Student id: 8971405

Course: Ethical Hacking & Cybersecurity

Module: 207SE Operating Systems, Security and Networks

Submission data: 28th Feb 2020

Portfolio 1

Contents

Lab Activity 1 – Operating Systems Tasks and Programming

a. Comparison between the Harvard Architecture and Von Neumann Architecture

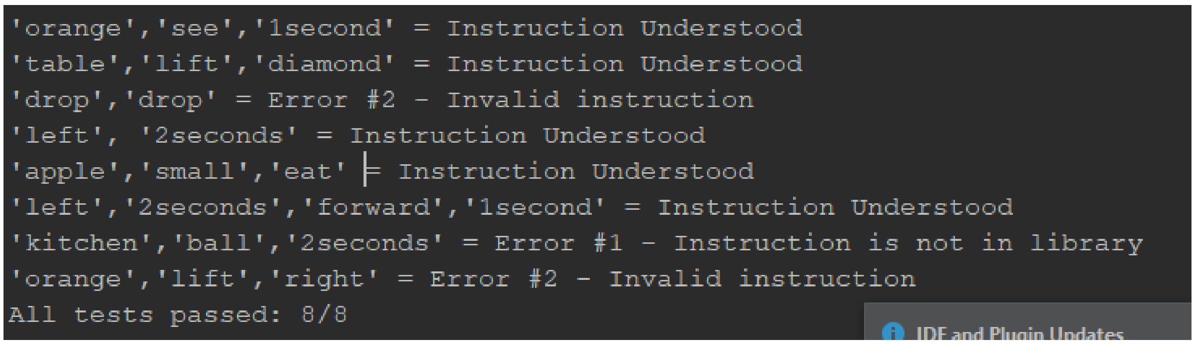
According to ARM Technical Support Knowledge Articles (2008), the Harvard Architecture the data and instruction busses are decoupled which allows for transfers to be executed at the same time. Whereas in a von Neumann Architecture, data and instruction transfers utilizes a single bus scheme. This forces the scheduling of data transfers and instruction fetches as they cannot be carried out simultaneously.

Due to this flaw in the von Neumann architecture, Bao (2017) quotes John Backus's 1977 ACM Turing Award lecture stating this method of transfer would eventually be the bottleneck of this architecture due to Moore's law, resulting in the "Memory Wall" problem (Xu et al. 2016). However, the von Neumann architecture is preferable as it employs a single unified cache approach which drives down development costs as well as lowers the complexity. This is where the Harvard architecture failed to perform. Due to the fact that computers generally require write access to the instruction memory space, it wouldn't be possible for a Harvard architecture to perform this. According to the ARM Articles (2008), unless data can be fed into the busses simultaneously, it is better to use a von Neumann Architecture.

b. Programming activity

```
'''
Baxter robot that takes in min of 2 arguments, max of 4
arguments which are from a predefined instruction set and
checks if the instruction is in a correct instruction syntax.
It then returns an output if the instruction given was
understood or failed.
'''
# Instruction Set Library
time = ['1second','2seconds','5seconds','unlimited']
move = ['left','right','forward','backward','stop']
object = ['orange','apple','car','bus','diamond']
action = ['recognise','eat','see','lift','drop','fetch']
size = ['small','big','little','massive']
location = ['door','kitchen','table']

# Grouping combos based on row, ie first row only has obj,
action, time
combo_0 = [object, action, time]
combo_1 = [object,size,action]
combo_2 = [move,time]
combo_3 = [move,time,move,time]
combo_4 = [location,action,object]

# Grouping insructions together as main lib
full_library = time + move + object + action + size +
location
```

```
# Checks which combination the user is inputing
def baxter(instruction1, *args):
    instruction_set = [instruction1]
    combo=[]
    for ar in args:
         instruction_set.append(ar)
    if len(instruction_set) < 2 or len(instruction_set) > 4:
         return("Error #0 - 2 to 4 arguements required!")
    for inst in instruction_set:
         if inst not in full_library:
              return("Error #1 - Instruction is not in
library")
if len(instruction_set) == 2:
   combo = combo_2
   elif instruction_set[0] in location:
        combo = combo_4
 elif instruction_set[0] in object:
      if instruction_set[1] in action:
                  combo = combo_0
      else:
             combo = combo_1
   else:
      combo = combo_3
   return (check_combo(instruction_set,combo))

# Checks if the combo inputed actually matches the
instruction lib
def check_combo(instruction_set,combo):
    for i in range(len(instruction_set)):
    if instruction_set[i] not in combo[i]:
    return("Error #2 - Invalid instruction")
    return('Instruction Understood')
```



```
'orange','see','1second' = Instruction Understood
'table','lift','diamond' = Instruction Understood
'drop','drop' = Error #2 - Invalid instruction
'left', '2seconds' = Instruction Understood
'apple','small','eat' = Instruction Understood
'left','2seconds','forward','1second' = Instruction Understood
'kitchen','ball','2seconds' = Error #1 - Instruction is not in library
'orange','lift','right' = Error #2 - Invalid instruction
All tests passed: 8/8
```

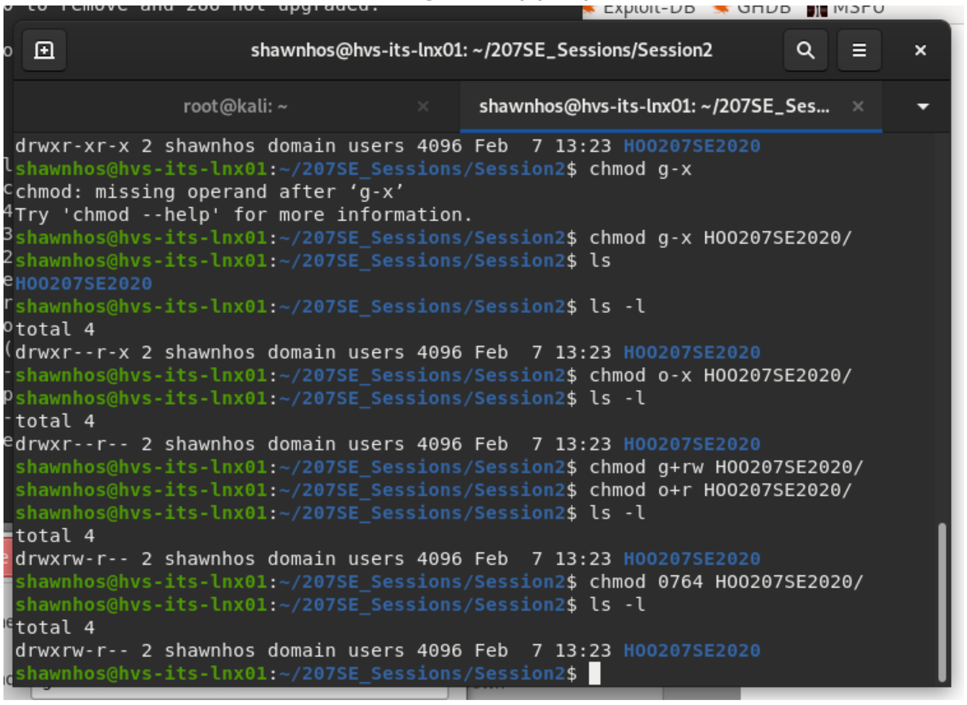
Lab Activity 2 – Linux Command Line (Commands and outcomes from a series of small tasks that require use of a number of Linux commands)

Tasks – Files

a. Create a directory in your area of the os-207SE server or your installation of Linux. The directory with a name made up of you second name followed by 207SE and the year (mine would be ELSHAW207SE2020). Make the directory read/write/executable only for you, read/write for your groups and read only for others.

mkdir HOO207SE2020
chmod 0764 HOO207SE2020
# Could have used chmod g-x / chmod o+r but this approach is much simpler [Insert citation http://www.filepermissions.com/directory-permission/0764]

b. Show evidence of this using the appropriate version of the **ls command**.

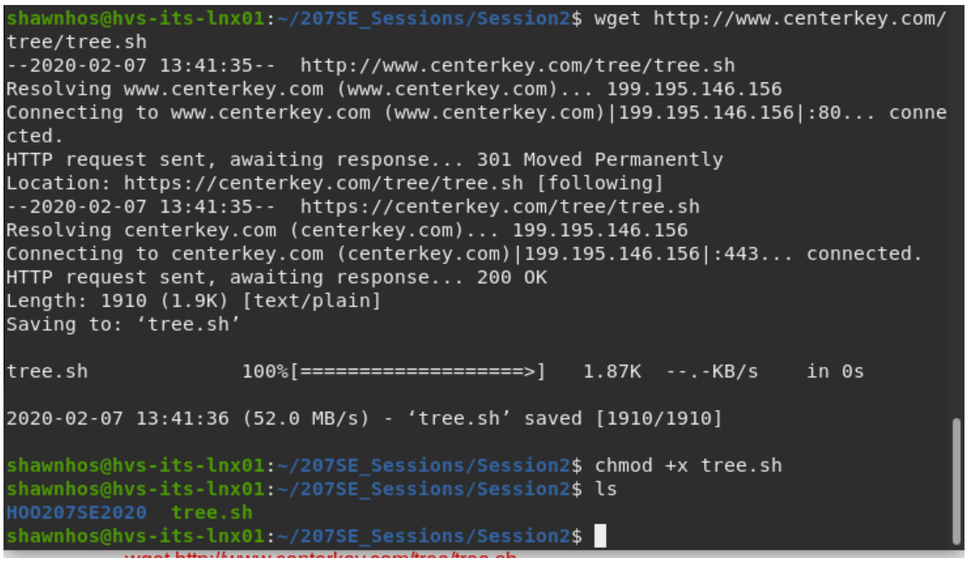


```
shawnhos@hvs-its-lnx01: ~/207SE_Sessions/Session2
root@kali: ~    shawnhos@hvs-its-lnx01: ~/207SE_Ses...
drwxr-xr-x 2 shawnhos domain users 4096 Feb  7 13:23 HOO207SE2020
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ chmod g-x
chmod: missing operand after ‘g-x’
Try 'chmod --help' for more information.
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ chmod g-x HOO207SE2020/
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ ls
HOO207SE2020
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ ls -l
total 4
drwxr--r-x 2 shawnhos domain users 4096 Feb  7 13:23 HOO207SE2020
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ chmod o-x HOO207SE2020/
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ ls -l
total 4
drwxr--r-- 2 shawnhos domain users 4096 Feb  7 13:23 HOO207SE2020
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ chmod g+rw HOO207SE2020/
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ chmod o+r HOO207SE2020/
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ ls -l
total 4
drwxrw-r-- 2 shawnhos domain users 4096 Feb  7 13:23 HOO207SE2020
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ chmod 0764 HOO207SE2020/
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ ls -l
total 4
drwxrw-r-- 2 shawnhos domain users 4096 Feb  7 13:23 HOO207SE2020
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ 
```

ls –l
# This shows the permissions for the dir as well

c. Download the script http://www.centerkey.com/tree/tree.sh to your home directory using wget and make the file executable.

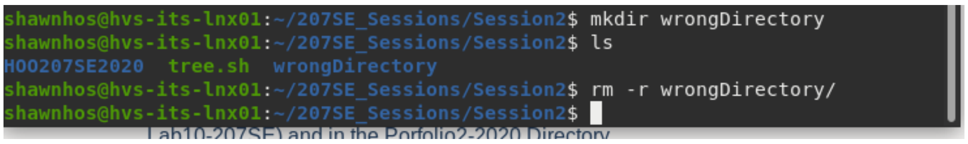
wget http://www.centerkey.com/tree/tree.sh chmod +x tree.sh



```
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ wget http://www.centerkey.com/
tree/tree.sh
--2020-02-07 13:41:35--  http://www.centerkey.com/tree/tree.sh
Resolving www.centerkey.com (www.centerkey.com)... 199.195.146.156
Connecting to www.centerkey.com (www.centerkey.com)|199.195.146.156|:80... conne
cted.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://centerkey.com/tree/tree.sh [following]
--2020-02-07 13:41:35--  https://centerkey.com/tree/tree.sh
Resolving centerkey.com (centerkey.com)... 199.195.146.156
Connecting to centerkey.com (centerkey.com)|199.195.146.156|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1910 (1.9K) [text/plain]
Saving to: ‘tree.sh’

tree.sh             100%[===================>]   1.87K  --.-KB/s    in 0s

2020-02-07 13:41:36 (52.0 MB/s) - ‘tree.sh’ saved [1910/1910]

shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ chmod +x tree.sh
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ ls
HOO207SE2020  tree.sh
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ 
```

d. Create a directory called **wrongDirectory**. You realise it is not what you wanted so delete it.

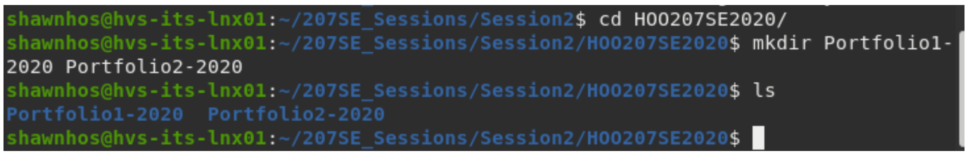
mkdir wrongDirectory
rm –r wrongDirectory
# -r parameter makes the command recursive which allows directories to be deleted.



```
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ mkdir wrongDirectory
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ ls
HOO207SE2020  tree.sh  wrongDirectory
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ rm -r wrongDirectory/
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ 
```

e. Create Portfolio1-2020 and Portfolio2-2020 directories in the directory you created in part a.

cd HOO207SE2020
mkdir Portfolio1-2020 Portfolio2-2020

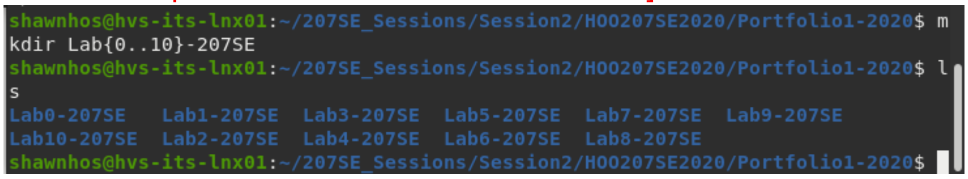


```
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2$ cd HOO207SE2020/
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HOO207SE2020$ mkdir Portfolio1-
2020 Portfolio2-2020
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HOO207SE2020$ ls
Portfolio1-2020  Portfolio2-2020
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HOO207SE2020$ 
```

f. Create numbered directories in the Portfolio1-2020 Directory (Lab0-207SE to Lab10-207SE) and in the Porfolio2-2020 Directory (Lab11-207SE to Lab20-207SE).

