FOSS LAB REPORT

Neethu S
Roll no: 42
TVE18CS043

College of Engineering Trivandrum
31 Jan 2020

1 Experiment 4

Shell programming

1.1 Aim

Write a shell script to show various system configuration like

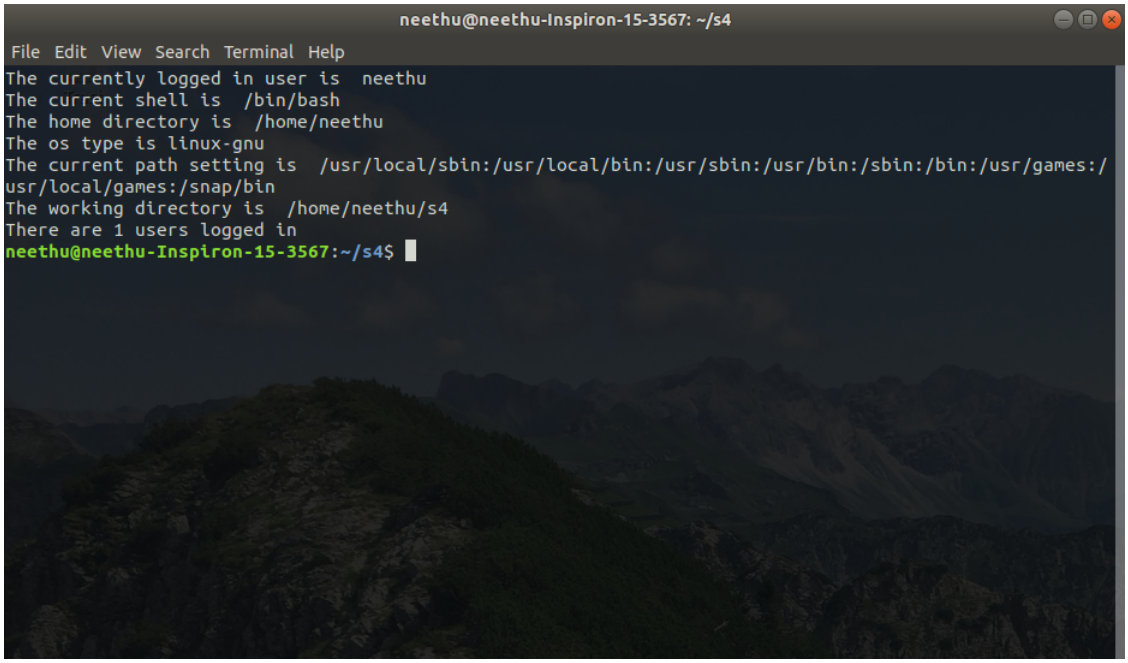
1. Currently logged user and his login name
2. Your current shell
3. Your home directory
4. Your operating system type
5. Your current path setting
6. Your current working directory
7. Number of users currently logged in

1.2 Overview

Shell script

```
#!/bin/bash
clear
log=‘who|wc -l‘
echo "The currently logged in user is  $USER "\
echo "The current shell is  $SHELL "\
echo "The home directory is  $HOME "\
echo "The os type is $OSTYPE "\
echo "The current path setting is  $PATH "\
echo "The working directory is  $PWD "\
echo "There are $log users logged in "\
```

Sample input and output:



1.3 Result

The shell script for displaying various system configurations was made and the output was verified. The shell script was run on Ubuntu 18.04.

