SLB 361: Software Engineering Tools Lab, MIDTERM EXAMINATION
Fall 2020 – CLOSED Book Exam –Total marks: 20 – Duration: 60 mins
This exam comes in five pages

STUDENT NAME _____

STUDENT ID # _____

Question 1 [7 marks]

1- ……../sndconfig file is responsible for Configuration of the sound card and sound events.

a- home

b- user/bin

c- etc

d- root

2- Which of the following are benefits of open source software for the user?

a. Code can survive the loss of the original developer or distributor.

b. Sensitive portions of code are protected and only available to the original developer.

c. Come with extensive support.

d. Code remains open as long as it is in a public repository but the license may change when included with closed source software.

3- Which statement is not a benefit of Linux?

a. Linux is developed entirely by volunteers making it a low cost operating system.

b. Linux is modular and can be configured as a full graphical desktop or a small appliance.

c. You can download the source code and change it any way you like.

d. Linux includes a powerful and scriptable command-line interface, enabling easier automation and provisioning.

4- Which term describes the part of the command line that adjusts the behavior of a command?

a. Argument

b. Command

c. Option

d. Prompt

5- Which term describes one of multiple logical consoles that can each support an independent login session?

a. Physical Console

b. Virtual Console

c. Shell

d. Terminal

6. Which Bash shortcut or command separates commands on the same line?

a. Pressing Tab

b. history

c. ;

d. !string

e. Pressing Esc+.

7. Which Bash shortcut or command displays the list of previous commands?

a. Pressing Tab

b. !string

c. !number

d. history

e. Pressing Esc+.

8. Which directory contains installed software programs and libraries?

a. /etc

b. /lib

c. /usr

d. /var

9. Which pattern will match only filenames that contain a number?

a. *#*

b. *[[:digit:]]*

c. *[digit]*

d. [0-9]

10. Which answer sends output and errors to the same file ensuring existing file content is preserved?

a. >file 2>file2

b. &>file

c. >>file 2>&1

d. >>file 1>&1

11- Using "No Login Shell" helps in:

a. Performance

b. Easy login

c. Security

d. Administration

Use the following information to solve the next questions

• User consultant1 is in groups consultant1 and database1

• User operator1 is in groups operator1 and database1

• User contractor1 is in groups contractor1 and contractor3

• User operator2 is in groups operator2 and contractor3

The current directory (.) contains four files with the following permissions information:

drwxrwxr-x. operator1 database1 .

-rw-rw-r--. consultant1 consultant1 lfile1

-rw-r--rw-. consultant1 database1 lfile2

-rw-rw-r--. operator1 database1 rfile1

-rw-r-----. operator1 database1 rfile2

12. Which file can be modified by the contractor1 user?

a. lfile1

b. lfile2

c. rfile1

d. rfile2

13. Which files can be deleted by the operator1 user?

a. rfile1

b. rfile2

c. All of the above.

d. None of the above.

14- Which of the following commands will allow user 'operator1' to edit and read 'lfile2'

a- chmod u=rw lfile2

b- chmod g=w lfile2

c- chmod 757 lfile2

d- chmod g+w lfile2

Question 2 [13 marks]: Apply the following actions using the RedHat Linux commands

1- Add a new user called **'miduser'** and set his password to **'rh-2020'**.

sudo useradd miduser ; passwd miduser **rh-2020**

2- Print the current user information to the screen and append it to **'userinfo.txt'** file using one command. Then you should check that it is saved in the file.

id | **tee -a** userinfo.txt
cat userinfo.txt

3- Grant superuser access to the **'miduser'** user. (Hint: you can edit /etc/sudoers file using visudo command by opening it in vim editor)

visudo
Press "i" : edit "/etc/sudoers" using vim insert mode
Add this line miduser All=(All) All
Press "ESC"
then :wq to save and close

4- Create a new group called **'midtermgroup'** then

- Add it as a secondary group to the root user
- Add it as a primary group to '**miduser**'

groupadd midtermgroup
usermod **-aG** midtermgroup **root**
usermod **-g** midtermgroup **miduser**

