American International University- Bangladesh (AIUB)

Course Name:	Operating System	**Course Code:**	CSC 3214
Semester:	Fall 2024-25	**Submission date:**	11/24/2024
Item:	Mid All Command List		
Section	I		

Group Information

No	Name	ID
1	A.B.M Mahmudul Hasan	22-47108-1
2	MD. Arafat Alam Leon	22-47118-1
3	MD. Mehedi Hasan Polas	22-46566-1
4	Tridib Sarkar	22-46444-1

Linux Commands Table

Table 1: LAB-1

Command	Explanation	Example
cd	Displays the current directory path.	pwd
uname -a	Displays complete operating system information.	uname -a
mkdir <dir_name>	Creates a new directory with the specified name.	mkdir my_directory
cd <dir_name>	Navigates to the specified directory.	cd my_directory
ls	Lists files in the directory.	ls
ls -a	Lists all files, including hidden ones .	ls -a
touch <file_name>	Creates a new, empty file with the specified name.	touch myfile.txt
echo "content" > <file_name>	Adds content to a file, overwriting any existing content.	echo "Hello World" > myfile.txt
echo "content" >> <file_name>	Appends content to a file without overwriting existing content.	echo "More content" >> myfile.txt
cat <file_name>	Displays the content of a file.	cat myfile.txt
touch file1 file2 file3	Creates multiple new files.	touch file1.txt file2.txt file3.txt
ls *.ext	Lists files with a specific extension.	ls *.txt
mkdir -p dir1/dir2/dir3	Creates a nested directory structure.	mkdir -p parent/child/grandchild
cd ..	Moves to the parent directory.	cd ..
uname -r	Displays system kernel version information.	uname -r
uname -m	Displays system architecture.	uname -m
touch .<file_name>	Creates a hidden file (file starting with a dot).	touch .hiddenfile
command > <file_name>	Redirects the output of a command to a new file.	ls > filelist.txt
command >> <file_name>	Appends the output of a command to an existing file.	uname -a >> systemlog.txt

Table 2: LAB-2

Command	Explanation	Example
touch file1.txt	Creates an empty file named `file1.txt`.	touch file1.txt
mkdir mydir	Creates a new directory named `mydir`.	mkdir mydir
echo "Hello, World!" > output.txt	Writes 'Hello, World!' into a new file `output.txt` using redirection.	echo "Hello, World!" > output.txt
echo "Appending a new line." >> output.txt	Appends text to the existing `output.txt` file.	echo "Appending a new line." >> output.txt
cat output.txt	Displays the contents of the file `output.txt`.	cat output.txt
pwd	Prints the current working directory path.	pwd
echo "This is file2 content." > file2.txt	Writes content to a file `file2.txt`.	echo "This is file2 content." > file2.txt
cat output.txt file2.txt > combined.txt	Combines the contents of `output.txt` and `file2.txt` into `combined.txt`.	cat output.txt file2.txt > combined.txt
less example.txt	Views the contents of `example.txt` one screen at a time.	less example.txt
cat file2.txt >> output.txt	Appends the contents of `file2.txt` to `output.txt`.	cat file2.txt >> output.txt
nl jtp.txt	Displays `jtp.txt` with line numbers.	nl jtp.txt
touch .hiddenfile	Creates a hidden file named `.hiddenfile`.	touch .hiddenfile
sort output.txt	Displays the sorted contents of `output.txt`.	sort output.txt
cp file1.txt file1_backup.txt	Copies `file1.txt` to `file1_backup.txt`.	cp file1.txt file1_backup.txt
cp output.txt file2.txt mydir	Copies `output.txt` and `file2.txt` into the `mydir` directory.	cp output.txt file2.txt mydir
mv file2.txt mydir/	Moves `file2.txt` to the `mydir` directory.	mv file2.txt mydir/
rm file1_backup.txt	Deletes the file `file1_backup.txt`.	rm file1_backup.txt

Serial No.	Command	Explanation	Example
1	pwd	Displays the current working directory.	pwd
2	reset	Resets the terminal screen to default settings.	reset
3	mkdir dir1	Creates a new directory named dir1.	mkdir dir1
4	cd dir1	Changes the current directory to dir1.	cd dir1
5	cd	Changes to the home directory.	cd
6	mkdir dir2	Creates a new directory named dir2.	mkdir dir2
7	uname -a	Displays system information, including kernel version and architecture.	uname -a
8	touch ab.txt	Creates a new empty file named ab.txt.	touch ab.txt
9	echo "text" > file	Writes "text" to file, overwriting any existing content.	echo "hello linux" > ab.txt
10	echo "text" >> file	Appends "text" to file.	echo "hello append" >> ab.txt
11	cat file	Displays the content of file.	cat ab.txt
12	ls *.txt	Lists all .txt files in the current directory.	ls *.txt
13	mkdir -p path	Creates nested directories specified by path.	mkdir -p ab/bc/cd/de
14	mv source dest	Moves or renames a file or directory.	mv dir1 dir2
15	ls -a	Lists all files, including hidden ones (starting with .).	ls -a
16	ls -la	Lists all files with detailed information, including permissions, size, and modification date.	ls -la
17	touch .hidden	Creates a hidden file named .hidden.	touch .hidden
18	ls -a > file	Saves the list of all files, including hidden ones, to file.	ls -a > output.txt
19	uname -m	Displays the machine hardware name (architecture).	uname -m
20	uname -r	Displays the kernel version.	uname -r

