Student id:

Course:

Module: 207SE Operating Systems, Security and Networks

Submission data: 28th Feb 2020

Portfolio 1

Contents

[Lab Activity 1 – Operating Systems Tasks and Programming 5](#_Toc30930444)

[a. Comparison between the Harvard Architecture and Von Neumann Architecture 5](#_Toc30930445)

[b. Programming activity 5](#_Toc30930446)

[Lab Activity 2 – Linux Command Line (Commands and outcomes from a series of small tasks that require use of a number of Linux commands) 6](#_Toc30930447)

[Tasks – Files 6](#_Toc30930448)

[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. 6](#_Toc30930449)

[b. Show evidence of this using the appropriate version of the **ls command**. 6](#_Toc30930450)

[c. Download the script http://www.centerkey.com/tree/tree.sh to your home directory using wget and make the file executable. 6](#_Toc30930451)

[d. Create a directory called **wrongDirectory**. You release it is not what you wanted so delete it. 6](#_Toc30930452)

[e. Create Portfolio1-2020 and Portfolio2-2020 directories in the directory you created in part a. 6](#_Toc30930453)

[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). 6](#_Toc30930454)

[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. 7](#_Toc30930455)

[Linux Commands – Mixed 7](#_Toc30930456)

[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. 7](#_Toc30930457)

[b. Move into the lab1-207SE directory and use the appropriate command to show the current directory. 7](#_Toc30930458)

[c. Display the time of a user (ab0487) last login. 7](#_Toc30930459)

[d. Find out how to prevent the effects of talk, write and wall from interrupting you. What command can you use? 7](#_Toc30930460)

[e. Show the command to verify that www.coventry.ac.uk exists and can accept requests. 7](#_Toc30930461)

[f. Display your name and favourite programming language on the screen using the echo command. 7](#_Toc30930462)

[g. Find out how you can display your username on the screen and at least two ways to display who is logged on. 7](#_Toc30930463)

[h. Use two ways to list the processes that are running. 8](#_Toc30930464)

[i. What are the differences between the Linux commands copy (cp), rename and move? 8](#_Toc30930465)

[j. With a single command, how would you get systems information such as processes, memory, paging and CPU activity? 8](#_Toc30930466)

[Tasks – Document Manipulation 8](#_Toc30930467)

[a. Use **cat** to show the contents of the file. 8](#_Toc30930468)

[b. Use an appropriate command to display the CRC checksum and byte count of the file. 8](#_Toc30930469)

[c. Use **grep** to show only lines not containing the words "she" or "he". Lines contain both “she” and “he” should be shown. 8](#_Toc30930470)

[d. Use **grep** to show the 5 lines above a line containing the text ‘the’. 8](#_Toc30930471)

[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. 8](#_Toc30930472)

[f. Find a command to list the top 3 lines of the **poem.txt** file and then the bottom line of these top 3. 9](#_Toc30930473)

[g. Find a command to split the **poem.txt** file into different files each containing 2 lines. 9](#_Toc30930474)

[h. Use **sort** and **rev** to reverse the sorted contents of poem.txt and append the output to poem2.txt. 9](#_Toc30930475)

[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. 9](#_Toc30930476)

[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**. 9](#_Toc30930477)

[k. Sort the **poem.txt** file, remove the duplicates and reverse the sorted contents and append the output to **poem2.txt**. 9](#_Toc30930478)

[l. Create an **alias** so rather than having to type the full command for k) you can type **yourSort**. 9](#_Toc30930479)

[Lab Activity 4 Bootloader 10](#_Toc30930480)

[a. Brief description of the Lab activity and what you did 10](#_Toc30930481)

[b. Boot pragma Linux with bochs 10](#_Toc30930482)

[c. Make a bootloader that displays your student details and diamond 10](#_Toc30930483)

[Commented bootloader code to display your student details and diamond 10](#_Toc30930484)

[Output from bochs showing student details and diamond 10](#_Toc30930485)

[Lab Activity 5 Exploring what is going on outside the processor 11](#_Toc30930486)

[a. List the information found in the /proc directory about the computer CPUs. 11](#_Toc30930487)

[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. 11](#_Toc30930488)

[c. Show the number of CPUs, the producer of the CPUs and the CPU model. 11](#_Toc30930489)