5- Using the root user to create a new directory **'/home/miduser/mid term/SE lab'** in **one** command then:

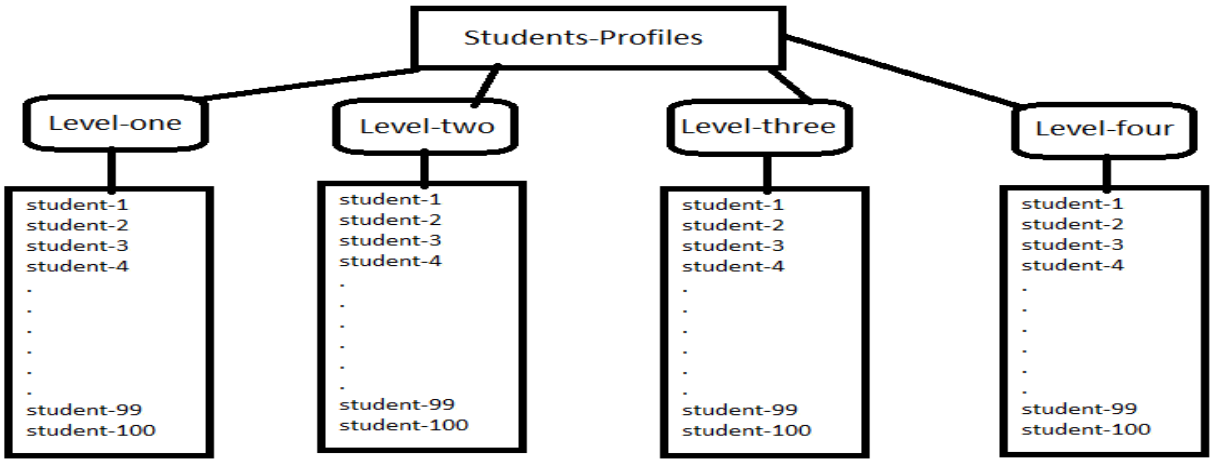
- Change the owner group of this directory to be the **'midtermgroup'** group
- Set the directory access permission to allow:
  - The owner user: read, write, and execute
  - The owner group: read and write
  - The others: read
- List the directory information to make sure that the access permissions are applied

* mkdir **-p** '/home/miduser/mid term/SE lab'
* **chown** :midtermgroup '/home/miduser/mid term/SE lab'
* **chmod 764** '/home/miduser/mid term/SE lab'
* ls **-ld** '/home/miduser/mid term/SE lab'

6- Add the **sgid** permission to **'SE lab'** directory
chmod g+s '/home/miduser/mid term/SE lab'

7- Prevent users from removing files that they don't own.
chmod o+t '/home/miduser/mid term/SE lab'

8- Create the following file structure in **one command** using expansions.



The **Students-Profiles** directory contains 4 directories **Level-one, Level-two, Level-three, Level-four**. Inside each directory there are 100 directories named as **student-1** to **student-100**

mkdir -p Students-Profiles/Level-{one,two,three,four}/student-{1..100}

9- Change your current directory to be at the "**Level-one"** directory. Then list its contents in detail to see their access permissions and display the output page by page

cd Level-one
ls -l | more (less)

10- Remove the write access from group permissions for any file or directory that ends with 0 (i.e. student-10, student-20) in one command directory then list all the directory contents in details to make sure that permissions are updated successfully

```
chmod g-w  *0
```

11- Go to the parent directory of your current location (don't use absolute path) and print current directory (it should be '**Students-Profiles**') then remove the '**Level-four**' directory

```
cd ..
rm -r Level-four
```

12- Print the first 20 command you have executed in the terminal to a file named as **'first20commands.txt'** and if any error happened redirect it to the **'errorlog.txt'** file

```
history | head -20 > first20commands.txt  2 > errorlog.txt
```