2 Experiment 5

Shell script to show various system configurations

2.1 Aim

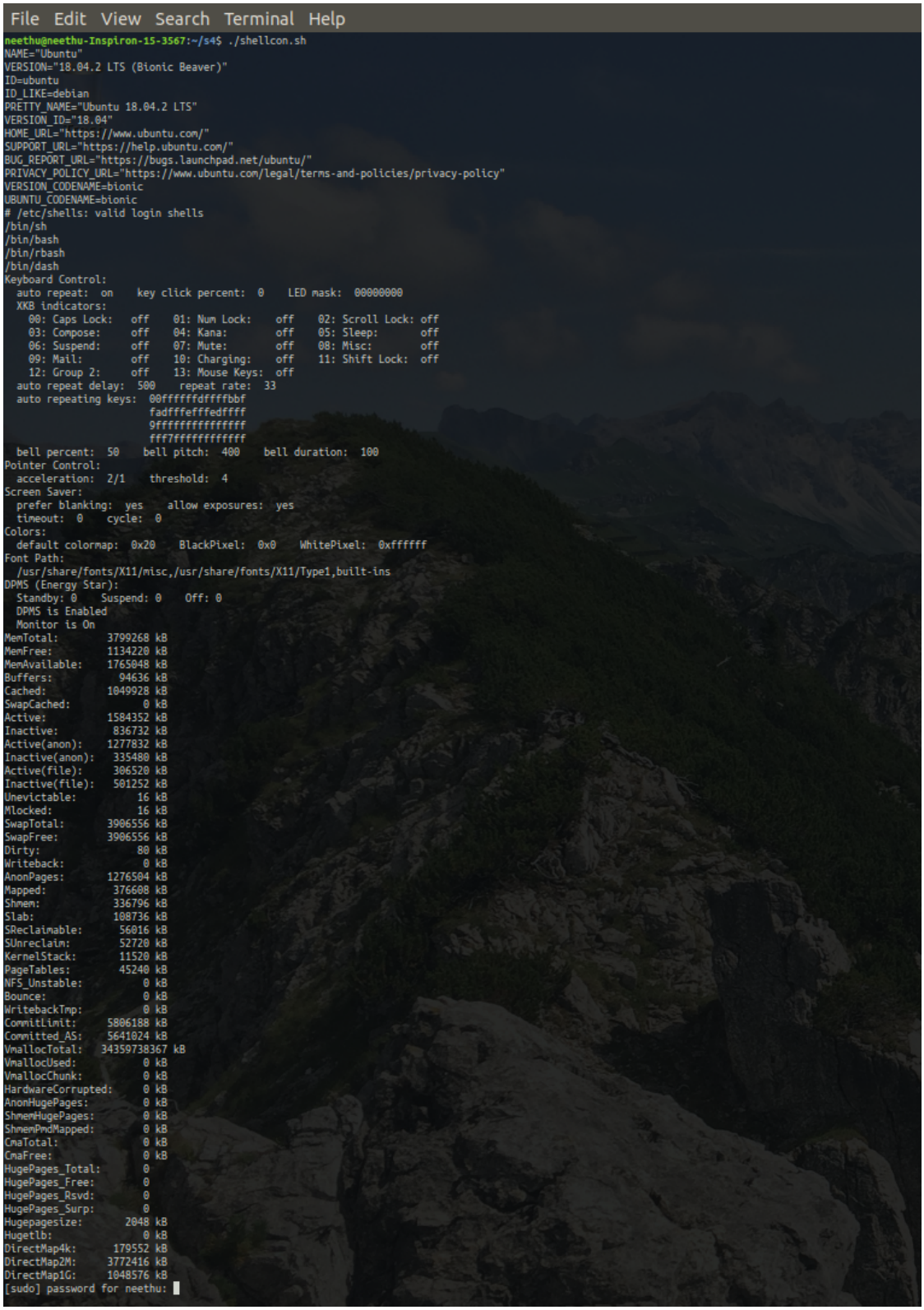
Write a shell script to show various system configurations like
1. your OS and version, release number, kernel version
2. all available shells
3. computer CPU information like processor type, speed etc
4. memory information
5. hard disk information like size of hard-disk, cache memory, model etc
6. File system (Mounted)

2.2 Overview

Shell accept human readable commands from user and convert them into something which kernel can understand. It is a command language interpreter that execute commands read from input devices such as keyboards or from files.
Shell script

```
#!/ bin/ bash
echo -e "‘ cat /etc/os - release ‘"
echo -e "‘ cat /etc/shells ‘"
echo -e "‘ xset q ‘"
echo -e "‘ cat / proc / meminfo ‘"
echo -e " Driver : ‘sudo hdparm -I /dev /sda ‘"
echo -e "‘ cat / proc /mounts ‘"
```