[d. Display a list of all modules that have been loaded by the system. 11](#_Toc30930490)

[e. Using the /proc/diskstats show the names of the output devices and the number of megabytes read per second during the sampled interval. 11](#_Toc30930491)

[f. Menu based shell script. 11](#_Toc30930492)

[Lab Activity 6 Memory Management 12](#_Toc30930493)

[a. Memory Allocation 12](#_Toc30930494)

[b. Virtual Memory 12](#_Toc30930495)

[c. Paging Program 12](#_Toc30930496)

[Lab Activity 8 Cache Buffer 13](#_Toc30930497)

[a. Brief Description of Cache Buffer Activity 13](#_Toc30930498)

[b. Commented implementation of the cr\_handle function 13](#_Toc30930499)

[c. Comment updated code to show that each byte is being read, and when the buffer is being refilled. 13](#_Toc30930500)

[d. Commented updated code showing the required statistical information as well as how many times the words ‘Iran’, ‘Tehran’ and ‘email’ appear. 13](#_Toc30930501)

[Lab 10: The Cache Buffer from week 8 with system calls 14](#_Toc30930502)

[a. Brief description of the activity 14](#_Toc30930503)

[b. Changes the cache\_handle library from using the fopen, fread, fclose functions to the system call versions open, read, close 14](#_Toc30930504)

[c. Changes cache\_handle library to remove (as far as possible) the effects of caching on the library. 14](#_Toc30930505)

[References 15](#_Toc30930506)

# Lab Activity 1 – Operating Systems Tasks and Programming

## Comparison between the Harvard Architecture and Von Neumann Architecture

[Paragraph on the future of operating systems here]

## Programming activity

[Commented code here]

[Outcomes from code here]

# 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

### 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.

[Linux Command and command running]

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

[Linux Command and command running]

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

[Linux Command and command running]

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

[Linux Command and command running]

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

[Linux Command and command running]

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

[Linux Command and command running]

### 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.

[Linux Command and command running]

## Linux Commands – Mixed

### 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.

[Linux Command and command running]

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

[Linux Command and command running]

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

[Linux Command and command running]

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

[Description here]

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

[Linux Command and command running]

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

[Linux Command and command running]

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

[Linux Command and command running]

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

[Linux Command and command running]

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

[Description here]

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

[Linux Command and command running]

## Tasks – Document Manipulation

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

[Linux Command and command running]

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

[Linux Command and command running]

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

[Linux Command and command running]

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

[Linux Command and command running]

### 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.

[Linux Command and command running]

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

[Linux Command and command running]

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

[Linux Command and command running]

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

[Linux Command and command running]

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

[Linux Command and command running]

### 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**.

[Linux Command and command running]

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

[Linux Command and command running]

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

[Linux Command and command running]

# Lab Activity 4 Bootloader

## Brief description of the Lab activity and what you did

[Brief Description of activity]

## Boot pragma Linux with bochs

[Screenshot or script showing how you compiled and created the Linux image and used Bochs to simulate bootloader]

## Make a bootloader that displays your student details and diamond

### Commented bootloader code to display your student details and diamond

[Commented code here]

### Output from bochs showing student details and diamond

[Output from bochs show student details and triangle]

# Lab Activity 5 Exploring what is going on outside the processor

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

[Linux command and output]

## 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.

[Linux command and output]

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

[Linux command and output]

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

[Linux command and output]

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

[Linux command and output]

## Menu based shell script.

[Shell script and output]

# Lab Activity 6 Memory Management

## Memory Allocation

[First fit, Best fit and Worst fit allocation]

## Virtual Memory

[First in first out and random paging]

## Paging Program

[Commented code and examples of the program running]

# Lab Activity 8 Cache Buffer

## Brief Description of Cache Buffer Activity

[Brief Description here]

## Commented implementation of the return\_character function

[Comment code of the cr\_handle function here]

[Output of running code here]

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

[Comment code outlining your changes here]

[Output of running code here]

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

[Comment code outlining your changes here]

[Output of running code here]

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

## Brief description of the activity

[Brief description here]

## 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]

[Output from running code here]

## Changes cache\_handle library to remove (as far as possible) the effects of caching on the library.

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

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

# References