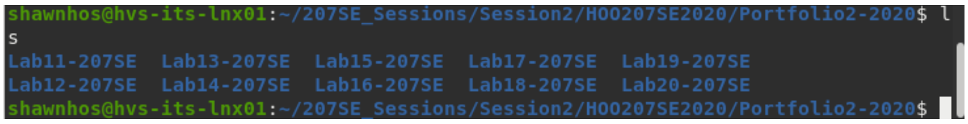
cd Portfolio1-2020
mkdir Lab{0..10}-207SE

# Rather than manually typing each file, this uses a single command to create all directories. [Citation https://askubuntu.com/questions/731721/is-there-a-way-to-create-multiple-directories-at-once-with-mkdir]



```
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HO0207SE2020/Portfolio1-2020$ m
kdir Lab{0..10}-207SE
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HO0207SE2020/Portfolio1-2020$ l
s
Lab0-207SE   Lab1-207SE  Lab3-207SE  Lab5-207SE  Lab7-207SE  Lab9-207SE
Lab10-207SE  Lab2-207SE  Lab4-207SE  Lab6-207SE  Lab8-207SE
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HO0207SE2020/Portfolio1-2020$
```

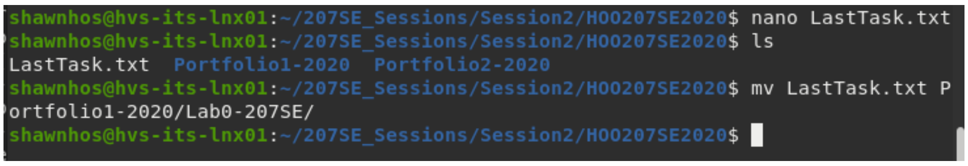
cd ..
cd Portfolio1-2020
mkdir Lab{11..20}-207SE



```
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HO0207SE2020/Portfolio2-2020$ l
s
Lab11-207SE  Lab13-207SE  Lab15-207SE  Lab17-207SE  Lab19-207SE
Lab12-207SE  Lab14-207SE  Lab16-207SE  Lab18-207SE  Lab20-207SE
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HO0207SE2020/Portfolio2-2020$
```

g. In <YourSecondName>207SE2020 directory create a text file called **LastTask.txt** and then using the appropriate Linux command copy this document into Directory Lab0-207SE.

nano LastTask.txt
# Add some text
ctrl + x, y # Exit the file and y to save
mv LastTask.txt Portfolio1-2020/Lab0-207SE

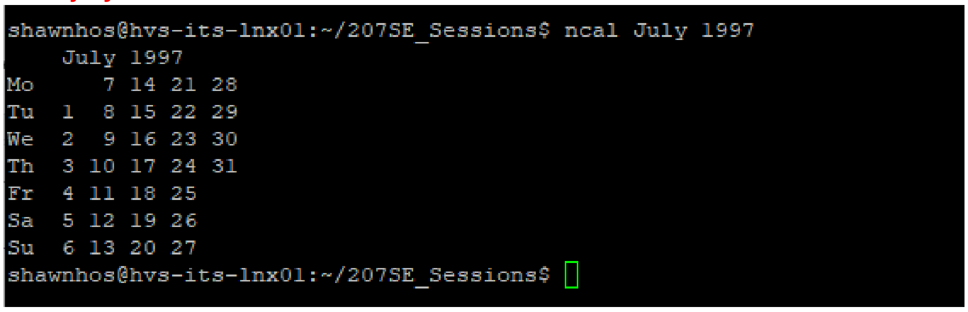


```
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HO0207SE2020$ nano LastTask.txt
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HO0207SE2020$ ls
LastTask.txt  Portfolio1-2020  Portfolio2-2020
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HO0207SE2020$ mv LastTask.txt P
ortfolio1-2020/Lab0-207SE/
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HO0207SE2020$
```

Linux Commands – Mixed

a. Using the date command show todays date and the time and date 5 years ago. Using the cal command show the month that you were born. Change this calendar to make Monday the first day of the week.

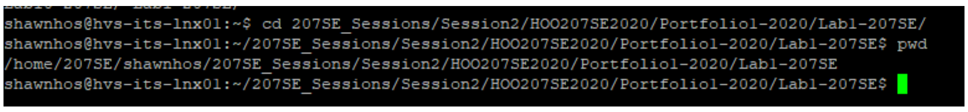
ncal july 1997



```
shawnhos@hvs-its-lnx01:~/207SE_Sessions$ ncal July 1997
    July 1997
Mo     7 14 21 28
Tu  1  8 15 22 29
We  2  9 16 23 30
Th  3 10 17 24 31
Fr  4 11 18 25
Sa  5 12 19 26
Su  6 13 20 27
shawnhos@hvs-its-lnx01:~/207SE_Sessions$
```

b. Move into the lab1-207SE directory and use the appropriate command to show the current directory.

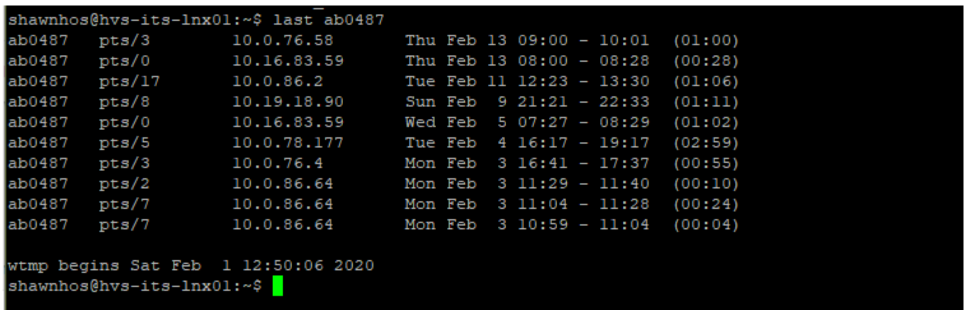
cd 207SE_Sessions/Session2/HOO207SE2020/Portfolio1-2020/Lab1-207SE/
pwd



```
shawnhos@hvs-its-lnx01:~$ cd 207SE_Sessions/Session2/HOO207SE2020/Portfolio1-2020/Lab1-207SE/
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HOO207SE2020/Portfolio1-2020/Lab1-207SE$ pwd
/home/207SE/shawnhos/207SE_Sessions/Session2/HOO207SE2020/Portfolio1-2020/Lab1-207SE
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session2/HOO207SE2020/Portfolio1-2020/Lab1-207SE$
```

c. Display the time of a user (ab0487) last login.

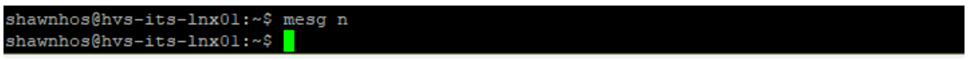
last ab0487



```
shawnhos@hvs-its-lnx01:~$ last ab0487
ab0487   pts/3        10.0.76.58       Thu Feb 13 09:00 - 10:01  (01:00)
ab0487   pts/0        10.16.83.59      Thu Feb 13 08:00 - 08:28  (00:28)
ab0487   pts/17       10.0.86.2        Tue Feb 11 12:23 - 13:30  (01:06)
ab0487   pts/8        10.19.18.90      Sun Feb  9 21:21 - 22:33  (01:11)
ab0487   pts/0        10.16.83.59      Wed Feb  5 07:27 - 08:29  (01:02)
ab0487   pts/5        10.0.78.177      Tue Feb  4 16:17 - 19:17  (02:59)
ab0487   pts/3        10.0.76.4        Mon Feb  3 16:41 - 17:37  (00:55)
ab0487   pts/2        10.0.86.64       Mon Feb  3 11:29 - 11:40  (00:10)
ab0487   pts/7        10.0.86.64       Mon Feb  3 11:04 - 11:28  (00:24)
ab0487   pts/7        10.0.86.64       Mon Feb  3 10:59 - 11:04  (00:04)

wtmp begins Sat Feb  1 12:50:06 2020
shawnhos@hvs-its-lnx01:~$
```

d. Find out how to prevent the effects of talk, write and wall from interrupting you. What command can you use?

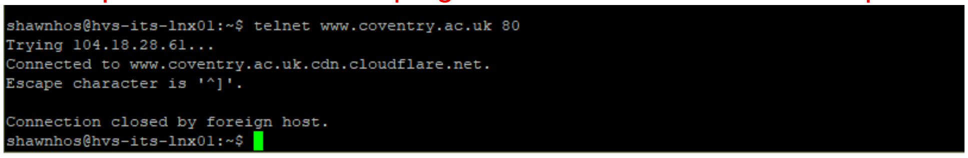
mesg n



```
shawnhos@hvs-its-lnx01:~$ mesg n
shawnhos@hvs-its-lnx01:~$
```

e. Show the command to verify that www.coventry.ac.uk exists and can accept requests.

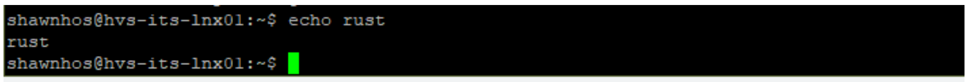
telnet www.coventry.ac.uk 80
# Also possible to use curl/ ping but this method is more fool proof and simple.



```
shawnhos@hvs-its-lnx01:~$ telnet www.coventry.ac.uk 80
Trying 104.18.28.61...
Connected to www.coventry.ac.uk.cdn.cloudflare.net.
Escape character is '^]'.

Connection closed by foreign host.
shawnhos@hvs-its-lnx01:~$
```

f. Display your name and favourite programming language on the screen using the echo command.

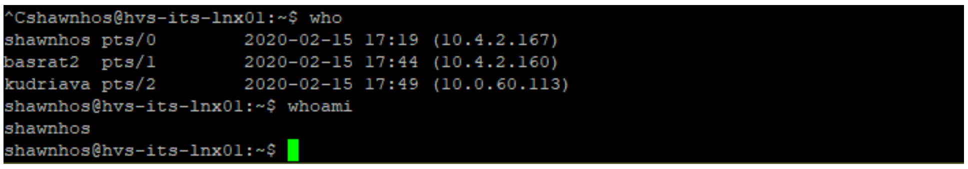
echo rust



```
shawnhos@hvs-its-lnx01:~$ echo rust
rust
shawnhos@hvs-its-lnx01:~$
```

g. Find out how you can display your username on the screen and at least two ways to display who is logged on.

who
whoami



```
^Cshawnhos@hvs-its-lnx01:~$ who
shawnhos pts/0        2020-02-15 17:19 (10.4.2.167)
basrat2  pts/1        2020-02-15 17:44 (10.4.2.160)
kudriava pts/2        2020-02-15 17:49 (10.0.60.113)
shawnhos@hvs-its-lnx01:~$ whoami
shawnhos
shawnhos@hvs-its-lnx01:~$
```

h. Use two ways to list the processes that are running.

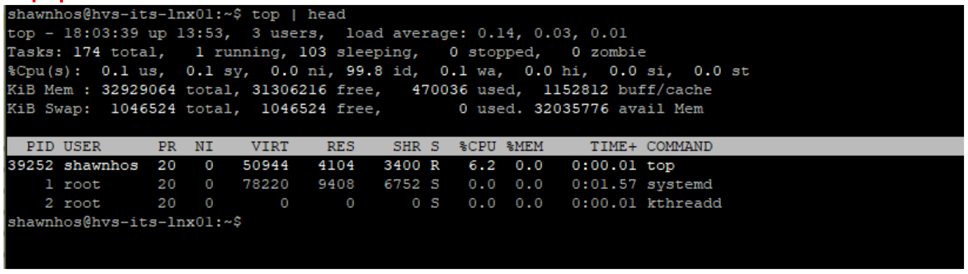
top | head
# Piped to head as top gives too many processes for the screenshot
ps

i. What are the differences between the Linux commands copy (cp), rename and move?

mv moves the file and deletes the original file, this applies to renaming as well. (moving and deleting) cp moves a file to a new location without deleting the original file, effectively copying it.

j. With a single command, how would you get systems information such as processes, memory, paging and CPU activity?

# Used previously
top | head

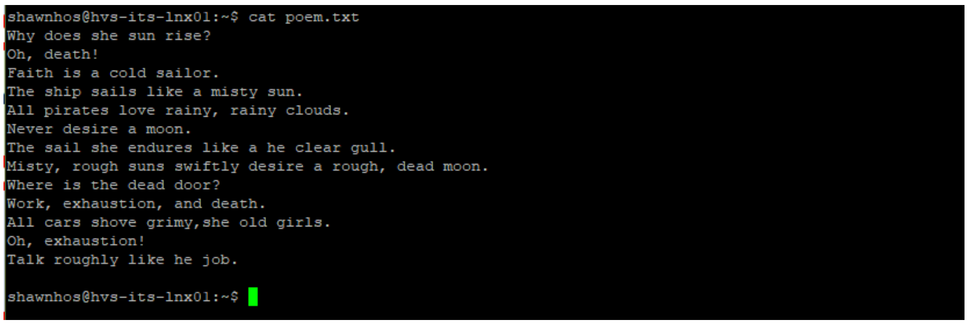


```
shawnhos@hvs-its-lnx01:~$ top | head
top - 18:03:39 up 13:53,  3 users,  load average: 0.14, 0.03, 0.01
Tasks: 174 total,   1 running, 103 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.1 us,  0.1 sy,  0.0 ni, 99.8 id,  0.1 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem : 32929064 total, 31306216 free,   470036 used,  1152812 buff/cache
KiB Swap:  1046524 total,  1046524 free,        0 used. 32035776 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
39252 shawnhos  20   0   50944   4104   3400 R   6.2  0.0   0:00.01 top
    1 root      20   0   78220   9408   6752 S   0.0  0.0   0:01.57 systemd
    2 root      20   0       0      0      0 S   0.0  0.0   0:00.01 kthreadd
shawnhos@hvs-its-lnx01:~$
```

Tasks – Document Manipulation

a. Use **cat** to show the contents of the file.

