


```
laiba@ubuntu:~/A2_LAB1$ mkdir SubDirectoryB1
laiba@ubuntu:~/A2_LAB1$ mkdir TestDirC
laiba@ubuntu:~/A2_LAB1$ cd ~
laiba@ubuntu:~$ tree A2_LAB1
```

Command 'tree' not found, but can be installed with:

```
sudo apt install tree
```

```
laiba@ubuntu:~$ sudo apt-get
[sudo] password for laiba:
apt 2.0.6 (amd64)
Usage: apt-get [options] command
       apt-get [options] install|remove pkg1 [pkg2 ...]
       apt-get [options] source pkg1 [pkg2 ...]
```



```
laiba@ubuntu:~$ sudo apt-get install tree
E: Could not get lock /var/lib/dpkg/lock-frontent. It is held by process 5549 (u
nattended-upgr)
N: Be aware that removing the lock file is not a solution and may break your sys
tem.
E: Unable to acquire the dpkg frontend lock (/var/lib/dpkg/lock-frontent), is an
other process using it?
laiba@ubuntu:~$ sudo rm /var/lib/dpkg/lock
laiba@ubuntu:~$ sudo apt-get install tree
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  linux-headers-5.11.0-27-generic linux-hwe-5.11-headers-5.11.0-27
  linux-image-5.11.0-27-generic linux-modules-5.11.0-27-generic
  linux-modules-extra-5.11.0-27-generic
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  tree
0 upgraded, 1 newly installed, 0 to remove and 86 not upgraded.
Need to get 43.0 kB of archives.
After this operation, 115 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 tree amd64 1.8.0-
1 [43.0 kB]
```


Question4:

```
laiba@ubuntu:~$ tail -3 /etc/group
user1:x:1001:
user2:x:1002:
user3:x:1003:
laiba@ubuntu:~$ tail -3 /etc/passwd
user1:x:1001:1001:user1,,,:/home/user1:/bin/bash
user2:x:1002:1002:user2,,,:/home/user2:/bin/bash
user3:x:1003:1003:user3,,,:/home/user3:/bin/bash
laiba@ubuntu:~$ man tail
laiba@ubuntu:~$
```

Describe the functionality of the tail command:

Ans:

- Print the last 10 lines of each FILE to standard output. With more than one FILE, precede each with a header giving the file name.
- With no FILE, or when FILE is -, read standard input.
- Mandatory arguments to long options are mandatory for short options too.

What is the purpose of the group and passwd files?

Ans:

Group files: extra skills, such as access to disks, printers, and other peripherals, may be allocated in an orderly manner.

Passwd file: to keep track of every person who has been granted access to a system

Explain the fields of the last line of the group file:

Ans:

1. group-name : Contains the name assigned to the group.
2. group-password: Usually contains an asterisk or is empty.
3. gid: Contains the group's GID number.
4. user-list: Contains a comma-separated list of user names, representing the user's secondary group memberships.

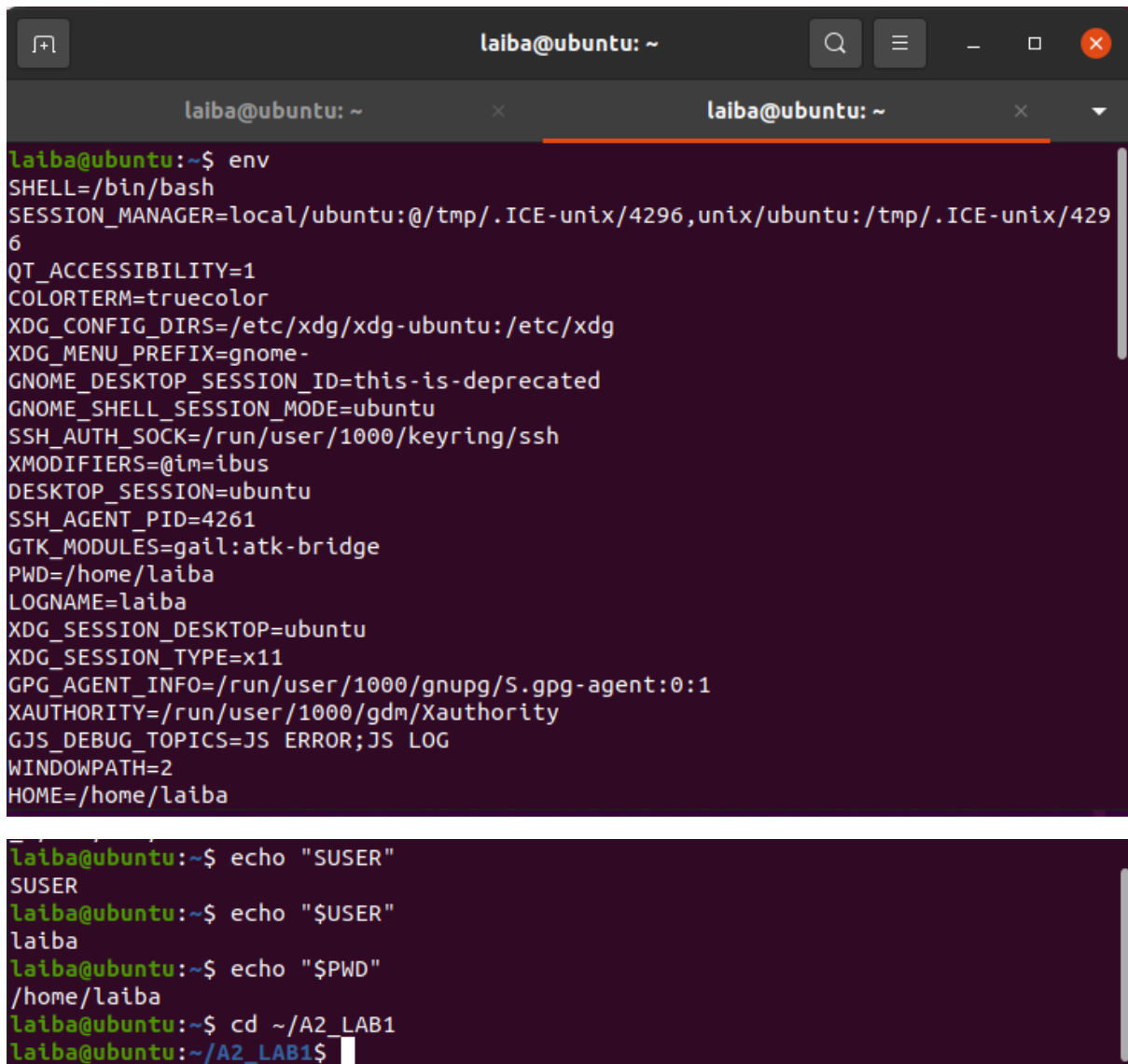
Explain the fields of the last line of the passwd file:

