

Assignment

by Lloyd Jones

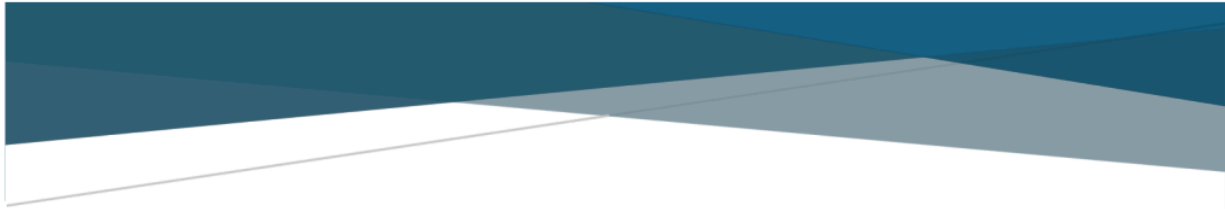
Submission date: 10-Apr-2025 08:14PM (UTC+0100)

Submission ID: 255401963

File name: 57727_Lloyd_Jones_Assignment_644541_1818430607.docx (842.24K)

Word count: 2061

Character count: 10117



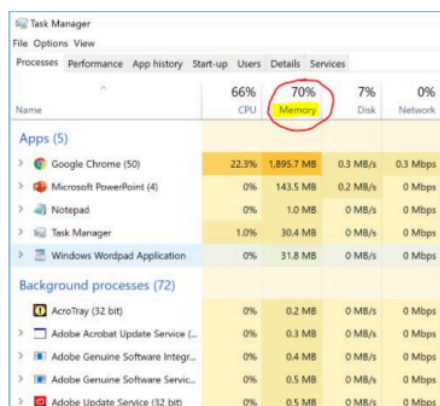
OPERATING SYSTEMS

Computer Architecture and Operating Systems

Lloyd Jones (2405771)
2405771@student.uwtsd.ac.uk

1. Key functions of general operating systems

A general operating system (OS) is software that manages your computer including its hardware and software. It allows you to use your computer like you know it, stable and running applications. The main goal of an operating system is to provide a stable environment and serve as an interface between the user and system. Below are some examples about the key functions of a general OS.



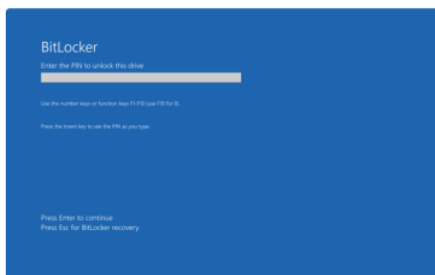
The screenshot shows the Windows Task Manager Performance tab. At the top, it displays overall system usage: 66% CPU, 70% Memory (highlighted with a red circle), 7% Disk, and 0% Network. Below this, a table lists running applications and background processes with their respective resource usage.

Name	CPU	Memory	Disk	Network
Apps (5)				
Google Chrome (50)	22.3%	1,895.7 MB	0.3 MB/s	0.3 Mbps
Microsoft PowerPoint (4)	0%	143.5 MB	0.2 MB/s	0 Mbps
Notepad	0%	1.0 MB	0 MB/s	0 Mbps
Task Manager	1.0%	30.4 MB	0 MB/s	0 Mbps
Windows Wordpad Application	0%	31.8 MB	0 MB/s	0 Mbps
Background processes (72)				
AcroTray (32 bit)	0%	0.2 MB	0 MB/s	0 Mbps
Adobe Acrobat Update Service (...)	0%	0.3 MB	0 MB/s	0 Mbps
Adobe Genuine Software Integr...	0%	0.4 MB	0 MB/s	0 Mbps
Adobe Genuine Software Servic...	0%	0.5 MB	0 MB/s	0 Mbps
Adobe Update Service (32 bit)	0%	0.5 MB	0 MB/s	0 Mbps

Memory & Process Management

A general Operating System (OS) plays a crucial role in managing memory and processes to ensure the system operates smoothly and while your computer is switched on, hundreds if not thousands of processes are running in the background. The image below was taken from task manager in windows which shows the allocated memory to different process that are running.