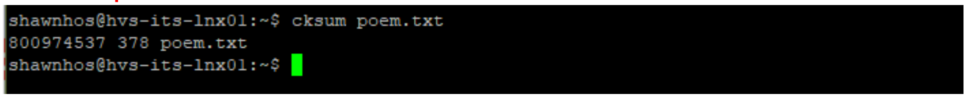
cat poem.txt



```
shawnhos@hvs-its-lnx01:~$ cat poem.txt
Why does she sun rise?
Oh, death!
Faith is a cold sailor.
The ship sails like a misty sun.
All pirates love rainy, rainy clouds.
Never desire a moon.
The sail she endures like a he clear gull.
Misty, rough suns swiftly desire a rough, dead moon.
Where is the dead door?
Work, exhaustion, and death.
All cars shove grimy,she old girls.
Oh, exhaustion!
Talk roughly like he job.

shawnhos@hvs-its-lnx01:~$
```

b. Use an appropriate command to display the CRC checksum and byte count of the file.

cksum poem.txt

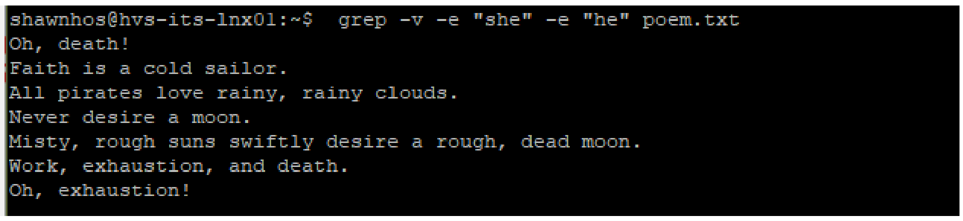


```
shawnhos@hvs-its-lnx01:~$ cksum poem.txt
800974537 378 poem.txt
shawnhos@hvs-its-lnx01:~$
```

c. Use **grep** to show only lines not containing the words "she" or "he". Lines contain both “she” and “he” should be shown.

grep -v -e "she" -e "he" poem.txt

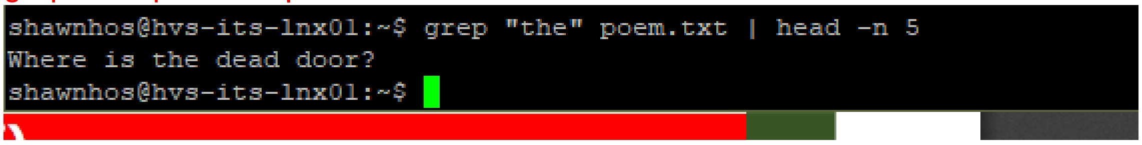
# Can’t figure out how to show a line when both terms exist



```
shawnhos@hvs-its-lnx01:~$  grep -v -e "she" -e "he" poem.txt
Oh, death!
Faith is a cold sailor.
All pirates love rainy, rainy clouds.
Never desire a moon.
Misty, rough suns swiftly desire a rough, dead moon.
Work, exhaustion, and death.
Oh, exhaustion!
```

d. Use **grep** to show the 5 lines above a line containing the text ‘the’.

grep "the" poem.txt | head -n 5

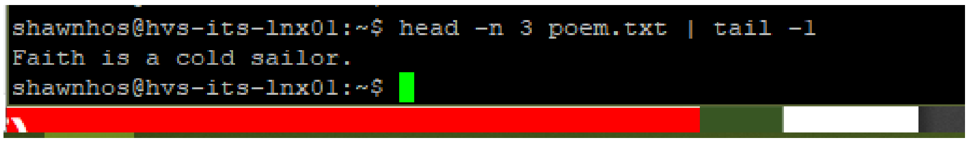


```
shawnhos@hvs-its-lnx01:~$ grep "the" poem.txt | head -n 5
Where is the dead door?
shawnhos@hvs-its-lnx01:~$
```

e. Using Linux commands you should count the lines containing “she” and “he” but not both and display the line numbers that “she” and “he” but not both appear on in the original document.

f. Find a command to list the top 3 lines of the **poem.txt** file and then the bottom line of these top 3.

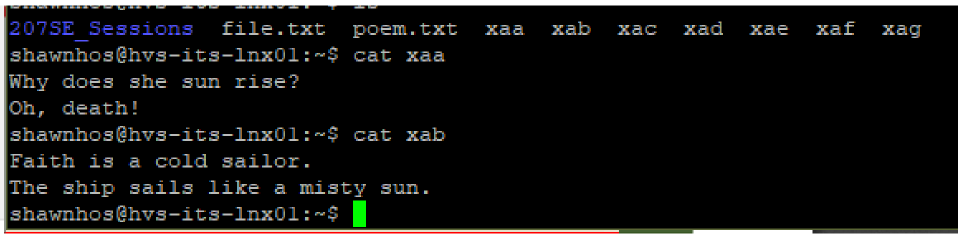
head -n 3 poem.txt | tail -1



```
shawnhos@hvs-its-lnx01:~$ head -n 3 poem.txt | tail -1
Faith is a cold sailor.
shawnhos@hvs-its-lnx01:~$
```

g. Find a command to split the **poem.txt** file into different files each containing 2 lines.

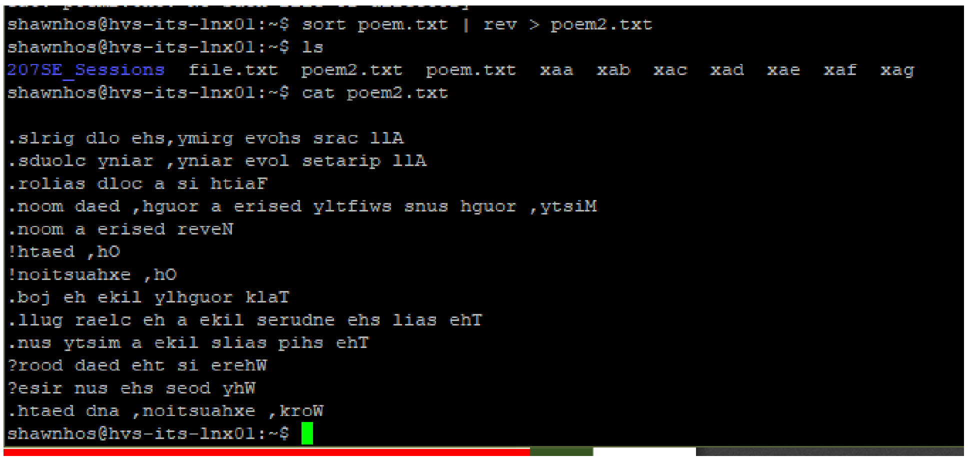
split -l 2 poem.txt



```
207SE_Sessions  file.txt  poem.txt  xaa  xab  xac  xad  xae  xaf  xag
shawnhos@hvs-its-lnx01:~$ cat xaa
Why does she sun rise?
Oh, death!
shawnhos@hvs-its-lnx01:~$ cat xab
Faith is a cold sailor.
The ship sails like a misty sun.
shawnhos@hvs-its-lnx01:~$
```

h. Use **sort** and **rev** to reverse the sorted contents of poem.txt and append the output to poem2.txt.

sort poem.txt | rev > poem2.txt

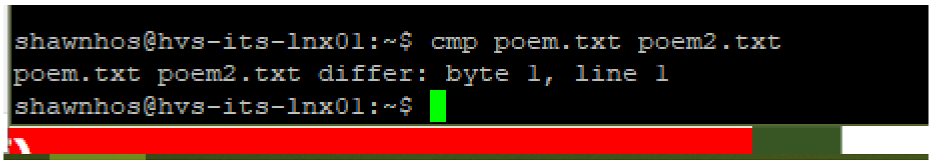


```
shawnhos@hvs-its-lnx01:~$ sort poem.txt | rev > poem2.txt
shawnhos@hvs-its-lnx01:~$ ls
207SE_Sessions  file.txt  poem2.txt  poem.txt  xaa  xab  xac  xad  xae  xaf  xag
shawnhos@hvs-its-lnx01:~$ cat poem2.txt

.slrig dlo ehs,ymirg evohs srac llA
.sduolc yniar ,yniar evol setarip llA
.rolias dloc a si htiaF
.noom daed ,hguor a erised yltfiws snus hguor ,ytsiM
.noom a erised reveN
!htaed ,hO
!noitsuahxe ,hO
.boj eh ekil ylhguor klaT
.llug raelc eh a ekil serudne ehs lias ehT
.nus ytsim a ekil slias pihs ehT
?rood daed eht si erehW
?esir nus ehs seod yhW
.htaed dna ,noitsuahxe ,kroW
shawnhos@hvs-its-lnx01:~$
```

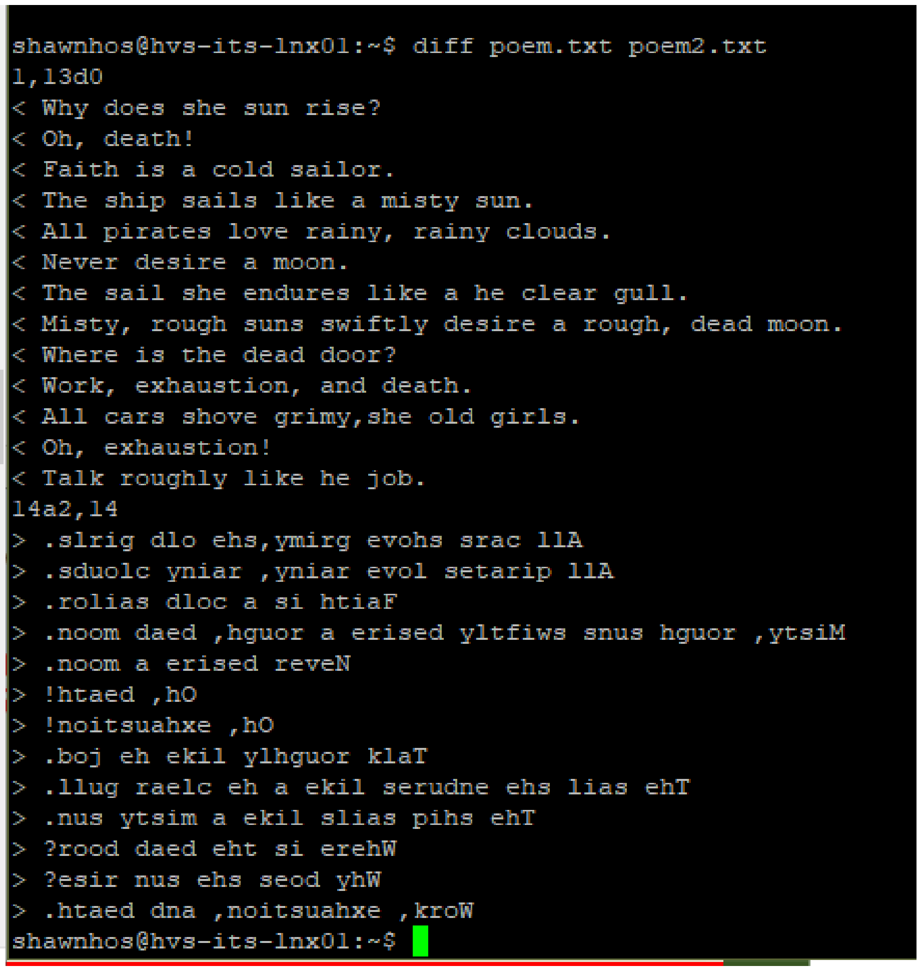
i. Use at least two appropriate Linux commands to compare these two files (poem.txt and poem2.txt) and see if they are the same.

cmp poem.txt poem2.txt

diff poem.txt poem2.txt



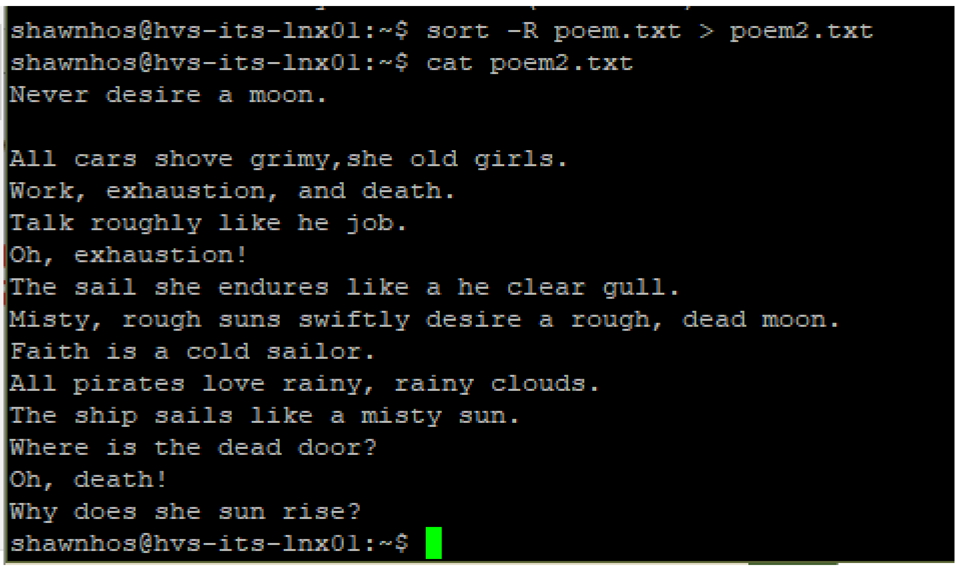
```
shawnhos@hvs-its-lnx01:~$ cmp poem.txt poem2.txt
poem.txt poem2.txt differ: byte 1, line 1
shawnhos@hvs-its-lnx01:~$
```



```
shawnhos@hvs-its-lnx01:~$ diff poem.txt poem2.txt
1,13d0
< Why does she sun rise?
< Oh, death!
< Faith is a cold sailor.
< The ship sails like a misty sun.
< All pirates love rainy, rainy clouds.
< Never desire a moon.
< The sail she endures like a he clear gull.
< Misty, rough suns swiftly desire a rough, dead moon.
< Where is the dead door?
< Work, exhaustion, and death.
< All cars shove grimy,she old girls.
< Oh, exhaustion!
< Talk roughly like he job.
14a2,14
> .slrig dlo ehs,ymirg evohs srac llA
> .sduolc yniar ,yniar evol setarip llA
> .rolias dloc a si htiaF
> .noom daed ,hguor a erised yltfiws snus hguor ,ytsiM
> .noom a erised reveN
> !htaed ,hO
> !noitsuahxe ,hO
> .boj eh ekil ylhguor klaT
> .llug raelc eh a ekil serudne ehs lias ehT
> .nus ytsim a ekil slias pihs ehT
> ?rood daed eht si erehW
> ?esir nus ehs seod yhW
> .htaed dna ,noitsuahxe ,kroW
shawnhos@hvs-its-lnx01:~$
```

j. Use **sort** to sort the content of poem.txt file in a random order and redirect the output to a new file called **poem2.txt**.

