

Operating Systems Assignment

- Mateusz Jadasz

by Mateusz Jadasz

FILE	17845_MATEUSZ_JADASZ_OPERATING_SYSTEMS_ASSIGNMENT_- _MATEUSZ_JADASZ_331893_789905705.DOCX (976.43K)		
TIME SUBMITTED	25-MAR-2021 09:13PM (UTC+0000)	WORD COUNT	2044
SUBMISSION ID	147988981	CHARACTER COUNT	10613

Operating Systems

Mateusz Jadasz

1901291

Computing (Game Development)

Computer Architecture and Operating Systems

25th March 2021

Table of Contents

Operating Systems	1
Introduction	4
Key Functions of General Operating Systems	4
Use of CLI commands in a Windows / Mac OS	5
cd	5
date	5
dir	5
del	6
ver	6
time	6
ping	7
ipconfig	7
mkdir	8
rename	8
tree	8
echo	9
Use of CLI Commands in a Linux OS	9
cal	9
ls	9
df	9
dig	10
mkdir	10
rmdir	10
pwd	11
mv	11
find	11
help	11
Use of Batch scripts in a Windows / Mac OS	12
Text Output Batch File	12
Website Batch File	12
Password Generator Batch File	12
Use of Bash scripts in a Linux OS	13
Text Output Bash File	13
Variable Test Bash File	13

Random Password Generator..... 13

Conclusion..... 14

References..... 15

Introduction

In this report I will be exploring and analysing different areas and functions of modern operating systems. I will be delving deeper into Linux OS which is specifically a command line operating system, as well as Windows 10 which is a graphical user interface-based operating system. This report is about how both of those operating systems handle some basic functions and features such as data management, file systems, use of commands and use of scripts.

Key Functions of General Operating Systems

The most common and basic features that every system must have is to be able to manage the resources and peripherals of the machine the system runs on, it needs to have a user interface and finally, it needs to be able to execute different software and services [1]. Other functions of an operating system which are very common also include memory and data management, security, input/output management, file systems and networking [2]. The resource and peripheral management are the responsibilities of the operating system which manages not only devices like printers and monitors, but also internal components like the CPU or a graphics card. This feature also links to the Input/Output management, and as an extension, it also slightly links to the user interface of the operating system, as they are all involved in establishing a way for the user to interact with the device, which is one of the many reasons operating systems exist. As mentioned previously, another feature of all operating systems is the user interface of the OS. There are four main types of user interface, those being [3]:

- Command Line
- Menu Driven
- Graphical User Interface
- Touchscreen Graphical Interface

Although there are four of them, the most common ones today are the Command Line User Interface and the Graphical User Interface, however with technology constantly advancing, the Touchscreen Graphical user Interface is also becoming very common across devices such as mobile phones, tablets, and laptops. The third key function of an operating system is the ability to execute additional software as well as providing services and resources for the said software. This is especially important for the security function of the machine, as with it, the computer can protect itself with anti-virus, anti-spyware, and anti-malware software as well as firewalls. The great thing about modernised operating systems like Windows 10, is that they come with an already built-in firewall. Not only that, but anti-virus software often has features of an anti-spyware and an anti-malware software, which makes it more convenient than before [4]. Aside from the three essential features there are the other very important features without which the operating systems essentially is

not of much use or just cannot do its job properly. One of those is security which was already discussed above. Apart from security there are things like file systems such as FAT32, NTFS, ext4 and APFS. There is also networking which allows the operating system and computer to connect to the world wide web, memory and data management which are responsible for things like formatting, optimising, reading/writing data, and defragmentation, as well as input/output which is what allows the user to interact with the system and therefore, the device through peripherals like a monitor, keyboard, mouse, and speakers.