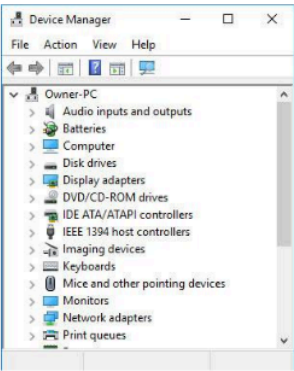
System Security



Windows system security has several built-in security measures to keep you and your system safe. Features such as user authentication and access control, malware protection, secure boot and network security. All these features are constantly active unless told otherwise to maintain system integrity ensuring smooth and efficient operation.

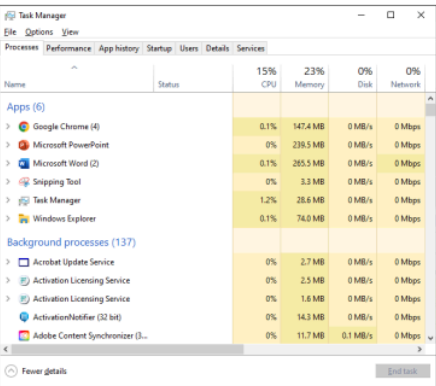
Some examples of windows system security include BitLocker Encryption, Microsoft Defender and Windows Update.

Device Management



Windows device management includes a set of tools and features which allows administrators and users to control devices that are currently connected to the system. This can be devices such as keyboards, mice, printers and other external devices. Device manager shown here is the most common forms that windows offers, with an extensive list of devices that are connected to the system. There are several drop downs within this list from audio devices to monitors which will show you the devices connected in these sections.