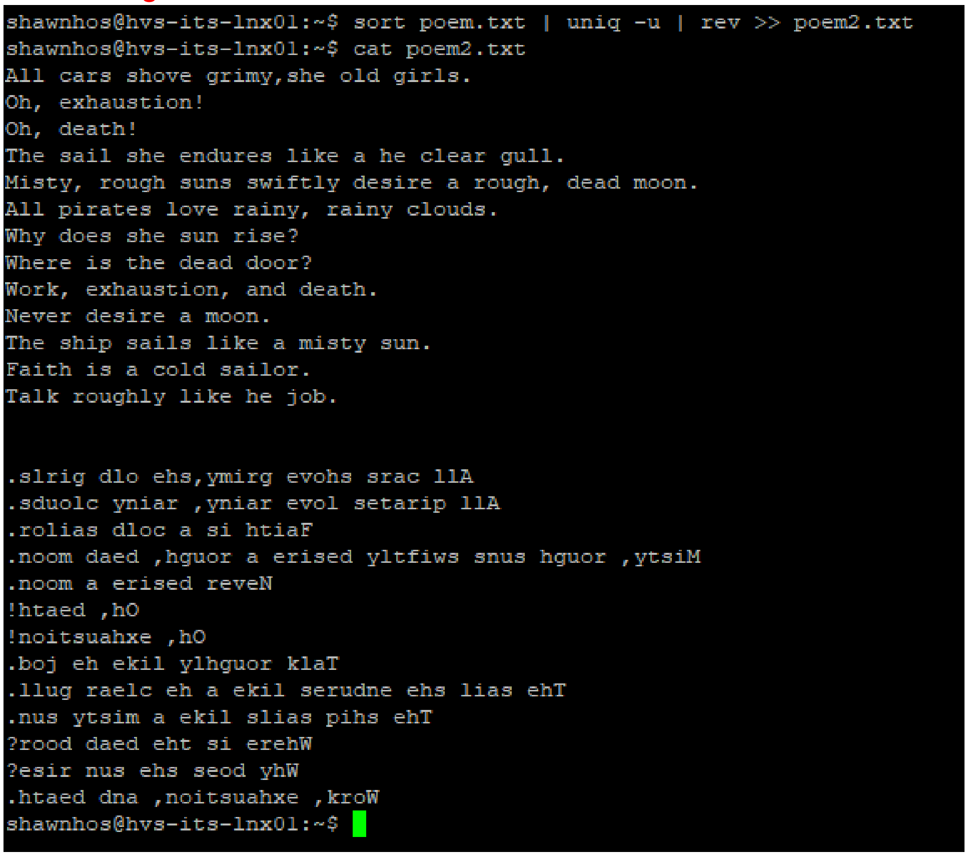
```
sort -R poem.txt > poem2.txt
```



```
shawnhos@hvs-its-lnx01:~$ sort -R poem.txt > poem2.txt
shawnhos@hvs-its-lnx01:~$ cat poem2.txt
Never desire a moon.

All cars shove grimy,she old girls.
Work, exhaustion, and death.
Talk roughly like he job.
Oh, exhaustion!
The sail she endures like a he clear gull.
Misty, rough suns swiftly desire a rough, dead moon.
Faith is a cold sailor.
All pirates love rainy, rainy clouds.
The ship sails like a misty sun.
Where is the dead door?
Oh, death!
Why does she sun rise?
shawnhos@hvs-its-lnx01:~$
```

k. Sort the **poem.txt** file, remove the duplicates and reverse the sorted contents and append the output to **poem2.txt**.

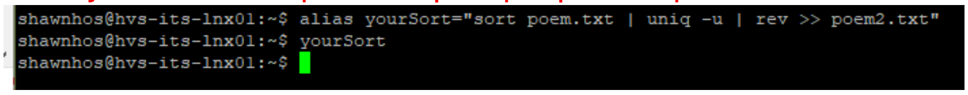
sort poem.txt | uniq -u | rev >> poem2.txt
# Being careful to use >> instead of > to append to end of file rather than overwriting.



```
shawnhos@hvs-its-lnx01:~$ sort poem.txt | uniq -u | rev >> poem2.txt
shawnhos@hvs-its-lnx01:~$ cat poem2.txt
All cars shove grimy,she old girls.
Oh, exhaustion!
Oh, death!
The sail she endures like a he clear gull.
Misty, rough suns swiftly desire a rough, dead moon.
All pirates love rainy, rainy clouds.
Why does she sun rise?
Where is the dead door?
Work, exhaustion, and death.
Never desire a moon.
The ship sails like a misty sun.
Faith is a cold sailor.
Talk roughly like he job.


.slrig dlo ehs,ymirg evohs srac llA
.sduolc yniar ,yniar evol setarip llA
.rolias dloc a si htiaF
.noom daed ,hguor a erised yltfiws snus hguor ,ytsiM
.noom a erised reveN
!htaed ,hO
!noitsuahxe ,hO
.boj eh ekil ylhguor klaT
.llug raelc eh a ekil serudne ehs lias ehT
.nus ytsim a ekil slias pihs ehT
?rood daed eht si erehW
?esir nus ehs seod yhW
.htaed dna ,noitsuahxe ,kroW
shawnhos@hvs-its-lnx01:~$
```

l. Create an **alias** so rather than having to type the full command for k) you can type **yourSort**.

alias yourSort="sort poem.txt | uniq -u | rev >> poem2.txt"



```
shawnhos@hvs-its-lnx01:~$ alias yourSort="sort poem.txt | uniq -u | rev >> poem2.txt"
shawnhos@hvs-its-lnx01:~$ yourSort
shawnhos@hvs-its-lnx01:~$
```

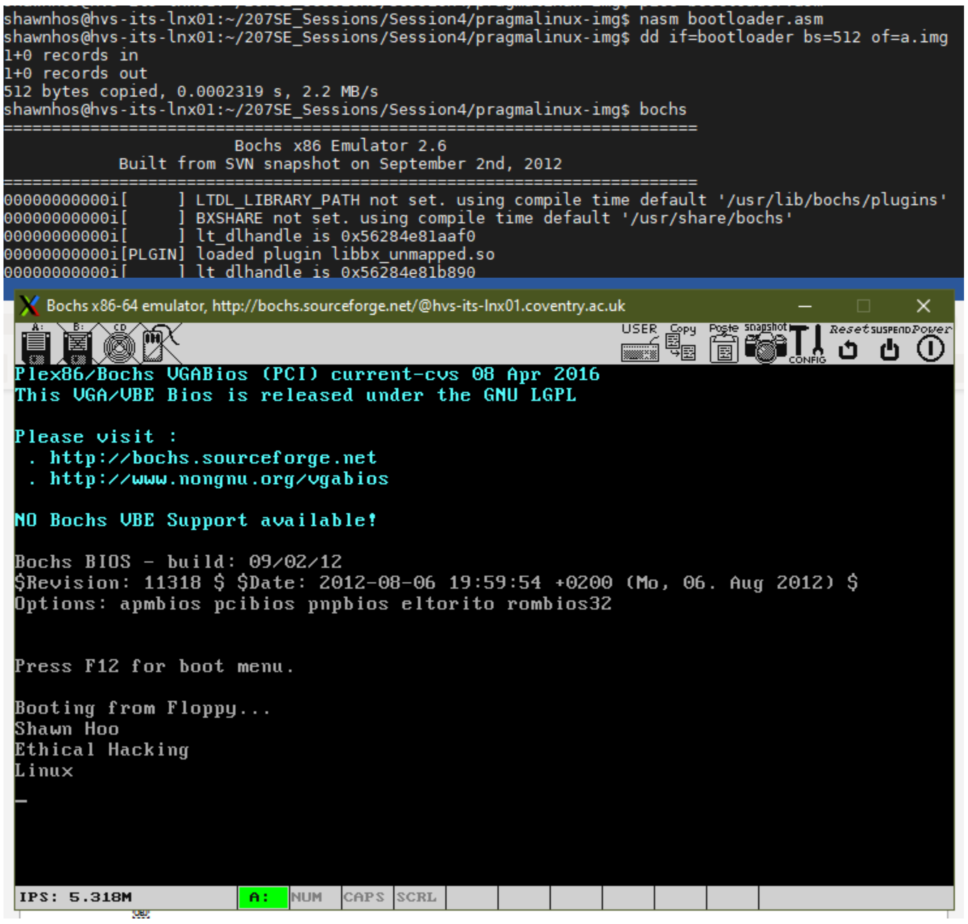
Lab Activity 4 Bootloader

a. Brief description of the Lab activity and what you did

The Bootlloader activity takes a look at assembly language. This activity is to create a bootloader which evolves writing some additional assembly code to be able to first print out a single line of characters and then printing out multiple lines of characters which in this case is the student information.

For this task, I took a look at the provided helloworld.asm bootloader program and copied the pre existing values that contact the words ‘hello world’, I then proceeded to make duplicates of these variables assigning them to different names eg. Name, course etc. I then used the pre existing functions for writeString and individually moving in each variable containing the different information, displaying it on the screen and moving the next variable. Then for a more difficult challenge, I printed a diamond + student information using the same principles.

b. Boot pragma Linux with bochs



```
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session4/pragmalinux-img$ nasm bootloader.asm
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session4/pragmalinux-img$ dd if=bootloader bs=512 of=a.img
1+0 records in
1+0 records out
512 bytes copied, 0.0002319 s, 2.2 MB/s
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session4/pragmalinux-img$ bochs
========================================================================
                       Bochs x86 Emulator 2.6
             Built from SVN snapshot on September 2nd, 2012
========================================================================
00000000000i[     ] LTDL_LIBRARY_PATH not set. using compile time default '/usr/lib/bochs/plugins'
00000000000i[     ] BXSHARE not set. using compile time default '/usr/share/bochs'
00000000000i[     ] lt_dlhandle is 0x56284e81aaf0
00000000000i[PLGIN] loaded plugin libbx_unmapped.so
00000000000i[     ] lt_dlhandle is 0x56284e81b890
```

Bochs x86-64 emulator, http://bochs.sourceforge.net/@hvs-its-lnx01.coventry.ac.uk
USER
Copy
Paste
snapshot
CONFIG
Reset
SUSPEND
Power

```
Plex86/Bochs VGABios (PCI) current-cvs 08 Apr 2016
This VGA/VBE Bios is released under the GNU LGPL

Please visit :
 . http://bochs.sourceforge.net
 . http://www.nongnu.org/vgabios

NO Bochs VBE Support available!

Bochs BIOS - build: 09/02/12
$Revision: 11318 $ $Date: 2012-08-06 19:59:54 +0200 (Mo, 06. Aug 2012) $
Options: apmbios pcibios pnpbios eltorito rombios32


Press F12 for boot menu.

Booting from Floppy...
Shawn Hoo
Ethical Hacking
Linux

_
```

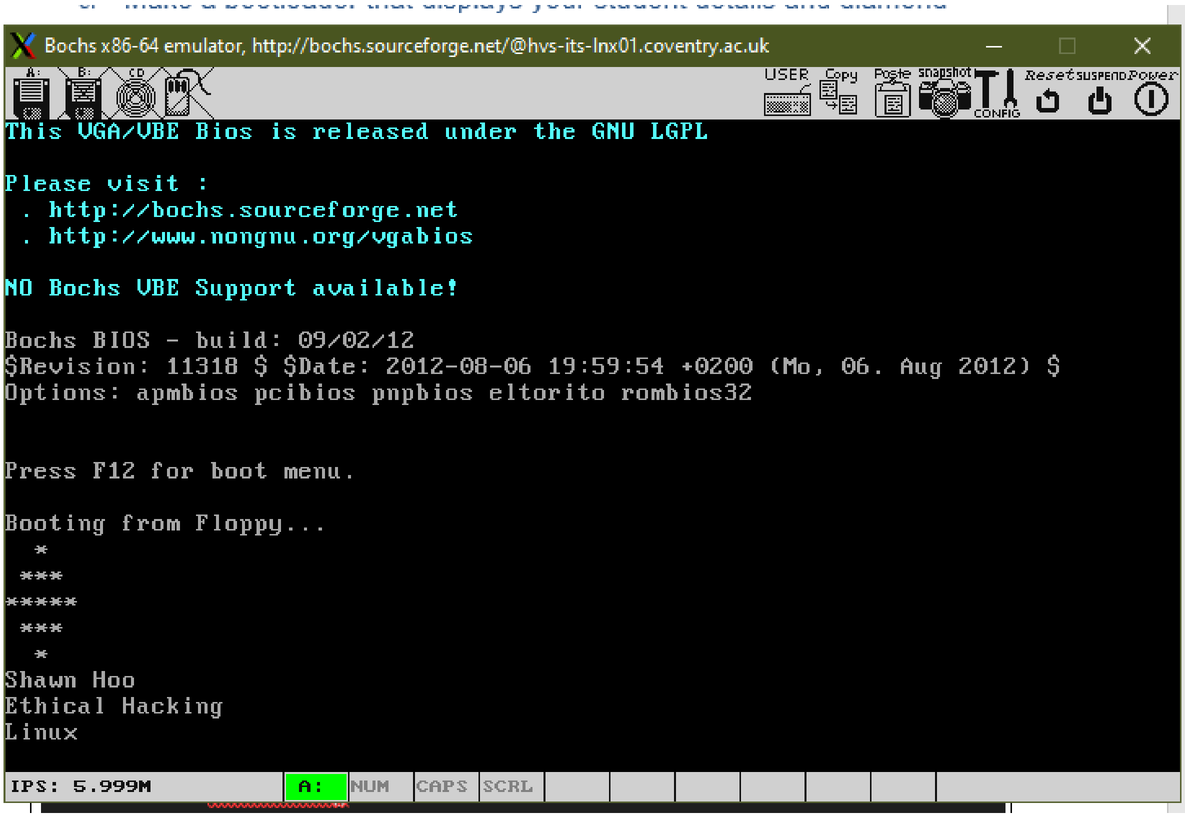
```
[BITS 16]
[ORG 0x7C00]
        top:
                ;; Put 0 into ds (data segment)
                ;; Can't do it directly
                mov ax,0x0000
                mov ds,ax
                ;; si is the location relative to the data segment of the
                ;; string/char to display
                mov si, Name
                call writeString ; See below
                mov si, Course
                call writeString ; See below
                mov si, fav_OS
                call writeString ; See below
                jmp $
        writeString:
                mov ah,0x0E ; Display a chacter (as before)
                mov bh,0x00
                mov bl,0x07
        nextchar:
                Lodsb ; Loads [SI] into AL and increases SI by one
                ;; Effectively "pumps" the string through AL
                cmp al,0 ; End of the string?
                jz done
                int 0x10 ; BIOS interrupt
                jmp nextchar
        done:
                ret
                Name db 'Shawn Hoo',13,10,0
                Course db 'Ethical Hacking',13,10,0
                fav_OS db 'Linux',13,10,0 ; Null-terminated
                times 510-($-$$) db 0
                dw 0xAA55
```

c. Make a bootloader that displays your student details and diamond

Commented bootloader code to display your student details and diamond

```
[BITS 16]
[ORG 0x7C00]
top:
        ;; Put 0 into ds (data segment)
        ;; Can't do it directly
        mov ax,0x0000
        mov ds,ax
        ;; si is the location relative to the data segment of the
        ;; string/char to display
        mov si, top_diamond
        call writeString
        mov si, middle_diamond
        call writeString
        mov si, bottom_diamond
        call writeString
        mov si , middle_diamond ; Calls the middle again
        call writeString
        mov si, top_diamond ; Finally calls the top again to finish the tip
        call writeString
        mov si, Name
        call writeString
        mov si, Course
        call writeString
        mov si, Fav_OS
        call writeString
        jmp $
writeString:
        mov ah,0x0E ; Display a chacter (as before)
        mov bh,0x00
        mov bl,0x07
nextchar:
        Lodsb ; Loads [SI] into AL and increases SI by one
        ;; Effectively "pumps" the string through AL
        cmp al,0 ; End of the string?
        jz done
        int 0x10 ; BIOS interrupt
        jmp nextchar
done:
        ret
        top_diamond db '  *',13,10,0 ; Definition for top of diamond
        middle_diamond db ' ***',13,10,0 ; Var for middle of diamond
        bottom_diamond db '*****',13,10,0; Var for “bottom” of diamond
        Name db 'Shawn Hoo',13,10,0
        Course db 'Ethical Hacking',13,10,0
        Fav_OS db 'Linux',13,10,0; Null-terminated
        times 510-($-$$) db 0
        dw 0xAA55
```