Use of CLI commands in a Windows / Mac OS

cd

```
c:\>cd c:\ExampleFolder1  
c:\ExampleFolder1>
```

The cd or change directory command allows the user to navigate to any directory that is on the computer.

date

```
C:\Users\mtii0>date /t  
13/03/2021
```

The date allows the user to see the date that is set on the machine. Normally when the date is shown, it is done so in a slightly different format and it would allow the user to change the date set, however with the /t argument, the date is set to display only which means the user does not get to change it after it is displayed.

dir

```
c:\ExampleFolder1>dir  
Volume in drive C is Windows  
Volume Serial Number is 0C07-B230  
  
Directory of c:\ExampleFolder1  
  
13/03/2021  06:36    <DIR>          .  
13/03/2021  06:36    <DIR>          ..  
13/03/2021  06:36                0 ExampleFile1.txt  
13/03/2021  06:36                0 ExampleFile2.txt  
13/03/2021  06:36                0 ExampleFile3.txt  
13/03/2021  06:36                0 ExampleFile4.txt  
                4 File(s)                0 bytes  
                2 Dir(s)  70,025,269,248 bytes free
```

The dir command displays every file and sub-directory in the directory the user is in now. It also displays the time and date the said files and directories were created, and it displays the size of the files in bytes. [5]

del

```
C:\ExampleFolder1>dir
Volume in drive C is Windows
Volume Serial Number is 0C07-B230

Directory of C:\ExampleFolder1

13/03/2021  06:38    <DIR>          .
13/03/2021  06:38    <DIR>          ..
13/03/2021  06:36             0 ExampleFile1.txt
13/03/2021  06:36             0 ExampleFile2.txt
13/03/2021  06:36             0 ExampleFile3.txt
13/03/2021  06:36             0 ExampleFile4.txt
13/03/2021  06:38    <DIR>          ExampleSubfolder
               4 File(s)              0 bytes
               3 Dir(s)  70,028,570,624 bytes free

C:\ExampleFolder1>del ExampleFile1
Could Not Find C:\ExampleFolder1\ExampleFile1

C:\ExampleFolder1>del ExampleFile1.txt

C:\ExampleFolder1>dir
Volume in drive C is Windows
Volume Serial Number is 0C07-B230

Directory of C:\ExampleFolder1

13/03/2021  06:44    <DIR>          .
13/03/2021  06:44    <DIR>          ..
13/03/2021  06:36             0 ExampleFile2.txt
13/03/2021  06:36             0 ExampleFile3.txt
13/03/2021  06:36             0 ExampleFile4.txt
13/03/2021  06:38    <DIR>          ExampleSubfolder
               3 File(s)              0 bytes
               3 Dir(s)  70,028,427,264 bytes free
```

The del command quite simply allows the user to delete a directory or file.

ver

```
C:\Users\mtii0>ver

Microsoft Windows [Version 10.0.19041.867]
```

The ver command displays the exact version of the operating system that is being used.

time

```
c:\Users\mtii0>time /t
06:32
```

The time command displays the current time set on the computer.

ping

```
C:\Users\mtii0>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:
Reply from 8.8.8.8: bytes=32 time=12ms TTL=116
Reply from 8.8.8.8: bytes=32 time=13ms TTL=116
Reply from 8.8.8.8: bytes=32 time=12ms TTL=116
Reply from 8.8.8.8: bytes=32 time=11ms TTL=116

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 11ms, Maximum = 13ms, Average = 12ms
```

The ping command allows the user to test if there is a connection established between the computer and a server, by sending out packets of data and seeing if there is a response.

ipconfig

```
C:\Users\mtii0>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Unknown adapter Local Area Connection:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 1:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 2:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter WiFi:

    Connection-specific DNS Suffix  . : home
```

ipconfig is a command often used for finding one's IP, DNS, Gateway and more. It is also commonly used when troubleshooting connections.

mkdir

```
c:\ExampleFolder1>mkdir ExampleSubfolder

c:\ExampleFolder1>dir
Volume in drive C is Windows
Volume Serial Number is 0C07-B230

Directory of c:\ExampleFolder1

13/03/2021  06:38    <DIR>        .
13/03/2021  06:38    <DIR>        ..
13/03/2021  06:36             0 ExampleFile1.txt
13/03/2021  06:36             0 ExampleFile2.txt
13/03/2021  06:36             0 ExampleFile3.txt
13/03/2021  06:36             0 ExampleFile4.txt
13/03/2021  06:38    <DIR>        ExampleSubfolder
               4 File(s)              0 bytes
               3 Dir(s)  70,046,863,360 bytes free
```

