

CYSE 270: Linux System for Cybersecurity
Lab 8 – Shell Scripting
(Total 100 Points)

Please refer to the slides for week 8 - Shell scripting and write shell scripts to complete the following tasks. Submit the screenshot for the script and its output, both.

NOTE: Please replace the name of the script with the name you used for the script. In the sample screenshot, I have used those names to create my script.

Step-1: Use vi or nano editor to write your script (Ex, vi YourScriptName.sh) for the following tasks.

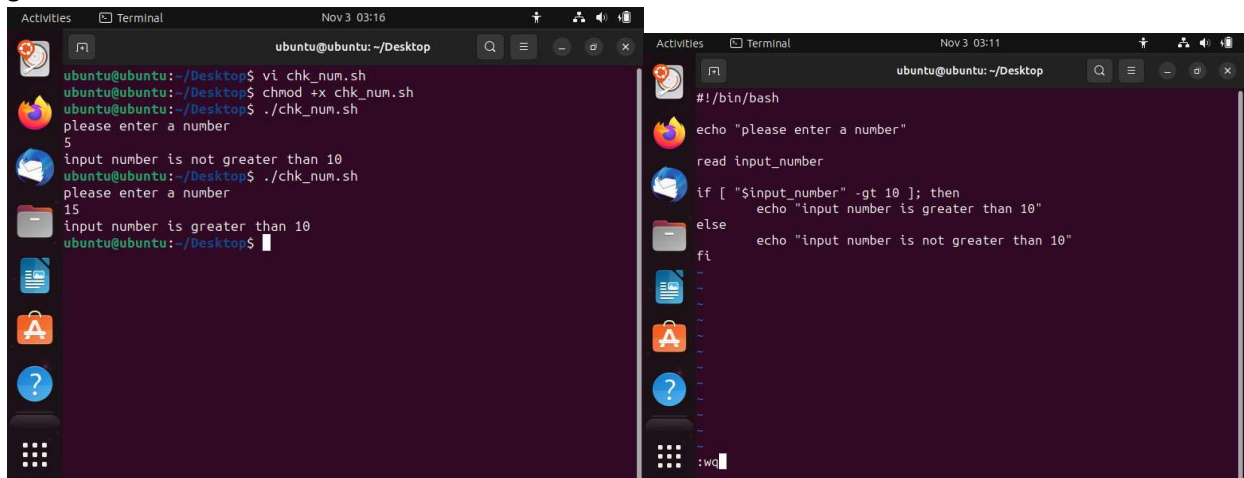
Step-2: After saving the script, save and exit out of the editor and make the script executable by adding execute permission (chmod +x YourScriptName.sh)

Step-3: Run your script using ./YourScriptName.sh

Task A (Correct script (25 points) + result/output after executing the script (25 points)-
Conditional Statement

Write a shell script using nano or vi editor (eg, vi scriptname.sh) like below, that performs the following task:

1. Add the Shebang (#!/bin/bash) as the first line in your script.
2. Read a number using read function
3. Using if statement, check if the input number is greater than 10, then print the message “Input number is greater than 10”.
4. If the number is not greater than 10, then print the message, “Input number is not greater than 10”.



```
ubuntu@ubuntu: ~/Desktop
ubuntu@ubuntu:~/Desktop$ vi chk_num.sh
ubuntu@ubuntu:~/Desktop$ chmod +x chk_num.sh
ubuntu@ubuntu:~/Desktop$ ./chk_num.sh
please enter a number
5
input number is not greater than 10
ubuntu@ubuntu:~/Desktop$ ./chk_num.sh
please enter a number
15
input number is greater than 10
ubuntu@ubuntu:~/Desktop$
```

```
#!/bin/bash
echo "please enter a number"
read input_number
if [ "$input_number" -gt 10 ]; then
    echo "input number is greater than 10"
else
    echo "input number is not greater than 10"
fi
```

Task B (Correct script (25 points) + result/output after executing the script (25 points) -
Shell Script to Create a new file