Output from bochs showing student details and diamond

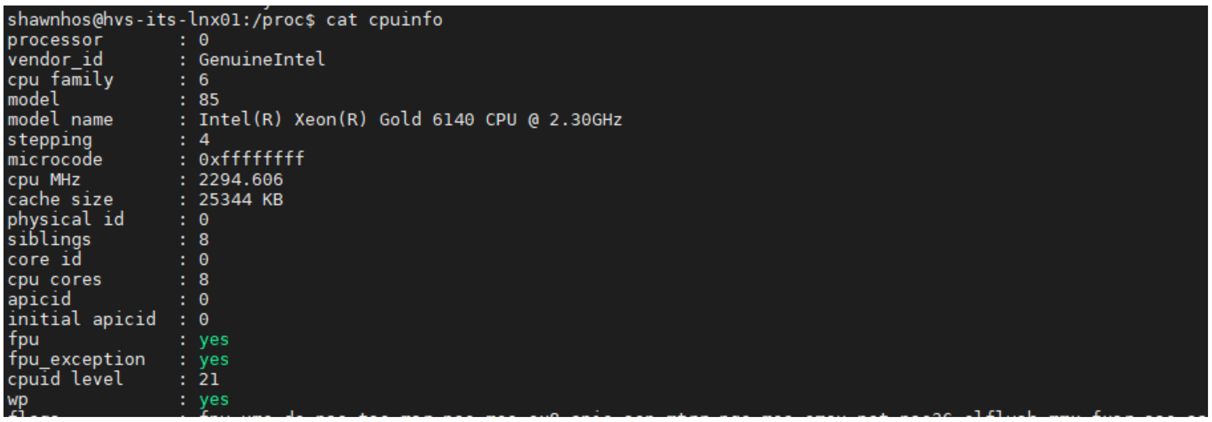

Bochs x86-64 emulator, http://bochs.sourceforge.net/@hvs-its-lnx01.coventry.ac.uk
USER
Copy
Paste
snapshot
CONFIG
Reset
SUSPEND
Power
IPS: 5.999M
A:
NUM
CAPS
SCRL

```
This VGA/VBE Bios is released under the GNU LGPL

Please visit :
 . http://bochs.sourceforge.net
 . http://www.nongnu.org/vgabios

NO Bochs VBE Support available!

Bochs BIOS - build: 09/02/12
$Revision: 11318 $ $Date: 2012-08-06 19:59:54 +0200 (Mo, 06. Aug 2012) $
Options: apmbios pcibios pnpbios eltorito rombios32


Press F12 for boot menu.

Booting from Floppy...
  *
 ***
*****
 ***
  *
Shawn Hoo
Ethical Hacking
Linux
```

Lab Activity 5 Exploring what is going on outside the processor

a. List the information found in the /proc directory about the computer CPUs.

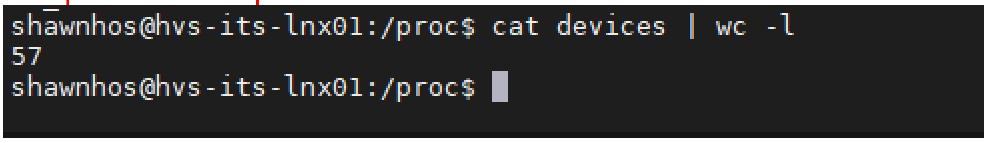
cat /proc/cpuinfo



```
shawnhos@hvs-its-lnx01:/proc$ cat cpuinfo
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 6
model           : 85
model name      : Intel(R) Xeon(R) Gold 6140 CPU @ 2.30GHz
stepping        : 4
microcode       : 0xffffffff
cpu MHz         : 2294.606
cache size      : 25344 KB
physical id     : 0
siblings        : 8
core id         : 0
cpu cores       : 8
apicid          : 0
initial apicid  : 0
fpu             : yes
fpu_exception   : yes
cpuid level     : 21
wp              : yes
```

b. Provide a list of the device drivers configured into the currently running kernel. Count the number of different device drivers that are included in the kernel.

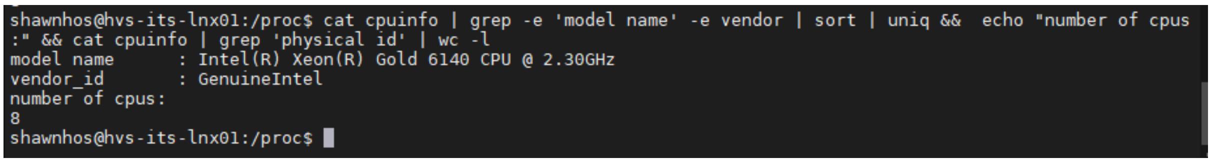
cat /proc/devices | wc -l



```
shawnhos@hvs-its-lnx01:/proc$ cat devices | wc -l
57
shawnhos@hvs-its-lnx01:/proc$
```

c. Show the number of CPUs, the producer of the CPUs and the CPU model.

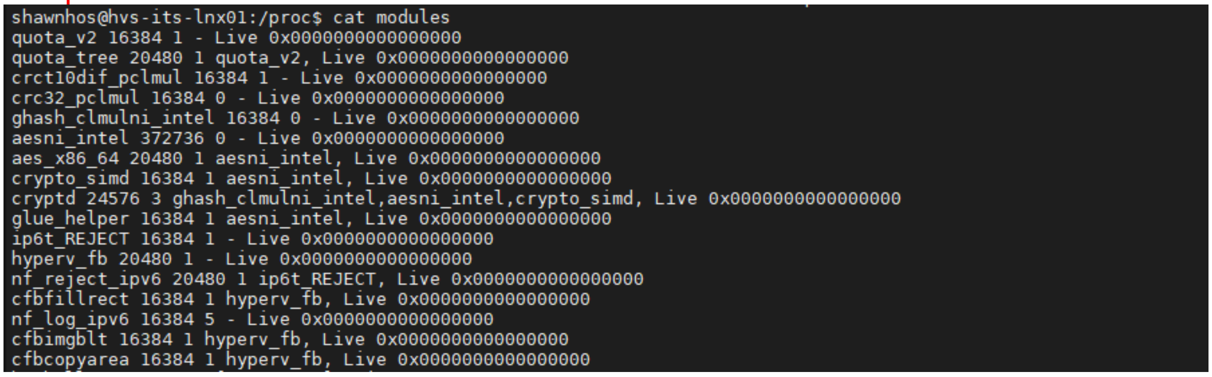
cat /proc/cpuinfo | grep -e 'model name' -e ‘vendor’ | sort | uniq && echo "number of cpus:" && cat cpuinfo | grep 'physical id' | wc -l



```
shawnhos@hvs-its-lnx01:/proc$ cat cpuinfo | grep -e 'model name' -e vendor | sort | uniq &&  echo "number of cpus
:" && cat cpuinfo | grep 'physical id' | wc -l
model name      : Intel(R) Xeon(R) Gold 6140 CPU @ 2.30GHz
vendor_id       : GenuineIntel
number of cpus:
8
shawnhos@hvs-its-lnx01:/proc$
```

d. Display a list of all modules that have been loaded by the system.

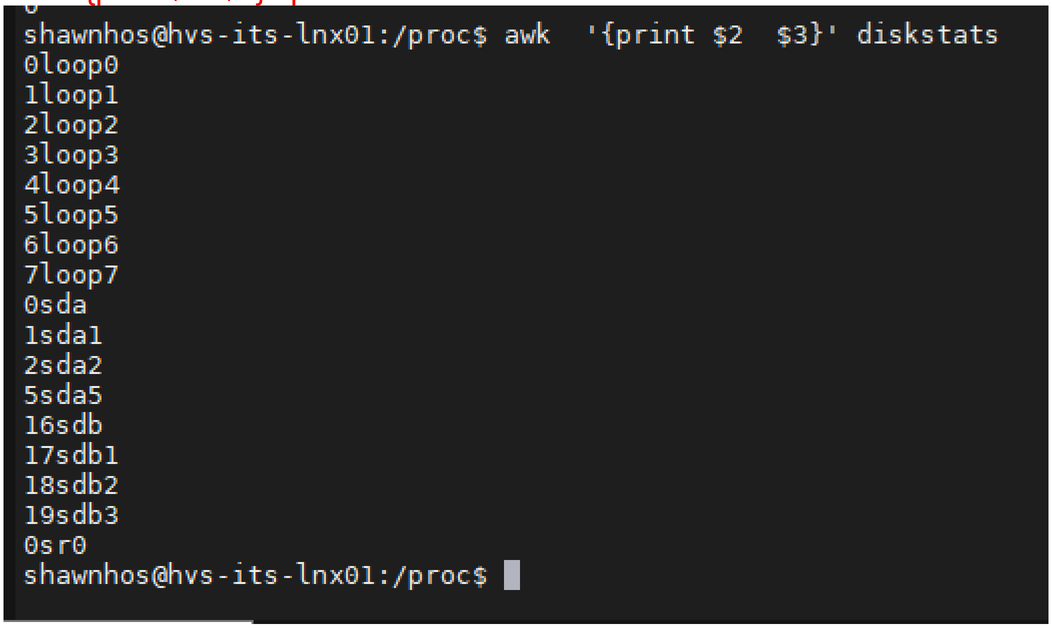
cat /proc/modules



```
shawnhos@hvs-its-lnx01:/proc$ cat modules
quota_v2 16384 1 - Live 0x0000000000000000
quota_tree 20480 1 quota_v2, Live 0x0000000000000000
crct10dif_pclmul 16384 1 - Live 0x0000000000000000
crc32_pclmul 16384 0 - Live 0x0000000000000000
ghash_clmulni_intel 16384 0 - Live 0x0000000000000000
aesni_intel 372736 0 - Live 0x0000000000000000
aes_x86_64 20480 1 aesni_intel, Live 0x0000000000000000
crypto_simd 16384 1 aesni_intel, Live 0x0000000000000000
cryptd 24576 3 ghash_clmulni_intel,aesni_intel,crypto_simd, Live 0x0000000000000000
glue_helper 16384 1 aesni_intel, Live 0x0000000000000000
ip6t_REJECT 16384 1 - Live 0x0000000000000000
hyperv_fb 20480 1 - Live 0x0000000000000000
nf_reject_ipv6 20480 1 ip6t_REJECT, Live 0x0000000000000000
cfbfillrect 16384 1 hyperv_fb, Live 0x0000000000000000
nf_log_ipv6 16384 5 - Live 0x0000000000000000
cfbimgblt 16384 1 hyperv_fb, Live 0x0000000000000000
cfbcopyarea 16384 1 hyperv_fb, Live 0x0000000000000000
```

e. Using the /proc/diskstats show the names of the output devices and the number of megabytes read per second during the sampled interval.