Sample input and output:



```
File  Edit  View  Search  Terminal  Help
neethu@neethu-Inspiron-15-3567:~/s4$ ./shellcon.sh
NAME="Ubuntu"
VERSION="18.04.2 LTS (Bionic Beaver)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 18.04.2 LTS"
VERSION_ID="18.04"
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
VERSION_CODENAME=bionic
UBUNTU_CODENAME=bionic
# /etc/shells: valid login shells
/bin/sh
/bin/bash
/bin/rbash
/bin/dash
Keyboard Control:
  auto repeat:  on    key click percent:  0    LED mask:  00000000
  XKB indicators:
    00: Caps Lock:   off    01: Num Lock:    off    02: Scroll Lock: off
    03: Compose:     off    04: Kana:        off    05: Sleep:       off
    06: Suspend:     off    07: Mute:        off    08: Misc:        off
    09: Mail:        off    10: Charging:    off    11: Shift Lock:  off
    12: Group 2:     off    13: Mouse Keys:  off
  auto repeat delay:  500    repeat rate:  33
  auto repeating keys:  00ffffffdffffbbf
                        fadffffffedffff
                        9fffffffffffffff
                        fff7ffffffffffff
  bell percent:  50    bell pitch:  400    bell duration:  100
Pointer Control:
  acceleration:  2/1    threshold:  4
Screen Saver:
  prefer blanking:  yes    allow exposures:  yes
  timeout:  0    cycle:  0
Colors:
  default colormap:  0x20    BlackPixel:  0x0    WhitePixel:  0xffffff
Font Path:
  /usr/share/fonts/X11/misc,/usr/share/fonts/X11/Type1,built-ins
DPMS (Energy Star):
  Standby: 0    Suspend: 0    Off: 0
  DPMS is Enabled
  Monitor is On
MemTotal:        3799268 kB
MemFree:         1134220 kB
MemAvailable:    1765048 kB
Buffers:           94636 kB
Cached:          1049928 kB
SwapCached:            0 kB
Active:          1584352 kB
Inactive:         836732 kB
Active(anon):    1277832 kB
Inactive(anon):   335480 kB
Active(file):     306520 kB
Inactive(file):   501252 kB
Unevictable:          16 kB
Mlocked:              16 kB
SwapTotal:       3906556 kB
SwapFree:        3906556 kB
Dirty:                80 kB
Writeback:             0 kB
AnonPages:       1276504 kB
Mapped:           376608 kB
Shmem:            336796 kB
Slab:             108736 kB
SReclaimable:      56016 kB
SUnreclaim:        52720 kB
KernelStack:       11520 kB
PageTables:        45240 kB
NFS_Unstable:          0 kB
Bounce:                0 kB
WritebackTmp:          0 kB
CommitLimit:     5806188 kB
Committed_AS:    5641024 kB
VmallocTotal:   34359738367 kB
VmallocUsed:           0 kB
VmallocChunk:          0 kB
HardwareCorrupted:     0 kB
AnonHugePages:         0 kB
ShmemHugePages:        0 kB
ShmemPmdMapped:        0 kB
CmaTotal:              0 kB
CmaFree:               0 kB
HugePages_Total:       0
HugePages_Free:        0
HugePages_Rsvd:        0
HugePages_Surp:        0
Hugepagesize:       2048 kB
Hugetlb:               0 kB
DirectMap4k:      179552 kB
DirectMap2M:     3772416 kB
DirectMap1G:     1048576 kB
[sudo] password for neethu:
```

2.3 Result

The shell script for displaying required system configurations was made and the output was verified. The shell script was run on Ubuntu 18.04.