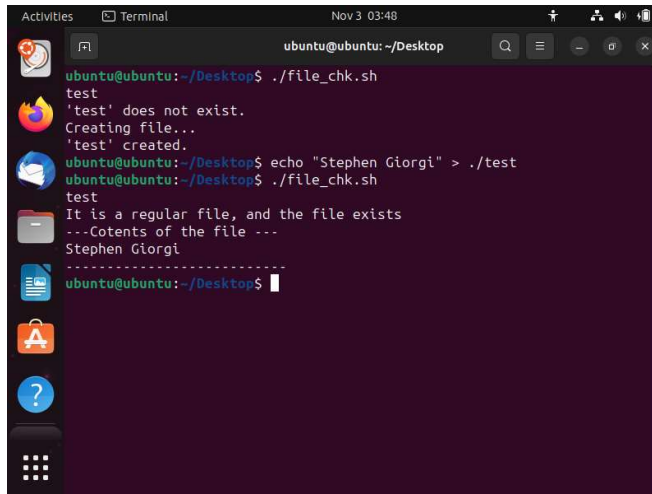
Write a shell script using nano or vi editor (eg, nano scriptname.sh) like below, that performs the following task:

1. Add the Shebang (#!/bin/bash) as the first line in your script.
2. Reads the name of the file to check for a filename that exists.
3. Check whether the given input is a directory or regular file.
4. If the input is a directory and exists, then display the message “Directory exists”.

5. If the input is a regular file, then display the message "It is a regular file, and the file exists" and display the contents of the file.
6. If the given input name in step-1 doesn't exist, then create the new file with the given name in step-1.

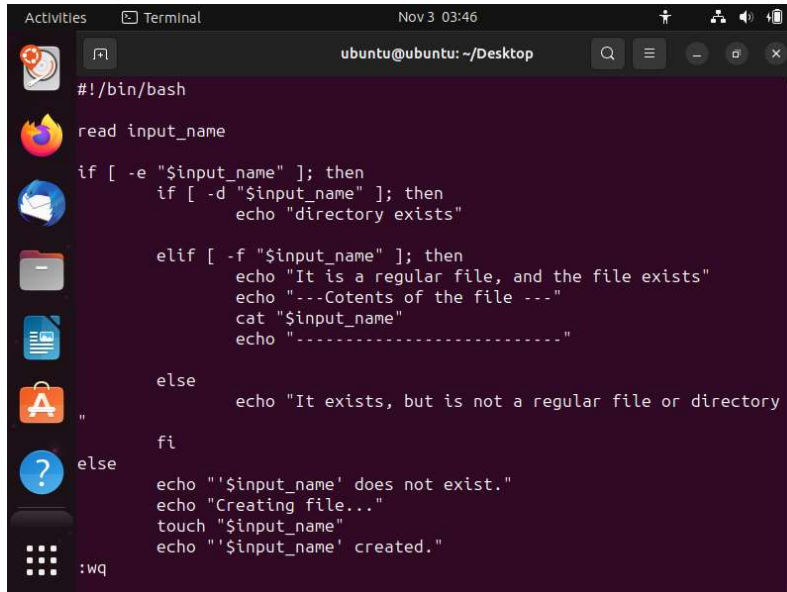
(Extra credit: 10 points) Add your name to the file (using redirection operator '>') and display the contents for the newly created file.

7. Save and exit the editor and remember to make the script executable using the command `chmod +x scriptname.sh`



A terminal window showing the execution of a script named `file_chk.sh`. The user runs `./file_chk.sh` with the input `test`. The script outputs: `'test' does not exist.`, `Creating file...`, and `'test' created.`. Then, the user runs `echo "Stephen Giorgi" > ./test`. Finally, the user runs `./file_chk.sh` with the input `test`. The script outputs: `It is a regular file, and the file exists`, `---Cotents of the file ---`, and `Stephen Giorgi`.

```
ubuntu@ubuntu: ~/Desktop
ubuntu@ubuntu:~/Desktop$ ./file_chk.sh
test
'test' does not exist.
Creating file...
'test' created.
ubuntu@ubuntu:~/Desktop$ echo "Stephen Giorgi" > ./test
ubuntu@ubuntu:~/Desktop$ ./file_chk.sh
test
It is a regular file, and the file exists
---Cotents of the file ---
Stephen Giorgi
ubuntu@ubuntu:~/Desktop$
```



A terminal window showing the source code of the `file_chk.sh` script. The code uses `#!/bin/bash`, `read` to get input, and `if` statements to check if the input is a directory, a regular file, or does not exist. It uses `echo` for output and `cat` to display file contents.

```
#!/bin/bash
read input_name

if [ -e "$input_name" ]; then
    if [ -d "$input_name" ]; then
        echo "directory exists"

    elif [ -f "$input_name" ]; then
        echo "It is a regular file, and the file exists"
        echo "---Cotents of the file ---"
        cat "$input_name"
        echo "-----"

    else
        echo "It exists, but is not a regular file or directory"
    fi
else
    echo "'$input_name' does not exist."
    echo "Creating file..."
    touch "$input_name"
    echo "'$input_name' created."

:wq
```