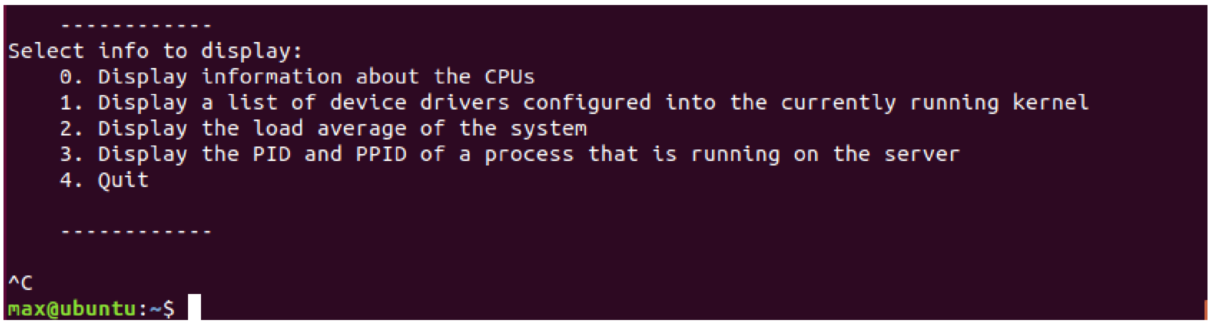
awk '{print $2 $3}' /proc/diskstats

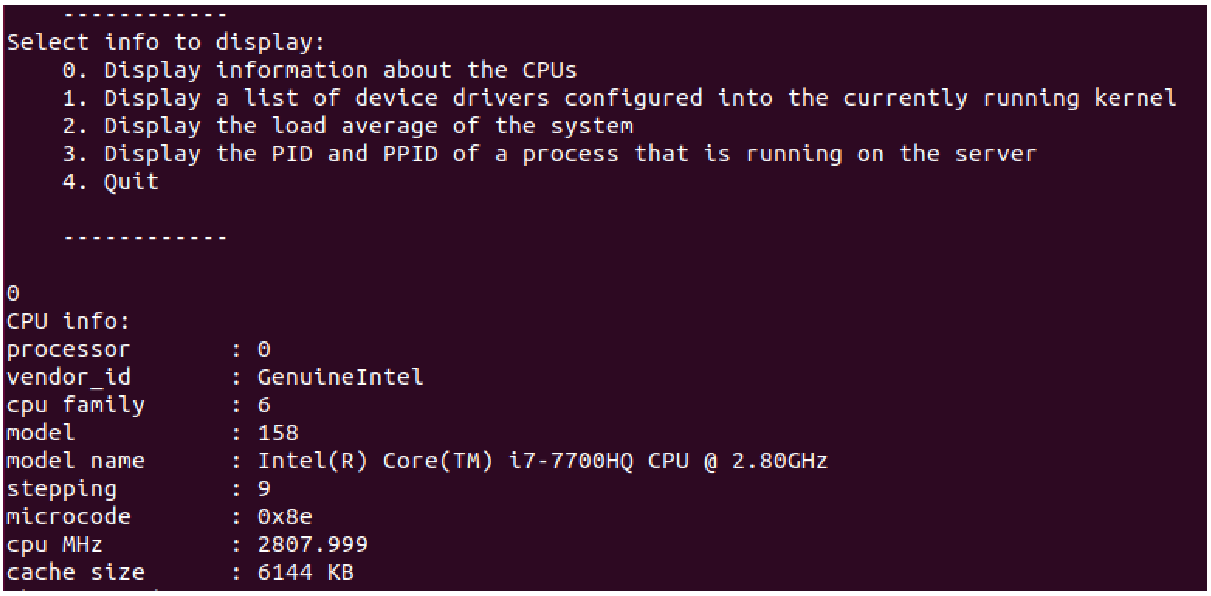


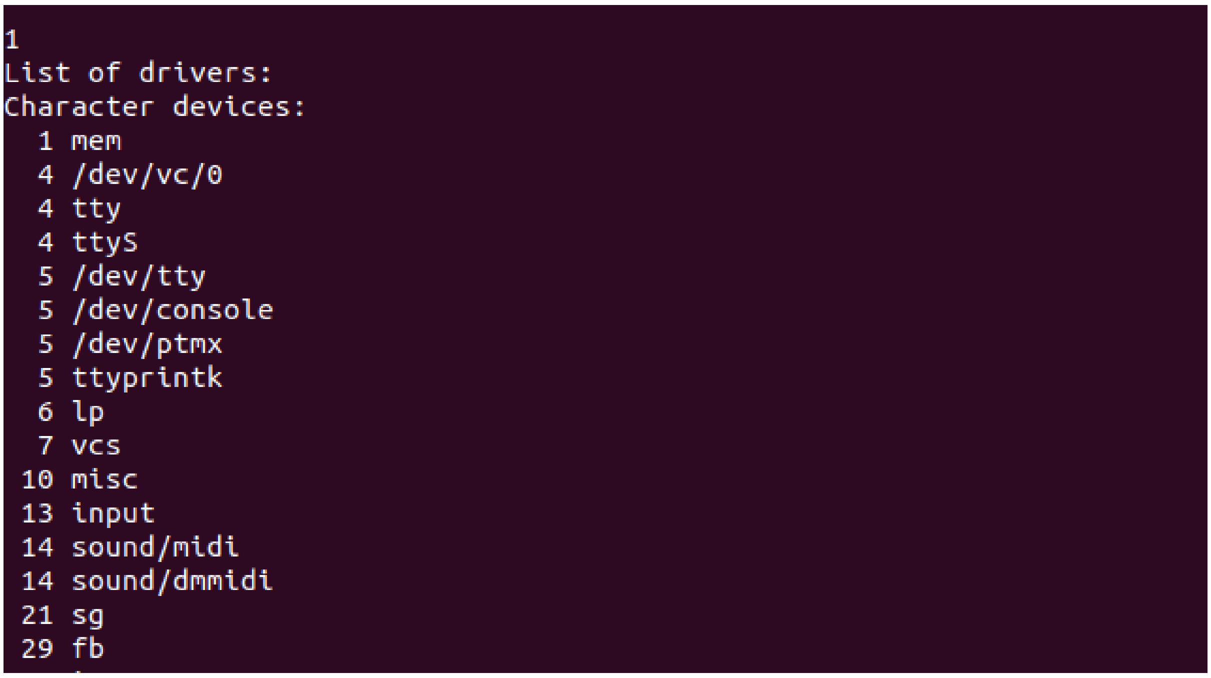
```
shawnhos@hvs-its-lnx01:/proc$ awk  '{print $2  $3}' diskstats
0loop0
1loop1
2loop2
3loop3
4loop4
5loop5
6loop6
7loop7
0sda
1sda1
2sda2
5sda5
16sdb
17sdb1
18sdb2
19sdb3
0sr0
shawnhos@hvs-its-lnx01:/proc$
```

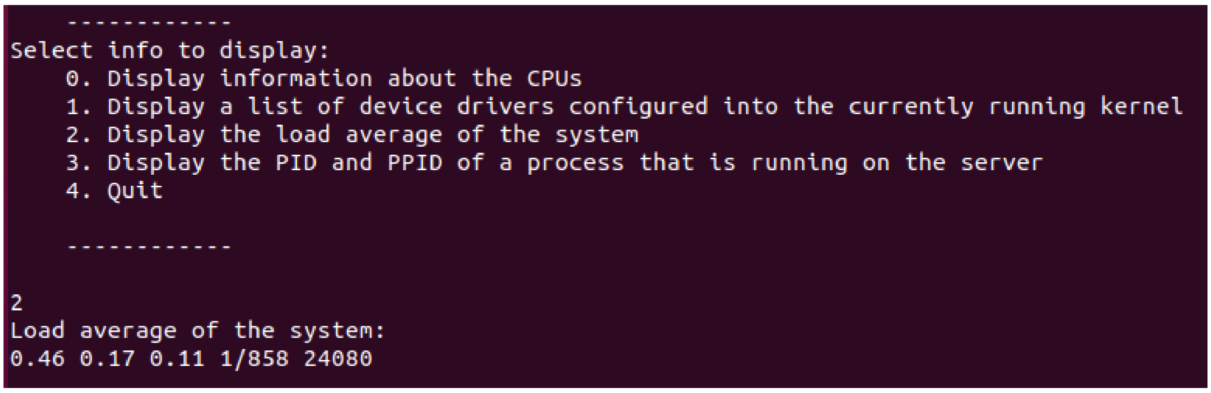
f. Menu based shell script.

```
#!/bin/bash
input=""
while [ "$input" != "4" ]; do
  echo "Select info to display:
    0. Display information about the CPUs
    1. Display a list of device drivers configured into the currently runnin
g kernel
    2. Display the load average of the system
    3. Display the PID and PPID of a process that is running on the server
    4. Quit
    ------------
    "
  read input
  input0=$(cat /proc/cpuinfo)
  input1=$(cat /proc/devices )
  input2=$(cat /proc/loadavg)
  input3=$(ps -ef | head)
  input4=$(exit)
  if [ "$input" = "0" ]; then
    echo "CPU info:
$input0

    ------------"
  fi
  if [ "$input" = "1" ]; then
    echo "List of drivers:
$input1

    ------------"
  fi
  if [ "$input" = "2" ]; then
    echo "Load average of the system:
$input2

    ------------"
  fi
  if [ "$input" = "3" ]; then
    echo "PID & PPID for processes:
$input3

    ------------"
  fi
  if [ "$input" = "4" ]; then
    echo "Goodbye $input4

    ------------"
  fi
done
```

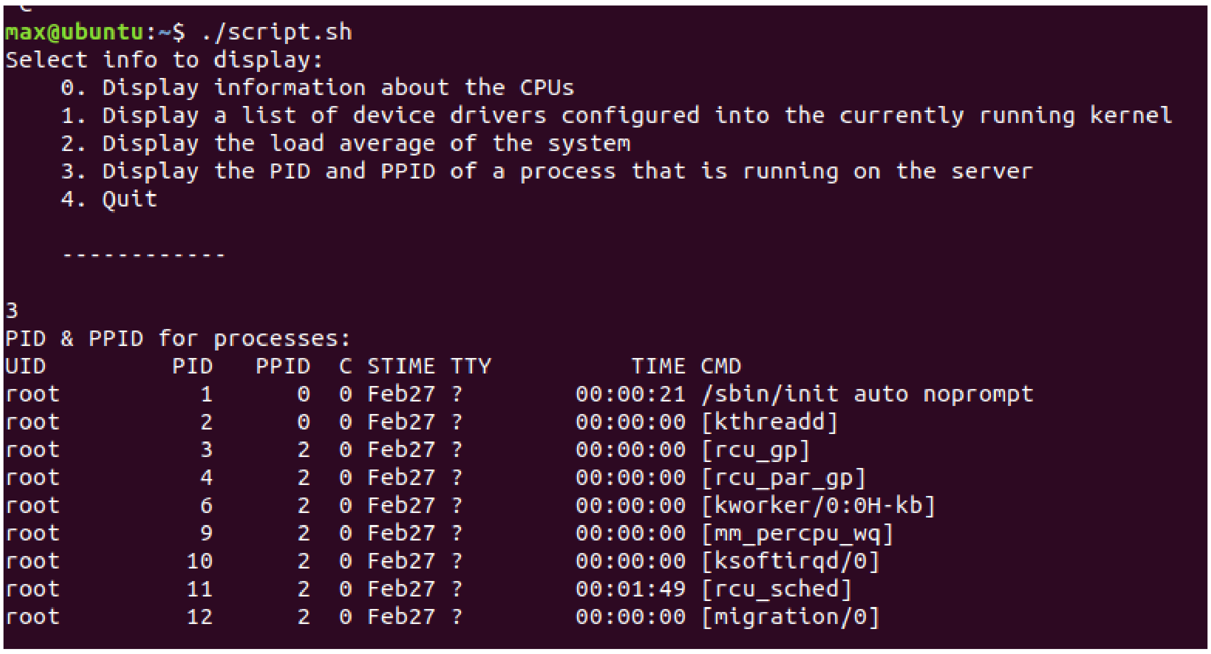

```
    ------------
Select info to display:
    0. Display information about the CPUs
    1. Display a list of device drivers configured into the currently running kernel
    2. Display the load average of the system
    3. Display the PID and PPID of a process that is running on the server
    4. Quit

    ------------

^C
max@ubuntu:~$ 
```


```
    ------------
Select info to display:
    0. Display information about the CPUs
    1. Display a list of device drivers configured into the currently running kernel
    2. Display the load average of the system
    3. Display the PID and PPID of a process that is running on the server
    4. Quit

    ------------

0
CPU info:
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 6
model           : 158
model name      : Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz
stepping        : 9
microcode       : 0x8e
cpu MHz         : 2807.999
cache size      : 6144 KB
```

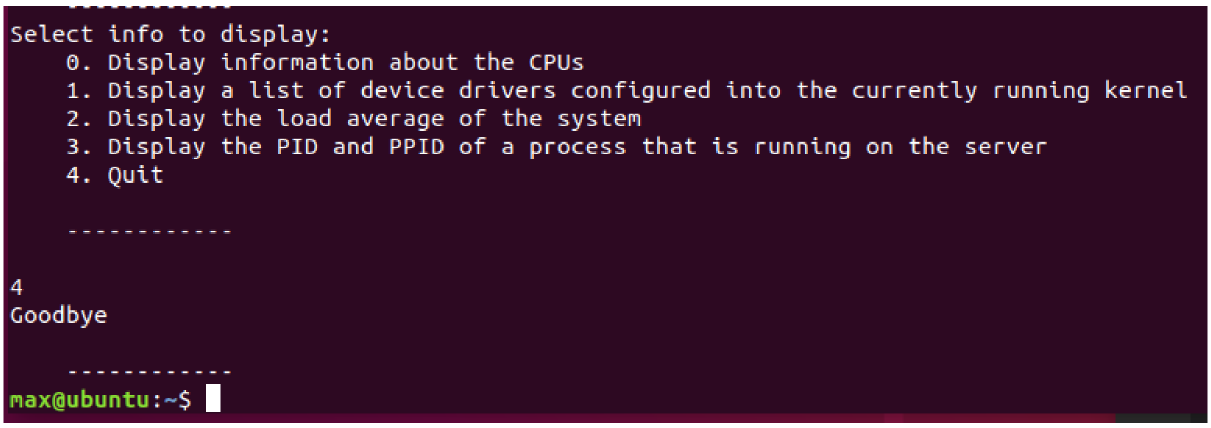

```
1
List of drivers:
Character devices:
  1 mem
  4 /dev/vc/0
  4 tty
  4 ttyS
  5 /dev/tty
  5 /dev/console
  5 /dev/ptmx
  5 ttyprintk
  6 lp
  7 vcs
 10 misc
 13 input
 14 sound/midi
 14 sound/dmmidi
 21 sg
 29 fb
```



```
    ------------
Select info to display:
    0. Display information about the CPUs
    1. Display a list of device drivers configured into the currently running kernel
    2. Display the load average of the system
    3. Display the PID and PPID of a process that is running on the server
    4. Quit

    ------------

2
Load average of the system:
0.46 0.17 0.11 1/858 24080
```



```
max@ubuntu:~$ ./script.sh
Select info to display:
    0. Display information about the CPUs
    1. Display a list of device drivers configured into the currently running kernel
    2. Display the load average of the system
    3. Display the PID and PPID of a process that is running on the server
    4. Quit

    ------------

3
PID & PPID for processes:
UID          PID    PPID  C STIME TTY          TIME CMD
root           1       0  0 Feb27 ?        00:00:21 /sbin/init auto noprompt
root           2       0  0 Feb27 ?        00:00:00 [kthreadd]
root           3       2  0 Feb27 ?        00:00:00 [rcu_gp]
root           4       2  0 Feb27 ?        00:00:00 [rcu_par_gp]
root           6       2  0 Feb27 ?        00:00:00 [kworker/0:0H-kb]
root           9       2  0 Feb27 ?        00:00:00 [mm_percpu_wq]
root          10       2  0 Feb27 ?        00:00:00 [ksoftirqd/0]
root          11       2  0 Feb27 ?        00:01:49 [rcu_sched]
root          12       2  0 Feb27 ?        00:00:00 [migration/0]
```



```
Select info to display:
    0. Display information about the CPUs
    1. Display a list of device drivers configured into the currently running kernel
    2. Display the load average of the system
    3. Display the PID and PPID of a process that is running on the server
    4. Quit

    ------------

4
Goodbye

    ------------
max@ubuntu:~$ 
```

Lab Activity 6 Memory Management

a. Memory Allocation

[First fit, Best fit and Worst fit allocation]

First Fit Allocation

M1 – 300	M2 – 500	M3 – 250	M4 – 280	M5 - 370
P1 – 300	P2 – 350	P3 – 250	P5 – 170	

P4 – Unallocated as process size is too big

Best Fit Allocation

M1 – 300	M2 – 500	M3 – 250	M4 – 280	M5 - 370
P1 – 300	P4 – 400	P3 – 250		P2 – 170

P4 – Unallocated as the process size is too big

Worst Fit Allocation

M1 – 300	M2 – 500	M3 – 250	M4 – 280	M5 - 370
P1 – 250	P5- 170			P1 - 300

Process 2 & 4 unallocated as insufficient space

b. Virtual Memory

First-In-first out

Random 12 digit Number: 158232114263

Paging Accessing Sequence	1	5	8	2	3	2	1	1	4	2	6	3
Page Entry 0	1	1	1	1	3	3	3	3	3	3	3	3
Page Entry 1		5	5	5	5	5	1	1	1	1	1	1
Page Entry 2			8	8	8	8	8	8	4	4	4	4
Page Entry 3				2	2	2	2	2	2	2	6	6
Page Fault	*	*	*	*	*		*		*			