Serial No.	Command	Explanation	Example
21	ls -R	Lists files in the current directory and all subdirectories recursively.	ls -R
22	cp source dest	Copies a file from source to dest.	cp ab.txt dir2
23	cp -r source dest	Copies directories and their contents recursively.	cp -r dir2/* newdir
24	mkdir dir5	Creates a new directory named dir5.	mkdir dir5
25	rm -r dir	Deletes a directory and its contents recursively.	rm -r dir5
26	ls -d */	Lists directories only in the current directory.	ls -d */
27	rmdir dir	Removes an empty directory.	rmdir dir5
28	clear	Clears the terminal screen.	clear

29	ls -a	Lists all files and directories in the current directory, including hidden ones.	ls -a
30	touch new.txt	Creates an empty file named new.txt.	touch new.txt
31	echo hello > new.txt	Writes the text "hello" into the file new.txt, overwriting existing content.	echo hello > new.txt
32	echo hello2 >> new.txt	Appends the text "hello2" to new.txt without overwriting the existing content.	echo hello2 >> new.txt
33	cat new.txt	Displays the content of the file new.txt on the terminal.	cat new.txt
34	mkdir -p a/b/c/d	Creates nested directories a/b/c/d in one command; -p ensures parent directories exist.	mkdir -p a/b/c/d
35	mv dir2/* dir0	Moves all contents from dir2 to dir0.	mv dir2/* dir0
36	cp file1 file2	Copies file1 to file2, creating a duplicate of file1.	cp file1.txt file2.txt
37	rm file.txt	Deletes the file file.txt.	rm file.txt
38	rmdir dir1	Removes the directory dir1 if it is empty.	rmdir emptydir
39	wc -l file.txt	Counts the number of lines in file.txt.	wc -l file.txt
40	cat > file.txt	Allows you to input text into file.txt. Press Ctrl+D to save and exit.	cat > file.txt

41	sort file.txt	Sorts the lines in file.txt alphabetically and displays them.	sort file.txt
42	grep "pattern" file	Searches for the specified pattern in the given file and displays matching lines.	grep "error" log.txt
43	chmod 777 file.txt	Changes the permissions of file.txt to read, write, and execute for all users.	chmod 777 script.sh
44	head -n 5 file.txt	Displays the first 5 lines of file.txt.	head -n 5 data.txt
45	tail -n 5 file.txt	Displays the last 5 lines of file.txt.	tail -n 5 log.txt
46	cut -d, -f2 file.txt	Splits lines of file.txt by the delimiter , and displays the second field.	cut -d, -f2 data.csv
47	paste file1 file2	Merges the content of file1 and file2 line by line.	paste file1.txt file2.txt
48	echo "text" >> file	Appends "text" to the specified file.	echo "hello world" >> notes.txt
49	sort -u file.txt	Sorts the lines in file.txt and removes duplicates.	sort -u names.txt
50	nl file.txt	Numbers the lines of file.txt and displays them.	nl file.txt
29	ls -a	Lists all files and directories in the current directory, including hidden ones.	ls -a
30	touch new.txt	Creates an empty file named new.txt.	touch new.txt
31	echo hello > new.txt	Writes the text "hello" into the file new.txt, overwriting existing content.	echo hello > new.txt
32	echo hello2 >> new.txt	Appends the text "hello2" to new.txt without overwriting the existing content.	echo hello2 >> new.txt
33	cat new.txt	Displays the content of the file new.txt on the terminal.	cat new.txt
34	mkdir -p a/b/c/d	Creates nested directories a/b/c/d in one command; -p ensures parent directories exist.	mkdir -p a/b/c/d
35	mv dir2/* dir0	Moves all contents from dir2 to dir0.	mv dir2/* dir0
36	cp file1 file2	Copies file1 to file2, creating a duplicate of file1.	cp file1.txt file2.txt
37	rm file.txt	Deletes the file file.txt.	rm file.txt
38	rmdir dir1	Removes the directory dir1 if it is empty.	rmdir emptydir
39	wc -l file.txt	Counts the number of lines in file.txt.	wc -l file.txt
40	cat > file.txt	Allows you to input text into file.txt. Press Ctrl+D to save and exit.	cat > file.txt
41	sort file.txt	Sorts the lines in file.txt alphabetically and displays them.	sort file.txt

42	grep "pattern" file	Searches for the specified pattern in the given file and displays matching lines.	grep "error" log.txt
43	chmod 777 file.txt	Changes the permissions of file.txt to read, write, and execute for all users.	chmod 777 script.sh
44	head -n 5 file.txt	Displays the first 5 lines of file.txt.	head -n 5 data.txt
45	tail -n 5 file.txt	Displays the last 5 lines of file.txt.	tail -n 5 log.txt
46	cut -d, -f2 file.txt	Splits lines of file.txt by the delimiter , and displays the second field.	cut -d, -f2 data.csv
47	paste file1 file2	Merges the content of file1 and file2 line by line.	paste file1.txt file2.txt
48	echo "text" >> file	Appends "text" to the specified file.	echo "hello world" >> notes.txt
49	sort -u file.txt	Sorts the lines in file.txt and removes duplicates.	sort -u names.txt
50	nl file.txt	Numbers the lines of file.txt and displays them.	nl file.txt
51	grep -i keyword filename	If we want to search a word like AIUB and ignore case distinctions than run the command	grep -i AIUB file1.txt
52	grep -v keyword file	This command displays only those lines without the word AIUB from the file1.txt	grep -v AIUB file1.txt
53	Grep -o pattern file1 | wc -l	Instead of displaying the entire line, the **-o** option prints only the pattern occurrences, which the wc command count.	grep -o apple file1.txt
54	grep -i keyword file	If we want to search for a word like AIUB and ignore case distinctions.	grep -i AIUB file1.txt