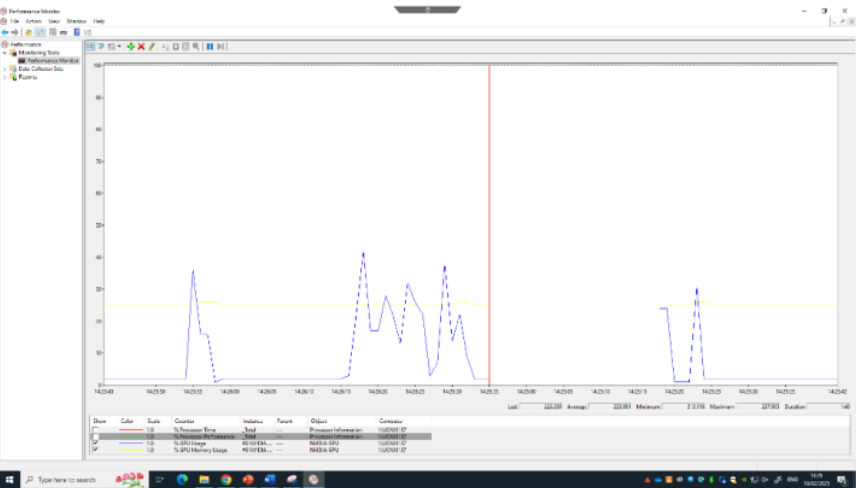
System Performance Monitoring



Task Manager

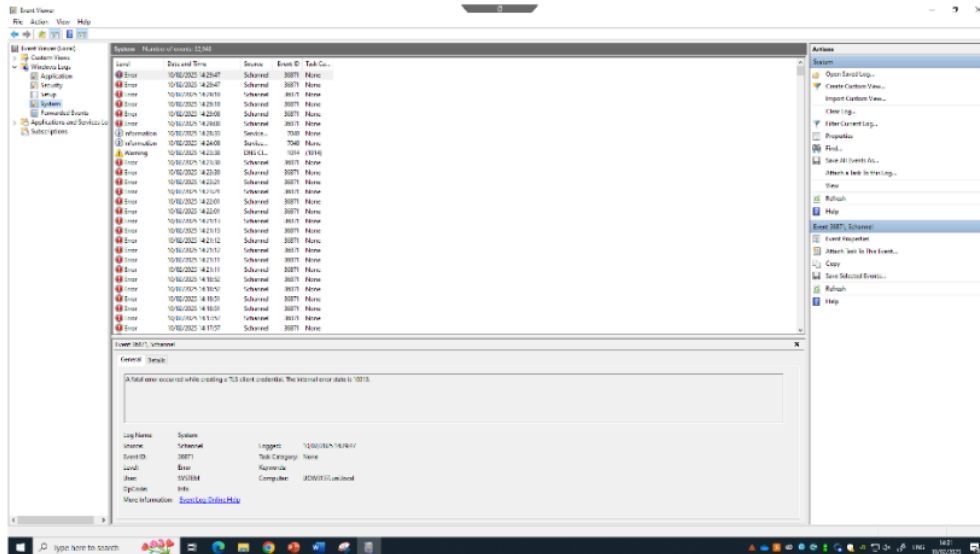
This is task manager which is a friendly resource monitoring tool within windows. It can show the current apps open and all background processes that go on behind the scenes of every machine. Across the top, there are different tabs which include the processes, performance and app history which will show you necessary information. The processes tab will show you all open apps and background processes. Performance will show you how much of the computers resources are being used such as CPU, GPU

and RAM usage. App history is a simple history of apps you have recently accessed.



Performance Monitor

This is performance monitor which can be used for a number of things. The graph I currently have open is the GPU and memory usage. By adding what's a counter, it will plot a graph in this case with the GPU and memory usage however you can add a massive amount of counters which will track all aspects of you system relating to performance.



Event Viewer

This is event viewer and it allows us to view system event logs which can be filtered into significant errors, warning and information messages. This is shown in a list and details can be viewed as to what error, warning or information message is shown.

2. Use of Windows OS CLI Commands

Windows CLI commands (Command Line Interface) are commands that can be commonly found in Command Prompt which directly access the operating system. This can be used for managing files or folders, troubleshooting and task automation. This can sometimes be effective if you know what you are doing but can be extremely harmful to your system if the correct knowledge is not applied.

Below are some commands demonstrating a few simple features of Windows CLI Commands:

[1]

```
PS C:\Users\soac-azure> echo "Hello World">newfile.txt
PS C:\Users\soac-azure> type newfile.txt
Hello World
PS C:\Users\soac-azure>
```

[1] This command was used in Windows Powershell and was used to create a new file which contained the text 'Hello World'. This can be developed to create a file which can contain whatever you want.

[2]

```
PS C:\Users\soac-azure> echo "Hello World">newfile.txt
PS C:\Users\soac-azure> type newfile.txt
Hello World
PS C:\Users\soac-azure> del newfile.txt
PS C:\Users\soac-azure> echo "Hello World">newfile.txt
PS C:\Users\soac-azure>
```

[2] This command was used to delete the newly created file. This can be used if the file you create is not to your liking, or sometime in the future can be deleted.

[3]

```

PS C:\Users\soac-azure> net statistics workstation
Workstation Statistics for \\JMGWL4Azure

Statistics since 3/24/2025 10:04:55 AM

Bytes received                                0
Server Message Blocks (SMBs) received        8
Bytes transmitted                             0
Server Message Blocks (SMBs) transmitted     0
Read operations                              0
Write operations                              0
Raw reads denied                             0
Raw writes denied                            0

Network errors                               0
Connections made                             0
Reconnections made                           0
Server disconnects                           0

Sessions started                             0
Hung sessions                               0
Failed sessions                              0
Failed operations                            0
Use count                                    0
Failed use count                             0

The command completed successfully.
PS C:\Users\soac-azure>

```

[3] The 'net' command seen above, can be used to view, update or fix existing network settings. This can be used to manage and troubleshoot networks including resources, account and services. Typically used by network managers to quickly identify and solve problems.

[4]

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\soac-azure> net user username password /add
The command completed successfully.

PS C:\Users\soac-azure> net user

User accounts for \\JMGWL4Azure

-----
DefaultAccount      Guest              soac-azure
username            WDAGUtilityAccount
The command completed successfully.

PS C:\Users\soac-azure> net localgroup administrators username /add
The command completed successfully.