Total Page Fault: 7

Random Paging

Paging Accessing Sequence	1	5	8	2	3	2	1	1	4	2	6	3
Page Entry 0	1	1	1	1	1	1	1	1	4	4	6	6
Page Entry 1		5	5	5	3	3	3	3	3	3	3	3
Page Entry 2			8	8	8	8	8	8	8	8	8	8
Page Entry 3				2	2	2	2	2	2	2	2	2
Page Fault	*	*	*	*	*				*		*	

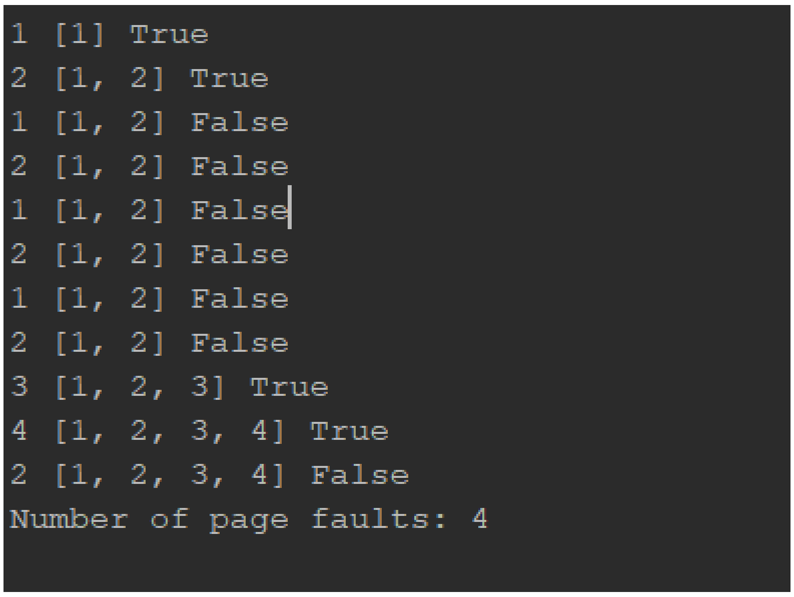
Total Page Faults: 7

[First in first out and random paging]

c. Paging Program

[Commented code and examples of the program running]

```python
'''
Paging program that employs the random paging approach.
This program takes in 2 arguemnts:
1. Sequence that the pages are called
2. Number of page frames in the memory

And outputs a table displaying how the pages are stored in RAM overtime.
'''
import random
from time import sleep

memory=[]
sequence=[]
def paging(sequence_input,frames_input):
    fault = None
    page_faults = 0
    # Converts inputs into appropriate types
    frames=int(frames_input)
    # Parses input sequence into list
    for i in str(sequence_input):
        sequence.append(int(i))
    # Iterates through the sequence, if the page exists in memory, no page faults occurs. Else, page fault occurs, page fault counter increases by 1 and a random page in memory is replaced with the current page.
    for i in range(len(sequence)):
        if sequence[i] in memory:
            fault= False
        else:
            fault=True
            page_faults += 1
            if len(memory) >= frames:
                val = random.randint(0, len(memory) - 1)
                memory[val] = sequence[i]
            else:
                memory.append(sequence[i])
        print(sequence[i], memory,fault)
    return("Number of page faults: "+ str(page_faults))

print(paging(12121212342,4))
```



```
1 [1] True
2 [1, 2] True
1 [1, 2] False
2 [1, 2] False
1 [1, 2] False
2 [1, 2] False
1 [1, 2] False
2 [1, 2] False
3 [1, 2, 3] True
4 [1, 2, 3, 4] True
2 [1, 2, 3, 4] False
Number of page faults: 4
```

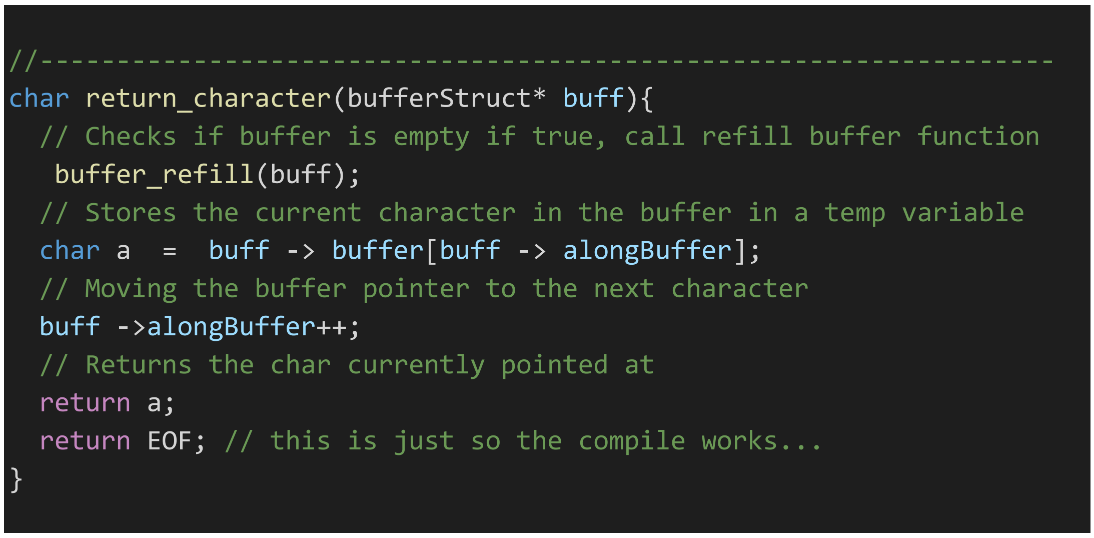
Lab Activity 8 Cache Buffer

a. Brief Description of Cache Buffer Activity

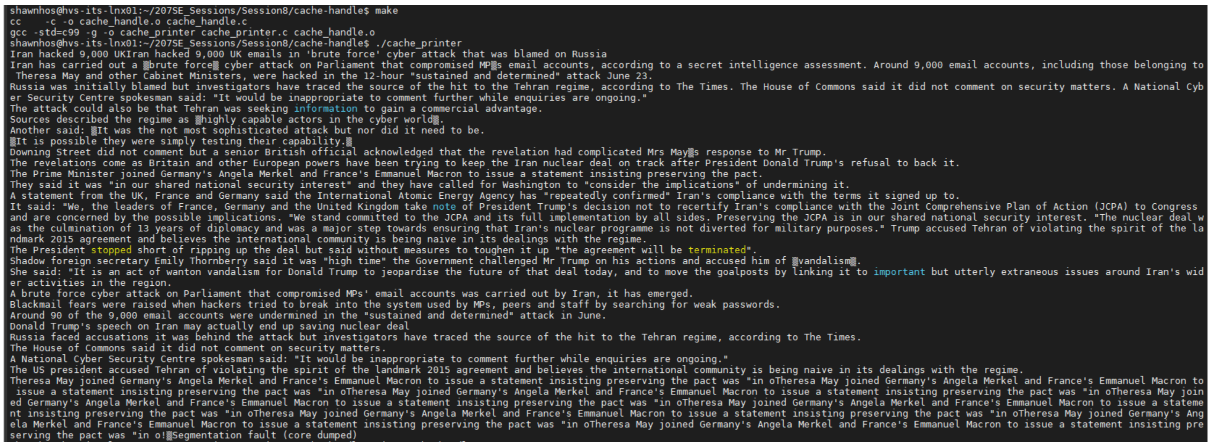
The cache buffer activity involves printing out a text file by first pointing to the text file and pulling a character from the buffer and printing that character. It then increases the buffer count by one, moving the pointer to the next character and printing the next character. This is all done in a while loop which iterates through the whole text file until it reaches the end of the file where the while loop is broken and all the text in the text file is printed onto the screen.

b. Commented implementation of the cr_handle function

[Comment code of the cr_handle function here]



```c
//-------------------------------------------------------------------
char return_character(bufferStruct* buff){
  // Checks if buffer is empty if true, call refill buffer function
   buffer_refill(buff);
  // Stores the current character in the buffer in a temp variable
  char a  =  buff -> buffer[buff -> alongBuffer];
  // Moving the buffer pointer to the next character
  buff ->alongBuffer++;
  // Returns the char currently pointed at
  return a;
  return EOF; // this is just so the compile works...
}
```



```
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session8/cache-handle$ make
cc    -c -o cache_handle.o cache_handle.c
gcc -std=c99 -g -o cache_printer cache_printer.c cache_handle.o
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session8/cache-handle$ ./cache_printer
Iran hacked 9,000 UKIran hacked 9,000 UK emails in 'brute force' cyber attack that was blamed on Russia
Iran has carried out a ▒brute force▒ cyber attack on Parliament that compromised MP▒s email accounts, according to a secret intelligence assessment. Around 9,000 email accounts, including those belonging to
 Theresa May and other Cabinet Ministers, were hacked in the 12-hour "sustained and determined" attack June 23.
Russia was initially blamed but investigators have traced the source of the hit to the Tehran regime, according to The Times. The House of Commons said it did not comment on security matters. A National Cyb
er Security Centre spokesman said: "It would be inappropriate to comment further while enquiries are ongoing."
The attack could also be that Tehran was seeking information to gain a commercial advantage.
Sources described the regime as ▒highly capable actors in the cyber world▒.
Another said: ▒It was the not most sophisticated attack but nor did it need to be.
▒It is possible they were simply testing their capability.▒
Downing Street did not comment but a senior British official acknowledged that the revelation had complicated Mrs May▒s response to Mr Trump.
The revelations come as Britain and other European powers have been trying to keep the Iran nuclear deal on track after President Donald Trump's refusal to back it.
The Prime Minister joined Germany's Angela Merkel and France's Emmanuel Macron to issue a statement insisting preserving the pact.
They said it was "in our shared national security interest" and they have called for Washington to "consider the implications" of undermining it.
A statement from the UK, France and Germany said the International Atomic Energy Agency has "repeatedly confirmed" Iran's compliance with the terms it signed up to.
It said: "We, the leaders of France, Germany and the United Kingdom take note of President Trump's decision not to recertify Iran's compliance with the Joint Comprehensive Plan of Action (JCPA) to Congress
and are concerned by the possible implications. "We stand committed to the JCPA and its full implementation by all sides. Preserving the JCPA is in our shared national security interest. "The nuclear deal w
as the culmination of 13 years of diplomacy and was a major step towards ensuring that Iran's nuclear programme is not diverted for military purposes." Trump accused Tehran of violating the spirit of the la
ndmark 2015 agreement and believes the international community is being naive in its dealings with the regime.
The President stopped short of ripping up the deal but said without measures to toughen it up "the agreement will be terminated".
Shadow foreign secretary Emily Thornberry said it was "high time" the Government challenged Mr Trump on his actions and accused him of ▒vandalism▒.
She said: "It is an act of wanton vandalism for Donald Trump to jeopardise the future of that deal today, and to move the goalposts by linking it to important but utterly extraneous issues around Iran's wid
er activities in the region.
A brute force cyber attack on Parliament that compromised MPs' email accounts was carried out by Iran, it has emerged.
Blackmail fears were raised when hackers tried to break into the system used by MPs, peers and staff by searching for weak passwords.
Around 90 of the 9,000 email accounts were undermined in the "sustained and determined" attack in June.
Donald Trump's speech on Iran may actually end up saving nuclear deal
Russia faced accusations it was behind the attack but investigators have traced the source of the hit to the Tehran regime, according to The Times.
The House of Commons said it did not comment on security matters.
A National Cyber Security Centre spokesman said: "It would be inappropriate to comment further while enquiries are ongoing."
The US president accused Tehran of violating the spirit of the landmark 2015 agreement and believes the international community is being naive in its dealings with the regime.
Theresa May joined Germany's Angela Merkel and France's Emmanuel Macron to issue a statement insisting preserving the pact was "in oTheresa May joined Germany's Angela Merkel and France's Emmanuel Macron to
 issue a statement insisting preserving the pact was "in oTheresa May joined Germany's Angela Merkel and France's Emmanuel Macron to issue a statement insisting preserving the pact was "in oTheresa May join
ed Germany's Angela Merkel and France's Emmanuel Macron to issue a statement insisting preserving the pact was "in oTheresa May joined Germany's Angela Merkel and France's Emmanuel Macron to issue a stateme
nt insisting preserving the pact was "in oTheresa May joined Germany's Angela Merkel and France's Emmanuel Macron to issue a statement insisting preserving the pact was "in oTheresa May joined Germany's Ang
ela Merkel and France's Emmanuel Macron to issue a statement insisting preserving the pact was "in oTheresa May joined Germany's Angela Merkel and France's Emmanuel Macron to issue a statement insisting pre
serving the pact was "in o!▒Segmentation fault (core dumped)
```

c. Comment updated code to show that each byte is being read, and when the buffer is being refilled.