3 Experiment 6

Menu driven calculator

3.1 Aim

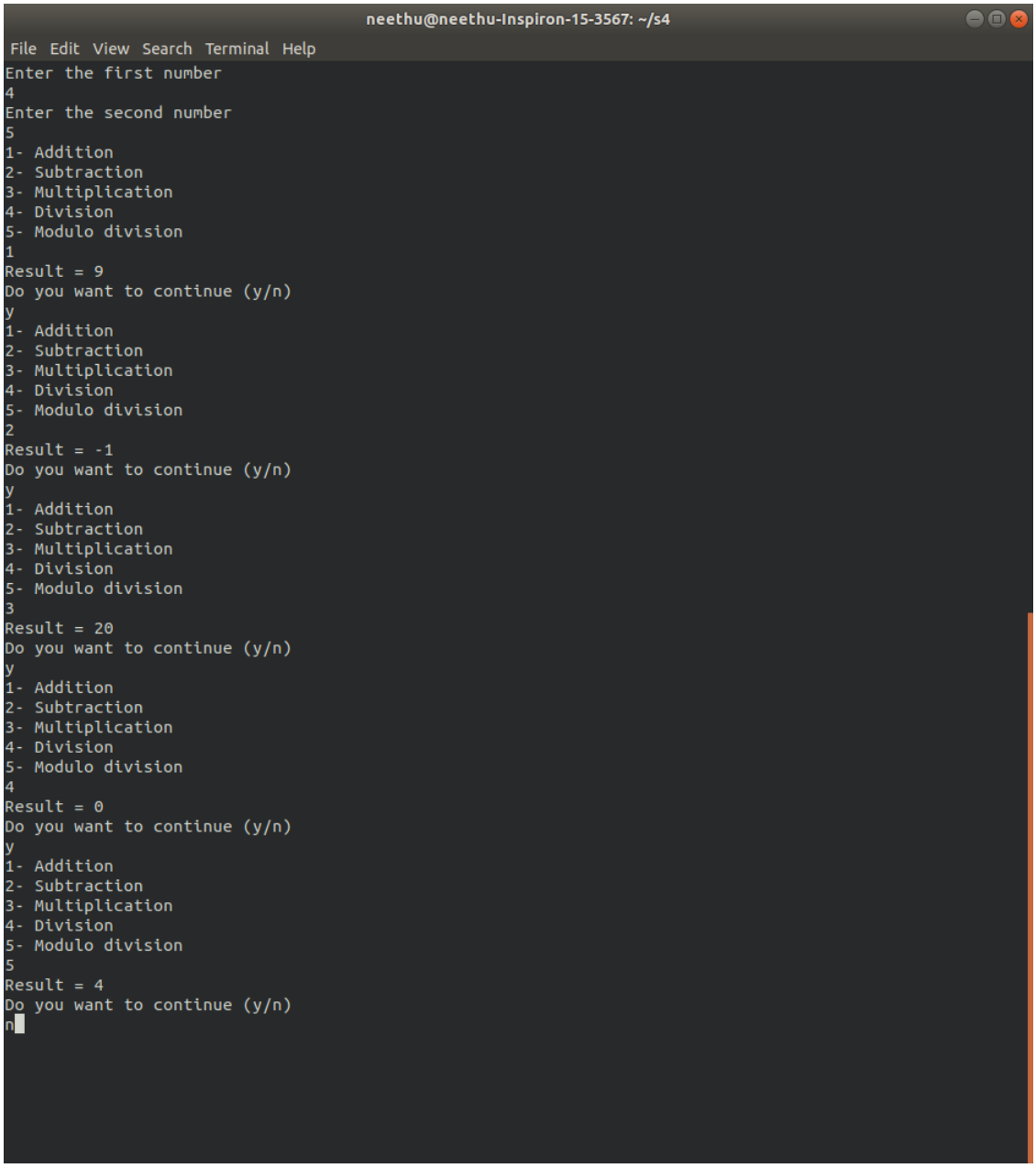
Write a shell script to implement a menu driven calculator with following functions
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Modulus

3.2 Overview

Shell script

```
i="y"
echo "Enter the first number "
read n1
echo "Enter the second number "
read n2
while [ $i = "y" ]
do
echo "1- Addition "
echo "2- Subtraction "
echo "3- Multiplication "
echo "4- Division "
echo "5- Modulo division "
read c
case $c in
1)sum=` expr $n1 + $n2 `
echo "Result =" $sum;;
2) sum=` expr $n1 - $n2 `
echo "Result =" $sum;;
3) sum=` expr $n1 \* $n2 `
echo "Result =" $sum;;
4) sum=` expr $n1 / $n2 `
echo "Result =" $sum;;
5) sum=` expr $n1 % $n2 `
echo "Result =" $sum;;
esac
echo "Do you want to continue (y/n)"
read i
if [ $i != "y" ]
then
exit
fi
done
```