PS C:\Users\soac-azure>

```

[4] The above command 'net user' followed by the username and password allows the user to add new users to the system. This can be further extended to add administrators allowing for higher access permissions to the systems.

[5]

```
PS C:\Users\soac-azure> net user username /delete
The command completed successfully.

PS C:\Users\soac-azure>
```

[5] The above command is an extension of [4], allowing the user to delete other users. This has to be done by including the username of the user which needs to be deleted followed by '/delete', this will then successfully run the command, deleting the user from the system.

3. Use of Linux OS CLI Commands

Linux CLI Commands, similar to Windows shown above, offer the ability to control a wide range of tasks throughout the operating system. This can be tasks such as navigating, managing and creating files, interacting with the core system and automating tasks. Linux offers a powerful and efficient way to interact with the

operating system and is open sourced, unlike Windows, opening up opportunities for more creativity and technical abilities.

[6]

```
This message is shown once a day. To disable it please create the
/home/l4soac/.hushlogin file.
l4soac@JMGWL4Azure:~$ ls
l4soac@JMGWL4Azure:~$ touch file.txt
l4soac@JMGWL4Azure:~$ echo "Hello World!!">file.txt
echo "Hello Worldtouch file.txt">file.txt
l4soac@JMGWL4Azure:~$ cat file.txt
Hello Worldtouch file.txt
l4soac@JMGWL4Azure:~$
```

[6] The above code is showing the operating and process of creating a new text file. This is a .txt file (Text File), which can then be imported with whatever text you would like. In this case, the text file contains 'Hello World' which is displayed once you input the code to display the text file.

[7]

```
l4soac@JMGWL4Azure:~$ ls
file.txt
l4soac@JMGWL4Azure:~$ cd home
-bash: cd: home: No such file or directory
l4soac@JMGWL4Azure:~$ touch file3.txt
l4soac@JMGWL4Azure:~$ echo "Hello world again"> file3.txt
l4soac@JMGWL4Azure:~$ cat file3.txt
Hello world again
l4soac@JMGWL4Azure:~$ ls -l
total 8
-rw-r--r-- 1 l4soac l4soac 26 Mar 24 11:23 file.txt
-rw-r--r-- 1 l4soac l4soac 18 Mar 24 11:25 file3.txt
l4soac@JMGWL4Azure:~$ chmod u+x file3.txt
l4soac@JMGWL4Azure:~$ chmod 755 file3.txt
l4soac@JMGWL4Azure:~$
```

[7] The above code is showing changing the permission levels of the new files created. The 'chmod' code followed by either u+x or 755, ups the permission level, only allowing system administrators to access the new file.

[8]

```

root@kali:~# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
uucp:x:4:2:uucp:/usr/lib/uucp:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uftp:x:10:10:uftp:/var/spool/uftpd:/usr/sbin/nologin
x11:x:11:11:x11:/var/spool/x11:/usr/sbin/nologin
ftp:x:14:14:ftp:/var/ftp:/usr/sbin/nologin
sshd:x:22:22:ssh:/var/run/ssh:/usr/sbin/nologin
nfsnobody:x:65534:65534:nfsnobody:/var/lib/nfs:/usr/sbin/nologin

```

[8] The above code contains all of the groups within the system. This can be used to manage people and their permission levels, allowing for managers of a possible business to allow for certain access levels to different files. A new user can be added by using 'sudo adduser username'. This allows for fast and easy creation of new users once the initial code is learnt.

4. Use of Windows OS Scripts

Windows OS scripts, usually seen in PowerShell, can be used for automating tasks such as system administration, managing software and hardware and overallly improving the efficiency of the system. Once the scrips are learnt, it can make complex tasks simpler and a lot quicker. It offers a powerful environment to the system administrator, allowing tem to oversee the entire operating system and manage accordingly.

[9]