The mkdir is a command which creates a folder in the directory the user is currently in

rename

```
C:\ExampleFolder1>rename ExampleFile4.txt ExampleFile5.txt

C:\ExampleFolder1>dir
Volume in drive C is Windows
Volume Serial Number is 0C07-B230

Directory of C:\ExampleFolder1

13/03/2021  06:47    <DIR>        .
13/03/2021  06:47    <DIR>        ..
13/03/2021  06:36             0 ExampleFile2.txt
13/03/2021  06:36             0 ExampleFile3.txt
13/03/2021  06:36             0 ExampleFile5.txt
13/03/2021  06:38    <DIR>        ExampleSubfolder
               3 File(s)              0 bytes
               3 Dir(s)  70,024,986,624 bytes free
```

The rename function can rename any file or directory. As always, in a CLI, it is important to use extensions when referring to a file. In the testing, I renamed ExampleFile4.txt to ExampleFile5.txt.

tree

```
C:\ExampleFolder1>tree
Folder PATH listing for volume Windows
Volume serial number is 0C07-B230
C:..
├──ExampleSubfolder
│   └──ExampleTreeFolder
```

The tree function displays every folder and every subfolder that follows in the directory the user is in at the time. They are displayed in order where the highest is the folder that is in the current directory, then expanding to the next and so on. [6]

echo

```
C:\ExampleFolder1>echo Hello World!
Hello World!
```

Echo is a function which returns the string that user has written after the initial command.

Use of CLI Commands in a Linux OS

For the testing of commands in a Linux OS I used an online terminal, which meant that all files and directories shown in the examples do not belong to me. [7]

cal

```
~$ cal
      March 2021
Su Mo Tu We Th Fr Sa
      1  2  3  4  5  6
 7  8  9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31
```

The cal function displays the calendar for the current month.

ls

```
~$ ls
2021-03-23-070016.term  2021-03-23-070051.term  'Welcome to CoCalc.term'
```

Ls is a function that prints back all the files and directories in the current directory.

df

```
~$ df
Filesystem                                1k-blocks    Used Available Use% Mounted on
overlay                                  122786736  9563696 113206656   8% /
tmpfs                                     65536         0    65536   0% /dev
tmpfs                                    16443880         0   16443880   0% /sys/fs/cgroup
10.105.159.225:/ubuntu2004_2021-03-21/linux/Ubuntu-20.04/bin  975175680  807675904 130014208  87% /bin
10.105.159.225:/ubuntu2004_2021-03-21/linux/Ubuntu-20.04/var  975175680  807675904 130014208  87% /var
10.105.159.225:/project/2021-03-18-180945-5216             975175680  807675904 130014208  87% /cocalc
10.105.159.225:/ubuntu2004_2021-03-21/linux/Ubuntu-20.04/lib  975175680  807675904 130014208  87% /lib
10.105.159.225:/ubuntu2004_2021-03-21/linux/Ubuntu-20.04/opt  975175680  807675904 130014208  87% /opt
10.105.159.225:/ubuntu2004_2021-03-21/linux/Ubuntu-20.04/usr  975175680  807675904 130014208  87% /usr
10.105.159.225:/ubuntu2004_2021-03-21/linux/Ubuntu-20.04/lib64  975175680  807675904 130014208  87% /lib64
10.105.159.225:/ubuntu2004_2021-03-21/linux/Ubuntu-20.04/lib32  975175680  807675904 130014208  87% /lib32
10.105.159.225:/ubuntu2004_2021-03-21/linux/Ubuntu-20.04/libx32  975175680  807675904 130014208  87% /libx32
10.105.159.225:/ubuntu2004_2021-03-21/linux/Ubuntu-20.04/sbin  975175680  807675904 130014208  87% /sbin
10.105.159.225:/ubuntu2004_2021-03-21/linux/Ubuntu-20.04/sbin  975175680  807675904 130014208  87% /ext
tmpfs                                    16443880      24   16443856   1% /tmp
tmpfs                                    16443880         0   16443880   0% /data
10.105.159.225:/ubuntu2004_2021-03-21/linux/Ubuntu-20.04/srv  975175680  807675904 130014208  87% /srv
10.105.159.225:/ubuntu2004_2021-03-21/linux/Ubuntu-20.04/etc  975175680  807675904 130014208  87% /etc
tmpfs                                    16443880         4   16443876   1% /secrets/secret-token
/dev/root                                122786736  9563696 113206656   8% /etc/hosts
tmpfs                                    16443880         4   16443876   1% /secrets/gateway-public
kucalc-default-project-8244f07c-6a56-4a3d-856b-67cf18671af9  3072384      128   3072256   1% /home/user
/var/lib/kubelet/pods/4b246c28-f73f-4dde-ba5d-dcbb680b4f2/volumes/cocalc-zfs/home/.zfs/snapshot  3072384      128   3072256   1% /home/user/.snapshots
shm                                       65536         0    65536   0% /dev/shm
tmpfs                                    16443880         0   16443880   0% /proc/acpi
tmpfs                                    16443880         0   16443880   0% /proc/scsi
tmpfs                                    16443880         0   16443880   0% /sys/firmware
```