Ans:

1. Username: It is used when user logs in.

2. Password: An x character indicates that encrypted password is stored in /etc/shadow file.
3. User ID (UID): Each user must be assigned a user ID (UID).
4. Group ID (GID): The primary group ID (stored in /etc/group file)
5. User ID Info (GECOS): The comment field.
6. Home directory: The absolute path to the directory the user will be in when they log in.
7. Command/shell: The absolute path of a command or shell (/bin/bash).

Question5:



```
laiba@ubuntu: ~  
laiba@ubuntu: ~  
laiba@ubuntu:~$ env  
SHELL=/bin/bash  
SESSION_MANAGER=local/ubuntu:@/tmp/.ICE-unix/4296,unix/ubuntu:/tmp/.ICE-unix/4296  
QT_ACCESSIBILITY=1  
COLORTERM=truecolor  
XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/etc/xdg  
XDG_MENU_PREFIX=gnome-  
GNOME_DESKTOP_SESSION_ID=this-is-deprecated  
GNOME_SHELL_SESSION_MODE=ubuntu  
SSH_AUTH_SOCK=/run/user/1000/keyring/ssh  
XMODIFIERS=@im=ibus  
DESKTOP_SESSION=ubuntu  
SSH_AGENT_PID=4261  
GTK_MODULES=gail:atk-bridge  
PWD=/home/laiba  
LOGNAME=laiba  
XDG_SESSION_DESKTOP=ubuntu  
XDG_SESSION_TYPE=x11  
GPG_AGENT_INFO=/run/user/1000/gnupg/S.gpg-agent:0:1  
XAUTHORITY=/run/user/1000/gdm/Xauthority  
GJS_DEBUG_TOPICS=JS ERROR;JS LOG  
WINDOWPATH=2  
HOME=/home/laiba  
  
laiba@ubuntu:~$ echo "SUSER"  
SUSER  
laiba@ubuntu:~$ echo "$USER"  
laiba  
laiba@ubuntu:~$ echo "$PWD"  
/home/laiba  
laiba@ubuntu:~$ cd ~/A2_LAB1  
laiba@ubuntu:~/A2_LAB1$
```

```
laiba@ubuntu:~/A2_LAB1$ nano MyProg2
laiba@ubuntu:~/A2_LAB1$ bash MyProg2
MyProg2: line 1: XDG_VTNR: command not found
MyProg2: line 2: SUDO_UID: command not found
MyProg2: line 3: MAIL: command not found
MyProg2: line 4: PATH: command not found
MyProg2: line 5: XDG_SESSION: command not found
MyProg2: line 6: SHELL: command not found
laiba@ubuntu:~/A2_LAB1$
```

Describe each of the environment variables displayed by your program:

Ans:

1. XDG_VTNR: specifies the VT number.
2. SUDO_UID: Set to the uid of the user who invoked sudo.
3. MAIL: is a command-line utility that is used to send and manage the emails from the command line.
4. PATH: is an environmental variable that tells the shell and other programs which directories to search for executable files.
5. XDG_SESSION: is used to open a file or URL in the user's preferred application.
6. SHELL: provides an interface between the user and the kernel and executes programs called commands.

Question6:

Describe the `|`, `>` and `>>` operators in your own words:

Ans:

`|`: Binary OR Operator copies a bit if it exists in either operand.

The output (STDOUT) direction operators "`>`" and "`>>`" are both output (STDOUT) direction operators, however they differ in the following ways:

"`>`" overwrites an existing file or creates a new one if the specified file name does not exist in the directory.

If the "`>>`" operator is used, it appends an existing file or creates a new one if the specified file name does not exist in the directory.

```
laiba@ubuntu: ~/A2_LAB1
laiba@ubuntu:~/Desktop$ cd ~/A2_LAB1
laiba@ubuntu:~/A2_LAB1$ tail -12 /etc/group | sort >> Result.txt
laiba@ubuntu:~/A2_LAB1$ ls
Myprog1  MyProg2  Result.txt  SubDirectoryB1  TestDirA  TestDirB  TestDirC
laiba@ubuntu:~/A2_LAB1$
```

Explain, in your own words, what this command, Line 3 above, is doing. Show snippets to help your explanations:

Ans: Instead of ending at the end of the file, the tail command publishes the data starting from the provided line number. Data will begin printed from line number 'n' until the end of the file given by the command tail +n file name.