[Comment code outlining your changes here]

cache_handle.c

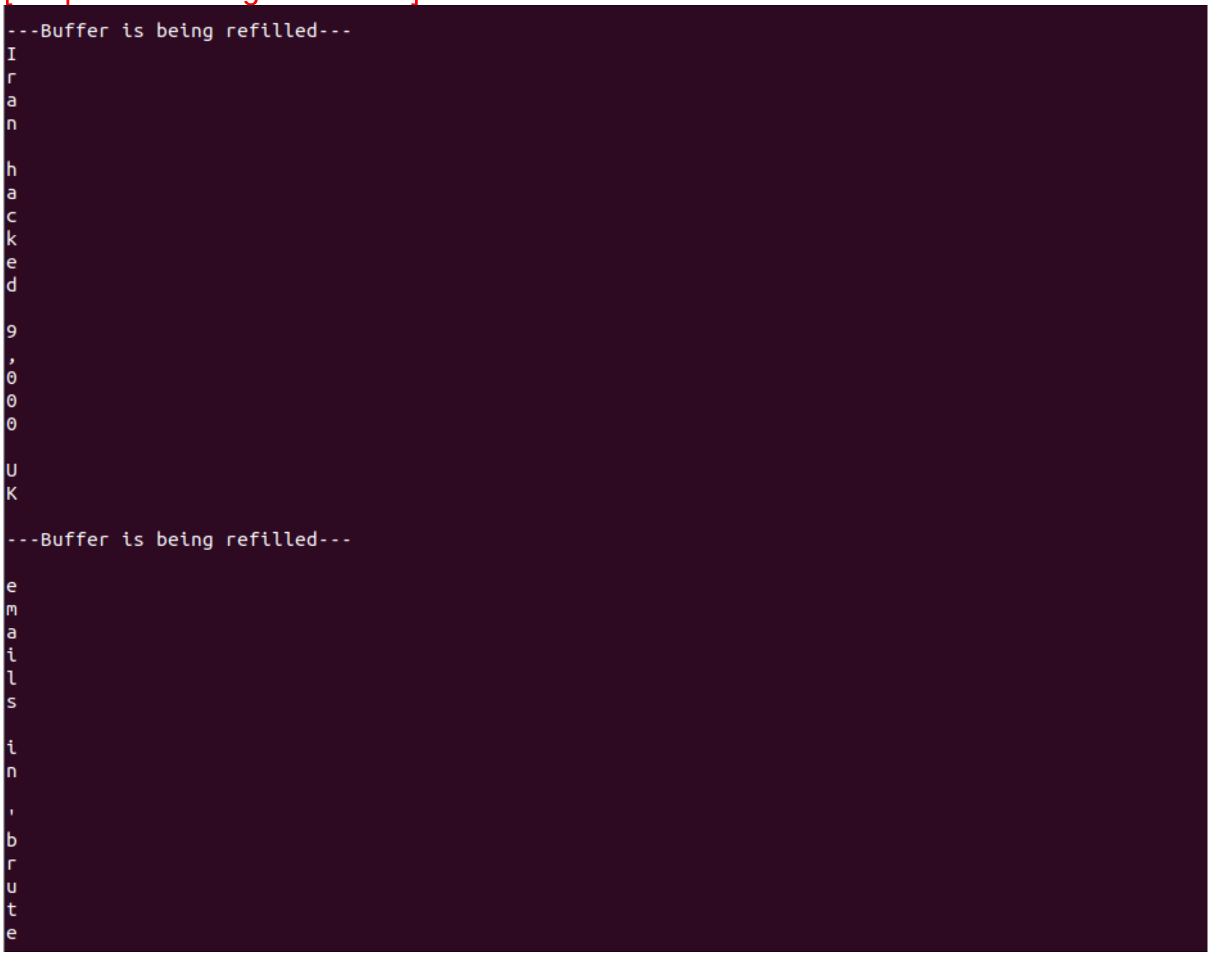
```
#include "cache_handle.h"


//http://www.phim.unibe.ch/comp_doc/c_manual/C/SYNTAX/struct.html
//http://vergil.chemistry.gatech.edu/resources/programming/c-
tutorial/structs.html


int buffer_refill(bufferStruct* buff){
  //Refills a buffer
  //Only works when completely used buffer
  if(buff->alongBuffer!=buff->bufferLength)
    return 0;
  else{
    // Everytime the buffer is being refilled, display it.
    printf("\n---Buffer is being refilled---\n");
    buff->alongBuffer=0;
    int len=fread(buff->buffer, sizeof(char), buff->bufferLength, buf
f->file);
    //If we didn't fill the buffer, fill up with EOF
    if(len<buff->bufferLength)
      for(int i=len;i<buff->bufferLength;i++)
    buff->buffer[i]=EOF;  //Accessing like an array!
    return len;
  }

}

void file_close(bufferStruct* buff){
  free(buff->buffer);
  fclose(buff->file);
}


bufferStruct* file_open(char * filename, int buffersize){

  //Info on malloc
  //http://www.space.unibe.ch/comp_doc/c_manual/C/FUNCTIONS/malloc.ht
ml
  FILE* f;
  if ((f = fopen(filename, "r")) == NULL){
    fprintf(stderr, "Cannot open %s\n", filename);
    return 0;
  }

  bufferStruct* initBuffer=(bufferStruct*)malloc(sizeof(bufferStruct)
);
  initBuffer->file=f;
  initBuffer->bufferLength=buffersize;
  initBuffer->alongBuffer=buffersize; //Start off with no characters,
 so refill will work as expected
  initBuffer->buffer=(char*)malloc(sizeof(char)*buffersize);

  buffer_refill(initBuffer);
  return initBuffer;
}



//--------------------------------------------------------------------
char return_character(bufferStruct* buff){
  // Checks if buffer is empty if true, call refill buffer function
   buffer_refill(buff);
  // Stores the current character in the buffer in a temp variable
  char a  =  buff -> buffer[buff -> alongBuffer];
  // Moving the buffer pointer to the next character
  buff ->alongBuffer++;
  // Returns the char currently pointed at
  return a;
  return EOF; // this is just so the compile works...
}
```

cache_printer.c

```
#include "cache_handle.h"

//Simple file display to show how easy it is to use the cached reader
 functions

int main(){
  char character;

  //Open a file
  bufferStruct* f = file_open("text",20);

  //While there are useful bytes coming from it
  while((character=return_character(f))!=EOF){
    //Printing each byte to a new line by adding \n
    printf("%c\n",character);
  }

  //Then close the file
  file_close(f);
```

```
    //And finish
    return 0;
}
```

[Output of running code here]



```
---Buffer is being refilled---
I
r
a
n

h
a
c
k
e
d

9
,
0
0
0

U
K

---Buffer is being refilled---

e
m
a
i
l
s

i
n

'
b
r
u
t
e
```

By printing out one character at a time, we know that the text file is being read out char by char, proving that the program is reading the text file by each byte. Every time the buffer requires a refill, it also displays this to the screen.

d. Commented updated code showing the required statistical information as well as how many times the words ‘Iran’, ‘Tehran’ and ‘email’ appear.

Tried to count buffer refills by adding in vars in buffer refill function and calling it from the cache_printer.c file which doesn’t work. Tried many attempts at other methods which all failed. No results to show.

[Output of running code here]

Lab 10: The Cache Buffer from week 8 with system calls

a. Brief description of the activity

This extension of week 8 looks at different approaches/ commands to use compared to the ones in week 8. This week's task involved changing all the fopen, fread & fclose commands to system call open, read and close commands. In this task we had to solve the problem of working with different syntaxes and input parameters in different commands as well as data types accepted and returned. Eg. The fopen command returns a pointer whereas a open sys call returns a non-negative integer.

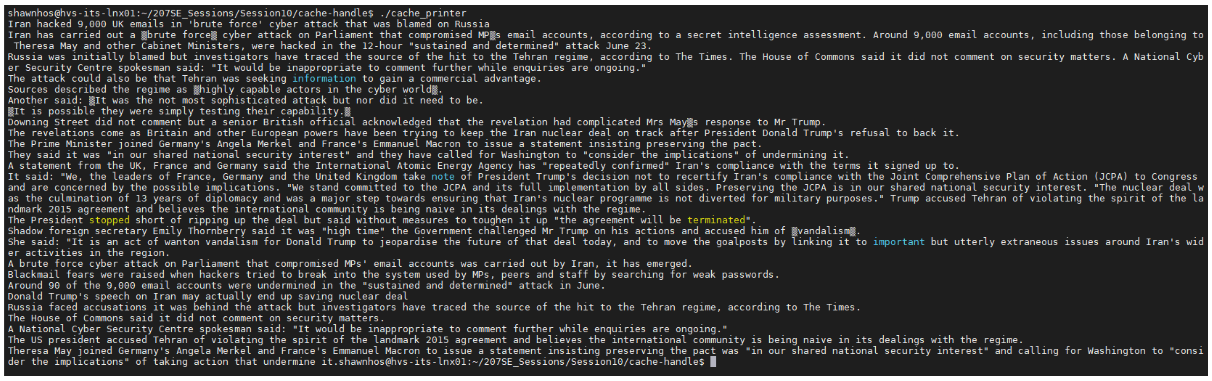
b. Changes the cache_handle library from using the fopen, fread, fclose functions to the system call versions open, read, close

[Commented code outlining your changes to the .h and .c files here]

```
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <unistd.h>
#include "cache_handle.h"

//http://www.phim.unibe.ch/comp_doc/c_manual/C/SYNTAX/struct.html
//http://vergil.chemistry.gatech.edu/resources/programming/c-
tutorial/structs.html


int buffer_refill(bufferStruct* buff){
  //Refills a buffer
  //Only works when completely used buffer
  if(buff->alongBuffer!=buff->bufferLength)
    return 0;
  else{
    buff->alongBuffer=0;
    // Changed fread to read and reorganised the arguments to fit the
correct syntax;
    int len = read(buff->file,buff->buffer,buff->bufferLength);
    //If we didn't fill the buffer, fill up with EOF
    if(len<buff->bufferLength)
      for(int i=len;i<buff->bufferLength;i++)
          buff->buffer[i]=EOF;  //Accessing like an array!
    return len;
  }
}

void file_close(bufferStruct* buff){
  free(buff->buffer);
  // Changed from fclose to close
  close(buff->file);
}


bufferStruct* file_open(char * filename, int buffersize){
//Info on malloc
//http://www.space.unibe.ch/comp_doc/c_manual/C/FUNCTIONS/malloc.html
// Created a variable f of type int instead of using FILE* pointer;
int f;
if ((f = open(filename, O_RDONLY)) == -1 ){
  fprintf(stderr, "Cannot open %s\n", filename);
  return 0;
}

  bufferStruct* initBuffer=(bufferStruct*)malloc(sizeof(bufferStruct))
;
  initBuffer->file=f;
  initBuffer->bufferLength=buffersize;
  initBuffer-
>alongBuffer=buffersize; //Start off with no characters, so refill wil
l work as expected
  initBuffer->buffer=(char*)malloc(sizeof(char)*buffersize);

  buffer_refill(initBuffer);
  return initBuffer;
}




//--------------------------------------------------------------------
char return_character(bufferStruct* buff){
  // Checks if buffer is empty if true, call refill buffer function
  buffer_refill(buff);
  // Stores the current character in the buffer in a temp variable
  char a  =  buff -> buffer[buff -> alongBuffer];
  // Moving the buffer pointer to the next character
  buff ->alongBuffer++;
  // Returns the char currently pointed at
  return a;
  return EOF; // this is just so the compile works...
}
```

```
#include <stdio.h>
#include <stdlib.h>

//The internals of this struct aren't important
//from the user's point of view
typedef struct{
  int file;         //File being read changed to an int var
  int bufferLength;  //Fixed buffer length
  int alongBuffer;    //Current point in the buffer
  char* buffer;      //A pointer to a piece of memory
                     //  same length as "bufferlength"
} bufferStruct;




//Open a file with a given size of buffer to cache with
bufferStruct* file_open(char* filename, int buffersize);


//Close an open file
void file_close(bufferStruct* buff);

//Read a byte.  Will return EOF if empty.
char return_character(bufferStruct* buff);



//-----------------------------------------------------------

//Refill an empty buffer.  Not intended for users
int buffer_refill(bufferStruct* buff);
```

[Output from running code here]



```
shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session10/cache-handle$ ./cache_printer
Iran hacked 9,000 UK emails in 'brute force' cyber attack that was blamed on Russia
Iran has carried out a ▒brute force▒ cyber attack on Parliament that compromised MP▒s email accounts, according to a secret intelligence assessment. Around 9,000 email accounts, including those belonging to
 Theresa May and other Cabinet Ministers, were hacked in the 12-hour "sustained and determined" attack June 23.
Russia was initially blamed but investigators have traced the source of the hit to the Tehran regime, according to The Times. The House of Commons said it did not comment on security matters. A National Cyb
er Security Centre spokesman said: "It would be inappropriate to comment further while enquiries are ongoing."
The attack could also be that Tehran was seeking information to gain a commercial advantage.
Sources described the regime as ▒highly capable actors in the cyber world▒.
Another said: ▒It was the not most sophisticated attack but nor did it need to be.
▒It is possible they were simply testing their capability.▒
Downing Street did not comment but a senior British official acknowledged that the revelation had complicated Mrs May▒s response to Mr Trump.
The revelations come as Britain and other European powers have been trying to keep the Iran nuclear deal on track after President Donald Trump's refusal to back it.
The Prime Minister joined Germany's Angela Merkel and France's Emmanuel Macron to issue a statement insisting preserving the pact.
They said it was "in our shared national security interest" and they have called for Washington to "consider the implications" of undermining it.
A statement from the UK, France and Germany said the International Atomic Energy Agency has "repeatedly confirmed" Iran's compliance with the terms it signed up to.
It said: "We, the leaders of France, Germany and the United Kingdom take note of President Trump's decision not to recertify Iran's compliance with the Joint Comprehensive Plan of Action (JCPA) to Congress
and are concerned by the possible implications. "We stand committed to the JCPA and its full implementation by all sides. Preserving the JCPA is in our shared national security interest. "The nuclear deal w
as the culmination of 13 years of diplomacy and was a major step towards ensuring that Iran's nuclear programme is not diverted for military purposes." Trump accused Tehran of violating the spirit of the la
ndmark 2015 agreement and believes the international community is being naive in its dealings with the regime.
The President stopped short of ripping up the deal but said without measures to toughen it up "the agreement will be terminated".
Shadow foreign secretary Emily Thornberry said it was "high time" the Government challenged Mr Trump on his actions and accused him of ▒vandalism▒.
She said: "It is an act of wanton vandalism for Donald Trump to jeopardise the future of that deal today, and to move the goalposts by linking it to important but utterly extraneous issues around Iran's wid
er activities in the region.
A brute force cyber attack on Parliament that compromised MPs' email accounts was carried out by Iran, it has emerged.
Blackmail fears were raised when hackers tried to break into the system used by MPs, peers and staff by searching for weak passwords.
Around 90 of the 9,000 email accounts were undermined in the "sustained and determined" attack in June.
Donald Trump's speech on Iran may actually end up saving nuclear deal
Russia faced accusations it was behind the attack but investigators have traced the source of the hit to the Tehran regime, according to The Times.
The House of Commons said it did not comment on security matters.
A National Cyber Security Centre spokesman said: "It would be inappropriate to comment further while enquiries are ongoing."
The US president accused Tehran of violating the spirit of the landmark 2015 agreement and believes the international community is being naive in its dealings with the regime.
Theresa May joined Germany's Angela Merkel and France's Emmanuel Macron to issue a statement insisting preserving the pact was "in our shared national security interest" and calling for Washington to "consi
der the implications" of taking action that undermine it.shawnhos@hvs-its-lnx01:~/207SE_Sessions/Session10/cache-handle$ 
```

c. Change cache_handle library to remove (as far as possible) the effects of caching on the library.

I tried to solve this problem but frankly I do not know enough about C programming and structs to even begin optimising this code to reduce the effects of caching. I will have to brush up on my C programming as well as my assembly language to do better in the next portfolio.

[Output from running code here and if possible prove not using cache here]

References

ARM Technical Support Knowledge Articles (2008) *What is the difference between a von Neumann architecture and a Harvard architecture?* [online] available from http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.faqs/ka3839.html[24 February 2020]

Bao YG, Wang S. (2017) 'Labeled von Neumann architecture for software-defined cloud.' *Journal Of Computer Science and Technology* 32(2), 219–223

Xu Wang, Yongxin Zhu, Linan Huang, (2016) 'A comprehensive reconfigurable computing approach to memory wall problem of large graph computation' *Journal of Systems Architecture*, 70, 59-69