Sample input and output:



```
neethu@neethu-Inspiron-15-3567: ~/s4
File  Edit  View  Search  Terminal  Help
Enter the first number
4
Enter the second number
5
1- Addition
2- Subtraction
3- Multiplication
4- Division
5- Modulo division
1
Result = 9
Do you want to continue (y/n)
y
1- Addition
2- Subtraction
3- Multiplication
4- Division
5- Modulo division
2
Result = -1
Do you want to continue (y/n)
y
1- Addition
2- Subtraction
3- Multiplication
4- Division
5- Modulo division
3
Result = 20
Do you want to continue (y/n)
y
1- Addition
2- Subtraction
3- Multiplication
4- Division
5- Modulo division
4
Result = 0
Do you want to continue (y/n)
y
1- Addition
2- Subtraction
3- Multiplication
4- Division
5- Modulo division
5
Result = 4
Do you want to continue (y/n)
n
```

3.3 Result

The shell script for a simple menu driven calculator was made and the output was verified. The script was run in Ubuntu 18.04. and screenshot of output is attached above.

4 Experiment 7

Script that accepts two arguments from the commad line and operates on them

4.1 Aim

Write a script called addnames that is to be called as follows ./addnames ulist username. Here ulist is the name of the file that contains list of usernames and username is a particular student's username. The script should
1. check that the correct number of arguments was received and print a message, in case the number of arguments is incorrect
2. check whether the ulist file exists and print an error message if it does not
3. check whether the username already exists in the file. If the username exists, print a message stating that the name already exists. Otherwise, add the username to the end of the list.

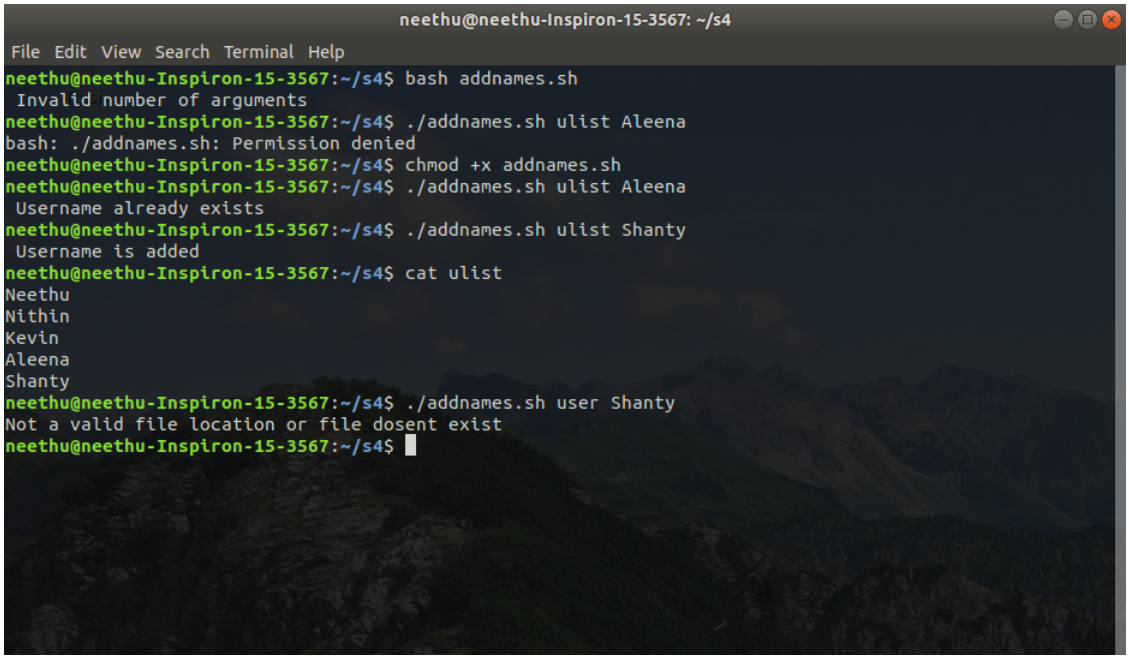
4.2 Overview

Shell script

```
if [[ $# -ne 2 ]]
then
echo " Invalid number of arguments "
exit
fi

if [[ ! (-a $1) ]]
then
echo "Not a valid file location or file dosent exist "
exit
fi
NO=$( grep -c -e $2 $1)
if [[ $NO -eq 0 ]]
then
echo $2 >> $1
echo " Username is added "
exit
else
echo " Username already exists "
exit
fi
```

Sample input and output:



4.3 Result

The required shell script was made and the output was verfied. The script was run on Ubuntu 18.04.