```
C: > Users > soac-azure > Desktop > test.bat
1  @echo off
2
3  :Name
4  Echo What is your Name?
5  set /p Input=Name:
6  echo So %Input% Is your name?
7  echo 1 for yes
8  echo 2 for no
9  Set /p Input=Answer
10 if %Input%==1 (echo Thank you, please continue) else (goto Name)
11
12 :Name
13 Echo What is your favourite sport?
14 set /p Input=Sport:
15 echo So %Input% Is your favourite sport??
16 echo 1 for yes
17 echo 2 for no
18 Set /p Input=Answer
19 if %input%==1 (echo Thank you) else (goto Sport)
20
21
```

[10]

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
So Lloyd Is your name?
1 for yes
2 for no
Answer1
Thank you, please continue
What is your favourite sport?
Sport:Golf
So Golf Is your favourite sport??
1 for yes
2 for no
Answer1
Thank you
c:\Users\soac-azure\Desktop>
```

The above two sections of code [9] being the code itself and [10] being the output, allow for a user to input their name, and favourite sport. This can be develop and further advanced to allow for more inputs to be codes, eventually being used for full system questions, allowing the administrator to gather informtion quickly and effectively.

[11]

```

C:\Users> soac-azure > Desktop > test.bat
1  @echo off
2  :menu
3  cls
4  echo USER MANAGEMENT
5  echo 1. Add USER
6  echo 2. View Users
7  echo 3. Delete USER
8  echo 4. Exit
9
10 set /p choice=Enter Choice (1-4):
11
12 if %choice==1 goto add_user
13 if %choice==2 goto view_user
14 if %choice==3 goto Delete_user
15 if %choice==4 goto Exit
16
17 :add_user
18 cls
19 set/p username=Enter the username:
20 net user %username% /add
21 echo User %username% added successfully!!
22 pause
23 goto menu
24
25 :view_user
26 cls
27 net user
28 pause
29 goto menu
30
31 :Delete_user
32 cls
33 set / username=Enter username
34 net user %username% /Delete
35 echo User %username% deleted successfully!!
36 pause
37 goto menu
38

```

[12]

```

USER MANAGEMENT
1. Add USER
2. View Users
3. Delete USER
4. Exit
Enter Choice (1-4):

```

[13]

```

Enter the username:Mia
The command completed successfully.

User Mia added successfully!!
Press any key to continue . . .

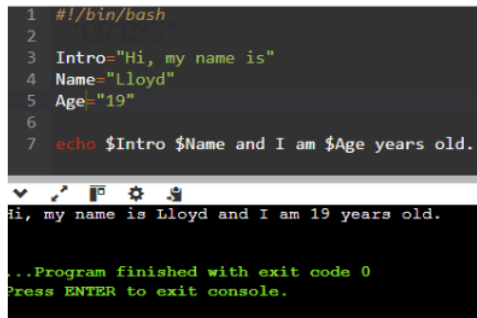
```

[11] Shows us the code, which allows the user or administrator to access the user management tab, created in the initial code to chose one of the following options, add user, view users, delete users and to exit the menu. In [12] and [13], it shows the output and in this case, the new user was created and 'Mia' was entered as the new username.

5. Use of Linux OS Scripts

Linux OS Scripts are typically written in Bash and similar to the Windows OS scripts, they are used for automation of tasks, saving time and streamlining the overall operation of the system. Network management, system maintenance and software updates are some of the examples which these scripts can be used for and can be used by system administrators and ordinary users.

[12]



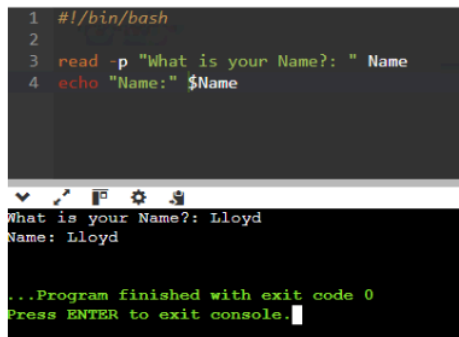
```
1 #!/bin/bash
2
3 Intro="Hi, my name is"
4 Name="Lloyd"
5 Age="19"
6
7 echo $Intro $Name and I am $Age years old.
```

Hi, my name is Lloyd and I am 19 years old.

...Program finished with exit code 0
Press ENTER to exit console.

[12] The above image is a basic variable, which can be coded to the users / administrators liking. It can be used and repeated throughout the code, constantly bringing back the variable which was created at the start of the code.