The df function is very useful because it displays the available amount of space on the disk. In this case, the command displayed the space of the server that the Linux terminal was running from.

dig

```
~$ dig

; <<>> Dig 9.16.1-Ubuntu <<>>
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 29032
;; flags: qr rd ra; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
; .                        IN      NS

;; ANSWER SECTION:
.                30      IN      NS      a.root-servers.net.
.                30      IN      NS      b.root-servers.net.
.                30      IN      NS      c.root-servers.net.
.                30      IN      NS      d.root-servers.net.
.                30      IN      NS      e.root-servers.net.
.                30      IN      NS      f.root-servers.net.
.                30      IN      NS      g.root-servers.net.
.                30      IN      NS      h.root-servers.net.
.                30      IN      NS      i.root-servers.net.
.                30      IN      NS      j.root-servers.net.
.                30      IN      NS      k.root-servers.net.
.                30      IN      NS      l.root-servers.net.
.                30      IN      NS      m.root-servers.net.

;; Query time: 4 msec
;; SERVER: 10.96.0.10#53(10.96.0.10)
;; WHEN: Tue Mar 23 07:06:40 UTC 2021
;; MSG SIZE rcvd: 431
```

The dig command is used to display and troubleshoot any problems involving the server DNS.

mkdir

```
~$ mkdir examplendir
~$ ls
2021-03-23-070016.term  2021-03-23-070051.term  'Welcome to CoCalc.term'  examplendir
```

The mkdir is a command just like in windows cmd. Mkdir will create a directory in the location the user is in.

rmdir

```
~$ mkdir examplendir2
~$ ls
2021-03-23-070016.term  2021-03-23-070051.term  'Welcome to CoCalc.term'  examplendir  examplendir2
~$ rmdir examplendir2
~$ ls
2021-03-23-070016.term  2021-03-23-070051.term  'Welcome to CoCalc.term'  examplendir
```

Rmdir is the opposite of the mkdir command, where it will remove a directory instead of creating one.

pwd

```
~$ pwd
/home/user
```

The pwd command simply displays the current location of the user in the system. This is extremely useful when trying to reference specific files and directories.

mv

```
~$ mkdir destinationdir
~$ ls
2021-03-23-070016.term  2021-03-23-070051.term  'Welcome to CoCalc.term'  destinationdir  exampledir
~$ mv exampledir destinationdir
~$ ls
2021-03-23-070016.term  2021-03-23-070051.term  'Welcome to CoCalc.term'  destinationdir
~$ cd /home/user/destinationdir
~/destinationdir$ ls
exampledir
```