5 Experiment 9

5.1 Aim

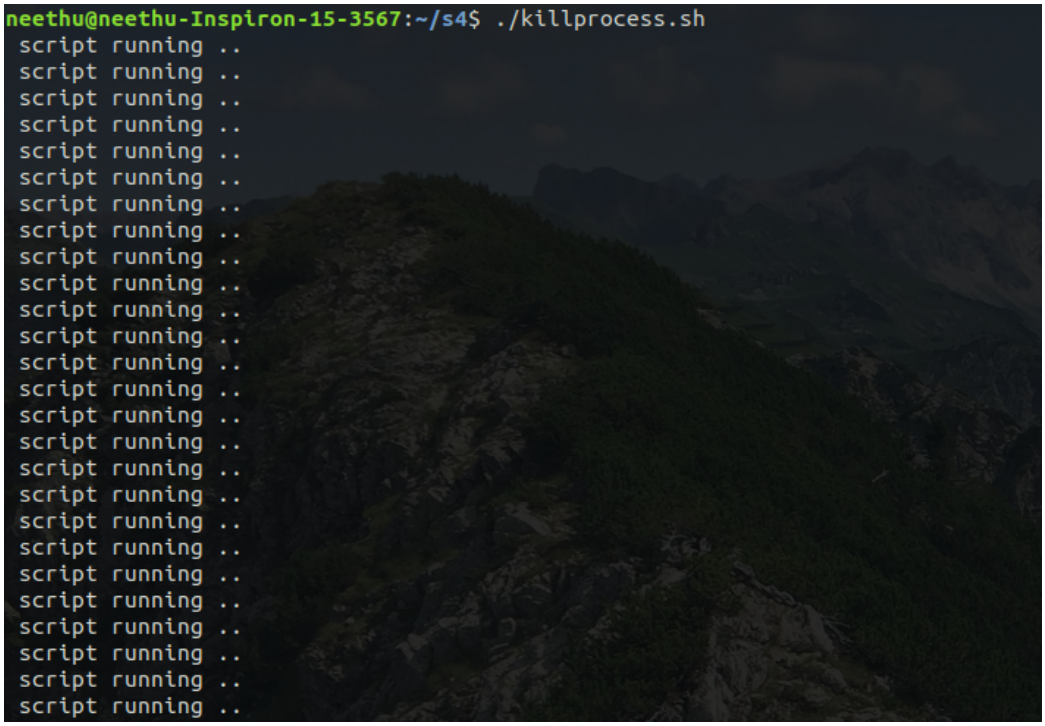
Write a Shell script which starts on system boot up and kills every process which uses more than a specified amount of memory or CPU.

5.2 Overview

Shell script

```
#!/ bin / sh
memlimit =10.0;
cpulimit =10.0;
while ( true )
do
echo " script running .."
ps -e -o pmem = , cpu = , pid = , user = , comm = | sort -r -k 1
while read size cpu pid user comm
do
kill_mem =0
kill_cpu =0
if [ "\ $user " = " neethu " ]
then
echo " Script Running ..."
kill_mem =‘ echo "$size$ > $$memlimit " $|$ bc ‘
kill_cpu =‘ echo "$cpu$ > $$cpulimit " $|$ bc ‘
if [ " $kill_mem = 1" ]
then
echo " process with PID $pid killed "
kill $pid
elif [ " $kill_cpu = 1" ]
then
echo " process with PID $pid killed "
kill $pid
else
continue
fi
fi
done
sleep 1
done
```

Sample input and output:



5.3 Result

A shell script for killing processes that consume more than a specific amount of memory and cpu was made and the output was verified. The shell script was run on Ubuntu 18.04.