[13]



```
1 #!/bin/bash
2
3 read -p "What is your Name?: " Name
4 echo "Name:" $Name
```

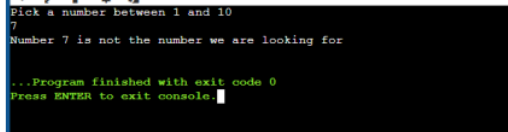
What is your Name?: Lloyd
Name: Lloyd

...Program finished with exit code 0
Press ENTER to exit console.

[13] The above image is a basic input which asks you what is your name. This will then be outputted and can be further advanced to have multiple inputs throughout the code.

[14]

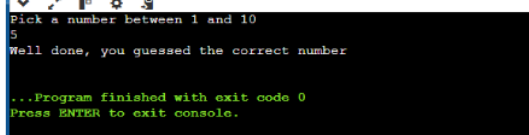
```
1 #!/bin/bash
2
3 echo "Pick a number between 1 and 10"
4 read number
5 if [ $number == 5 ];
6 then
7 echo "Well done, you guessed the correct number"
8 else
9 echo "Number $number is not the number we are looking for"
10 fi
```



The terminal window shows the script being executed. It prompts the user to pick a number between 1 and 10. The user enters 7. The script responds with "Number 7 is not the number we are looking for". It then displays "...Program finished with exit code 0" and "Press ENTER to exit console."

[15]

```
1 #!/bin/bash
2
3 echo "Pick a number between 1 and 10"
4 read number
5 if [ $number == 5 ];
6 then
7 echo "Well done, you guessed the correct number"
8 else
9 echo "Number $number is not the number we are looking for"
10 fi
```

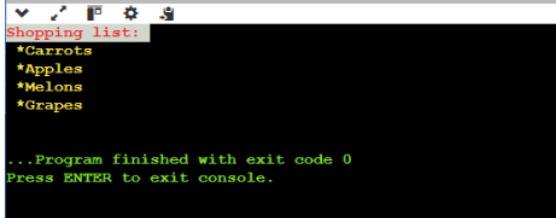


The terminal window shows the script being executed. It prompts the user to pick a number between 1 and 10. The user enters 5. The script responds with "Well done, you guessed the correct number". It then displays "...Program finished with exit code 0" and "Press ENTER to exit console."

The above images, [14] and [15] are a basic example of a for loop. This can be used for many scenarios, both professional and creating mini games. Like the example above, it is a number guessing game. If the number coded, in this case 5 is inputted, you will win the game, anything else would be incorrect

[16]

```
1 #!/bin/bash
2
3 echo -e "\e[1;47m\e[1;31mShopping list: \e[0m"
4 for Shopping in *Carrots *Apples *Melons *Grapes
5 do
6     echo -e "\e[1;33m $Shopping \e[0m"
7 done
```



[16] The above image is an example of a for loop, and in this case can be used as a shopping list. It can be amended, added to or deleted and can be used and advanced to add more complex loops.

[17]

```
1 #!/bin/bash
2
3 echo "Do you like to play golf?"
4 read -p "Yes/No:" Answer
5 case $Answer in
6     Yes|yes|y|Y)
7         echo "Good answer"
8         ;;
9     No|no|n|N)
10        echo "I would recommend playing or trying it"
11        ;;
12    *)
13        echo "Please enter Yes or No"
14        ;;
15 esac
```



[17] The above image is a case statement and can be used for different inputs. If the user inputs yes, it will output a certain answer, however if it is answer as no, it will give a different answer.

Conclusion

Throughout this assignment, I have discovered several key findings, from Operating System Scripts, to Command Line Interface commands which can be implemented into my everyday computer system. This newly learned knowledge can be further applied to my future roles and also with my current module of software development.

This has provided me with a valuable insight into core functions of the operating system of a computer, both through theory and practical elements, developing knowledge of key concepts such as process management, task automation and user access control.

The creation of OS basic OS scripts with both Windows and Linux allow me to manipulate the operating system to do what I want, and further enhance my ability to complete tasks automatically, improving efficiency and cutting time on certain tasks.