Mv is a command which is used to move folders and files to a specified location

find

```
~/destinationdir$ find /home/user -iname 2021-03-23-070016.term
/home/user/.snapshots/2021-03-23-073537/2021-03-23-070016.term
/home/user/2021-03-23-070016.term
```

The find command is used to find specific files, along with the -iname pattern, the search is going to be case sensitive of the name of the file. [8]

help

```
~/destinationdir$ help
GNU bash, version 5.0.17(1)-release (x86_64-pc-linux-gnu)
These shell commands are defined internally. Type `help' to see this list.
Type `help name' to find out more about the function `name'.
Use `info bash' to find out more about the shell in general.
Use `man -k' or `info' to find out more about commands not in this list.
```

A star (*) next to a name means that the command is disabled.

job_spec [&]	history [-c] [-d offset] [n] or history -anrw>
((expression))	if COMMANDS; then COMMANDS; [elif COMMANDS; >
. filename [arguments]	jobs [-lnprs] [jobspec ...] or jobs -x comman>
:	kill [-s sigspec -n signum -sigspec] pid >
[arg...]	let arg [arg ...]
[[expression]]	local [option] name[=value] ...
alias [-p] [name=value] ...]	logout [n]
bg [job_spec ...]	mapfile [-d delim] [-n count] [-O origin] [-s>
bind [-lpsvPSVX] [-m keymap] [-f filename] [-q>	popd [-n] [+N -N]
break [n]	printf [-v var] format [arguments]
builtin [shell-builtin [arg ...]]	pushd [-n] [+N -N dir]
caller [expr]	pwd [-LP]
case WORD in [PATTERN [PATTERN]...) COMMANDS>	read [-ers] [-a array] [-d delim] [-i text] [>
cd [-L [-P [-e]] [-@]] [dir]	readarray [-d delim] [-n count] [-O origin] [>
command [-pVv] command [arg ...]	readonly [-aAf] [name=value] ...] or readonl>
compgen [-abdefgjkusv] [-o option] [-A action>	return [n]
complete [-abdefgjkusv] [-pr] [-DEI] [-o opti>	select NAME [in WORDS ... ;] do COMMANDS; do>
compopt [-o +o option] [-DEI] [name ...]	set [-abefhkmnptuvxBCHP] [-o option-name] [-->
continue [n]	shift [n]
coproc [NAME] command [redirections]	shopt [-pqsu] [-o] [optname ...]
declare [-aAfFgIlNrtux] [-p] [name=value] ...>	source filename [arguments]

The help command is what displays the available to the user commands of the terminal, which is very helpful for new users as it also displays the syntax that the commands work with.

Use of Batch scripts in a Windows / Mac OS

Scripts are files which contain algorithm for several purposes. They can be made to do whatever the user needs them to do, and often they are just used so the user has a quick and easy way to do several things at once. The windows operating system supports .bat or batch scripts which is just a format that the windows command prompt supports. It differs slightly to Bash scripts in terms of syntax and the fact that bash files are supported Linux and Unix based operating systems.

Text Output Batch File

```
@echo off
title Example Batch File 1
echo This batch file simply outputs everything that is typed in the 'echo' line. As well as that, the 'title' line of code simply gives the cmd window the name that is specified after 'title'.
pause
```

The first batch file I wrote and decided to test was made to just print out whatever text I wanted it to. The echo command is what displays text for the user while the title function is used to name the cmd window that opens when the batch file is ran.

Website Batch File

```
@echo off
title Example Batch File 2
echo This batch file will look up the IP address of the UWTSD website. It will also attempt to open the site using the default browser
nslookup www.uwtsd.ac.uk
start "" http://www.uwtsd.ac.uk
```

The second batch file is used primarily for finding the IP address of the given website, in this case, the home UWTSD webpage. Aside from using the the echo command again, I also used the nslookup command which will look up the IP or the name of a domain. After that, the batch file will also open the webpage using the default browser [9] [10]

Password Generator Batch File

```
@echo off
echo Password: %random%
pause
```

The third batch file will generate a random password made up of four numbers. It does this by displaying the value of the built-in variable '%random%'.

Use of Bash scripts in a Linux OS

Just as mentioned above, Linux also uses scripts through its terminal, however Linux uses a different format of scripts, those being .sh or bash files, which serve the exact same purpose as the batch files, the Windows command prompt supports. Scripts and their usage can be compared to how modern programming languages allow users to create functions which are set to do exactly what the user needs, and as a quick means to do it.

Text Output Bash File

```
#!/bin/sh
echo "This bash file simply outputs everything that is typed in the 'echo' line of the code"
```

This bash file simply outputs everything that is typed in the 'echo' line of the code

Although it has a slightly different syntax, this bash file serves the same purpose as the very first batch file I wrote above. The script will display whatever text is written after the echo command, which in this case is the text above. [11]

Variable Test Bash File

```
#!/bin/bash
echo "Enter your name:"
read name
echo "Hello $name"
```

```
Enter your name:
mateusz
Hello mateusz
```

This script is a test of user input and output. The script will display the request for the user's name, then when the user enters their name, the input is stored in a variable 'name'. After the input is confirmed, the program will greet the user using their name by referencing the value of the variable, which holds the user input. [12]

Random Password Generator

```
#!/bin/bash
echo "Your password is: $RANDOM"
```

Your password is: 5954

The third bash script uses the built-in 'random' function, just like the Windows command prompt and the batch files Windows operating systems can use. This very small program however can choose numbers up to five digits long. [13]

Conclusion

In conclusion of this report, I have found many different uses of scripts and the command prompt/terminal. I have also found that there are many similarities between how the scripts were formatted and written compared to how today's languages like Java, Python 3 or C++ are structured. I have also learned how useful commands can be, especially when dealing with internet connections and servers.

References

- ["Operating Systems," [Online]. Available:
1 [https://homepage.cs.uri.edu/faculty/wolfe/book/Readings/Reading07.htm#:~:te](https://homepage.cs.uri.edu/faculty/wolfe/book/Readings/Reading07.htm#:~:text=An%20operating%20system%20has%20three,provide%20services%20for%20applications%20software..)
] [xt=An%20operating%20system%20has%20three,provide%20services%20for%20appl](https://homepage.cs.uri.edu/faculty/wolfe/book/Readings/Reading07.htm#:~:text=An%20operating%20system%20has%20three,provide%20services%20for%20applications%20software..)
] [ications%20software..](https://homepage.cs.uri.edu/faculty/wolfe/book/Readings/Reading07.htm#:~:text=An%20operating%20system%20has%20three,provide%20services%20for%20applications%20software..) [Accessed 13 March 2021].
- ["What is Operating System? Types of OS, Features and Examples," [Online].
2 Available: <https://www.guru99.com/operating-system-tutorial.html>. [Accessed 13
] March 2021].
- [C. Falk, "Exploring the UI Universe: Different Types of UI," [Online]. Available:
3 <https://www.altia.com/2014/09/22/different-types-of-ui/>. [Accessed 13 March
] 2021].
- ["8 Ways to Protect Your Computer from Viruses," 19 February 2020. [Online].
4 Available: [https://innotechtoday.com/8-ways-protect-you-computer-from-](https://innotechtoday.com/8-ways-protect-you-computer-from-viruses/)
] [viruses/](https://innotechtoday.com/8-ways-protect-you-computer-from-viruses/). [Accessed 13 March 2021].
- ["SS64," [Online]. Available: <https://ss64.com/nt/>. [Accessed 23 March 2021].
5
]
- [A. Oberneder, "Cmd commands under Windows," [Online]. Available:
6 https://www.thomas-krenn.com/en/wiki/Cmd_commands_under_Windows.
] [Accessed 13 March 2021].
- ["CoCalc," [Online]. Available: [https://cocalc.com/projects/8244f07c-6a56-4a3d-](https://cocalc.com/projects/8244f07c-6a56-4a3d-7856b-67cf18671af9/files/?anonymous=terminal&session=default)
7 [7856b-67cf18671af9/files/?anonymous=terminal&session=default](https://cocalc.com/projects/8244f07c-6a56-4a3d-7856b-67cf18671af9/files/?anonymous=terminal&session=default). [Accessed 23
] March 2021].
- [A. KS, "Hongkiat," 28 February 2016. [Online]. Available:
8 <https://www.hongkiat.com/blog/basic-linux-commands/>. [Accessed 23 March
] 2021].
- [I. Buckley, "How to Create a Batch (BAT) File in Five Simple Steps," 26 October
9 2018. [Online]. Available: [https://www.makeuseof.com/tag/write-simple-batch-](https://www.makeuseof.com/tag/write-simple-batch-bat-file/)
] [bat-file/](https://www.makeuseof.com/tag/write-simple-batch-bat-file/). [Accessed 23 March 2021].
- ["Batch Script Tutorial," [Online]. Available:
1 https://www.tutorialspoint.com/batch_script/index.htm. [Accessed 23 March
0 2021].
]
- ["JDoodle," [Online]. Available: [https://www.jdoodle.com/test-bash-shell-script-](https://www.jdoodle.com/test-bash-shell-script-online/)
1 [online/](https://www.jdoodle.com/test-bash-shell-script-online/). [Accessed 23 March 2021].
1
]