Lecture Activity Log

Introduction to Operating Systems

Q. What is an operating system?

An operating system is a system that is created to run a computer system such as its hardware and software and manages a computer. It gives us an interface to use or manager the computer system.

Q. What are the 5 most commonly used operating systems today?

- Windows, Linux, MacOS, Android and ChromeOS

Q. What is the Linux operating system

- Linux is an operating system that serves as the interface between computer hardware and software based on the UNIX architecture. It is an open-source operating system known for reliability making it a popular choice for servers.

Tutorial 1 – Preparation & OS Types

Q. What is the world's most powerful supercomputer

- El Capitan is now the world's most powerful supercomputer coming online at the end of 2024 with 1742 petaFlops of power located in California

Q. What is the world's most powerful quantum computer

- As of 2025, the world's most powerful quantum computer is considered to be IBM's Condor processor with an expected quantum processor containing 1121 qubits.

Q. How is AI being integrated into operating systems

- AI is now apart of everyday life and companies such as Microsoft with Copilot and Apple with Apple Intelligence are changing the world we live in with generative AI taking over day to day activities. It makes our work more productive than ever, allowing us to complete activities and tasks faster and sometimes AI does it for us with a simple command.

Tutorial 2 – OS Functions

Q. List of common OS functions

- Process Management, memory management, file system management, device management, security and access control, user interface and networking

Tutorial 4 – Linux

Q. What are some of the most common standard sub directories in Linux OS and what do they do

Some of the most common standard subdirectories in a Linux operating system include:

1. **/bin** - Contains essential command binaries that are needed in single-user mode and are used by all users.
2. **/etc** - Contains configuration files for the system.
3. **/home** - Contains the home directories for users.
4. **/lib** - Contains shared libraries needed by system programs.
5. **/tmp** - Used for temporary files created by system and users.
6. **/usr** - Contains multi-user utilities and applications.
7. **/var** - Contains variable data like logs, mail, and databases.
8. **/root** - The home directory for the root user.
9. **/sbin** - Contains essential system binaries.

These directories are part of the standard filesystem hierarchy in Linux.

Assignment

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

65 / 100

Total:		65
1. Engagement with Literature Skills:		
+ Some good use of citations. – No formatted reference list.	20	10
2. Knowledge and Understanding Skills:		
Key functions: + Very good overview of key OS functions documented. + Screenshots showing where some of the functions can be seen/monitored/managed are documented. Win Commands: + Good overview of relevant commands that demonstrates an understanding of the language. - could be extended to include some additional commands. Bash Commands: + Good overview of relevant commands that demonstrates an understanding of the language. - could be extended to include some additional commands. Win Scripts: + Good overview of some syntax areas that demonstrates an understanding of the scripting aspect of the language. - could be extended to include some additional syntax areas. Bash Scripts: + Good overview of the syntax that demonstrates a	20	14

clear understanding of the scripting aspect of the language. – some scripts could be enhanced to demonstrate related research to differentiate and enhance use of the scripting aspect of the language.		
3. Cognitive and Intellectual Skills:		
+ Evidence of good analysis and understanding of OS functions. + Screenshots showing where some of the functions can be seen/monitored/managed are included in the analysis. + Conclusion. – not supported by citations.	10	
4. Practical Application Skills:		
1. Win Commands: + Good use of some relevant Win. language commands. - could be extended to include some additional commands. 2. Bash Commands: + Good use of some relevant Bash commands. - could be extended to include some additional commands. 3. Win Scripts: + Good use of some Win scripts to demonstrate applied knowledge of the core syntax. - could be extended to include some additional syntax areas. 4. Bash Scripts: + Good use of Bash scripts to demonstrate applied knowledge of the core syntax. – some scripts could be further customised to differentiate and enhance the use of the scripting aspect	40	25

of the language		
5. Transferable Skills for Life and Professional Practice:		
+ Good report format, structure, and presentation. - No toc. - Tutorial Activity 1 on OS Types: relevant supplementary information on OS types not included in the Appendix of the report. - Evidence of engagement with a relevant CPD course not documented in the appendix. + Lecture activity evidence documented.	1 0	6
Plagiarism & On-time check:		/

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17