["Replit," [Online]. Available: <https://replit.com/languages/bash>. [Accessed 23
1 March 2021].

2
]

[mwunsch, "Coderwall," 18 June 2019. [Online]. Available:
1 <https://coderwall.com/p/s2ttyg/random-number-generator-in-bash>. [Accessed
3 23 March 2021].

]

Operating Systems Assignment - Mateusz Jadasz

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

Instructor

69/100

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

GRADING FORM: UWTSD GRADING FORM

1. RESEARCH LITERATURE

Extent of research and/or own reading, selection of credible sources, application of appropriate referencing conventions.



+ Good use of citations. + formatted reference list.
- 1/2 books and/or academic papers could be included in the reference list for additional marks.

18**2. KNOWLEDGE AND UNDERSTANDING**

Extent of knowledge and understanding of concepts and underlying principles associated with the discipline.



Key functions: + Good knowledge of key OS functions documented. – A minor enhancement would be the consideration of process scheduling as a key function. Bat Commands: + Very good use of relevant commands with accompanying narration that demonstrates a sound understanding of the commands. Bash Commands: + Very good use of relevant commands with accompanying narration that demonstrates a sound understanding of the commands used. Bat Scripts: + Some use and understanding of relevant scripts documented. - could be extended with accompanying narration to demonstrate additional knowledge of the core syntax of the language. Bash Scripts: + Some use

13

and understanding of relevant scripts documented. - could be extended with accompanying narration to demonstrate additional knowledge of the core syntax of the language.

3. ANALYSIS

Analysis, evaluation and synthesis; logic, argument and judgement; analytical reflection; organisation of ideas and evidence.



Evidence of some good analysis and understanding of OS functions and the relevant languages is clearly documented. - could be extended in some areas as mentioned above. + Conclusion.

6

4. PRACTICAL APPLICATION

Deployment of methods, materials, tools and techniques; application of concepts; formulation of innovative and creative solutions to solve problems.



1. Bat Commands: + Excellent use of relevant commands. 2. Bash Commands: + Excellent use of relevant commands. 3. Bat Scripts: + Some good use of relevant scripts, - could be extended to demonstrate some additional knowledge of the core syntax (see relevant Moodle slides). 4. Bash Scripts: + Some good use of relevant scripts, - could be extended to demonstrate some additional knowledge of the core syntax (see relevant Moodle slides).

25.5

5. PROFESSIONAL PRACTICE

Attributes in professional practice: individual and collaborative working; deployment of appropriate media; presentation and organisation.



+ Good report structure, and presentation. - Report sections could be numbered and briefly introduced. - Appendix: Activity 1 answer could be included.

6.5