**Concepts of Operating System**

**Assignment 2**

**Part A**

**What will the following commands do?**

** echo "Hello, World!"**

* The echo command is a simple but useful tool to print text to the terminal or command line. It’s often used for displaying messages, debugging, or scripting.

** name="Productive"**

* The command name="Productive" is used to set a variable named name with the value

** touch file.txt**

* If file.txt does not already exist, running touch file.txt will create an empty file named file.txt.

** ls -a**

* list directory contents, including hidden files and directories

** rm file.txt**

* remove file(delete)

** cp file1.txt file2.txt**

* Copies the contents of file1.txt to file2.txt, creating or overwriting file2.txt as needed

** mv file.txt /path/to/directory/**

* This is used to move or rename files and directories

** chmod 755 script.sh**

* Changes the permissions of script.sh so that:
* The owner can read, write, and execute the file.
* The group and others can read and execute the file.

** grep "pattern" file.txt**

* **grep:** global regular expression print
* This is used to search for a specific pattern within a file in

** kill PID**

* The command kill PID is used to send a signal to a process identified by its Process ID (PID)
* kill 1234

** mkdir mydir && cd mydir && touch file.txt && echo "Hello, World!" > file.txt && cat file.txt**

* The command mkdir mydir && cd mydir && touch file.txt && echo "Hello, World!" > file.txt && cat file.txt is a series of commands connected by &&, which are executed sequentially.

 **mkdir mydir**: Creates a new directory named mydir.

 **cd mydir**: Changes to the mydir directory.

 **touch file.txt**: Creates an empty file named file.txt within mydir.

 **echo "Hello, World!" > file.txt**: Writes the text "Hello, World!" into file.txt, replacing any existing content.

 **cat file.txt**: Displays the content of file.txt, which will show "Hello, World!".

This series of commands is used to:

1. Create a new directory and move into it.
2. Create a new file inside that directory.
3. Write a specific string to the file.
4. Display the content of the file to verify the operation.

** ls -l | grep ".txt"**

* this used to list detailed information about files in the current directory and then filter that list to show only files with a .txt extension

** cat file1.txt file2.txt | sort | uniq**

* this used to process and filter text from multiple files

 **cat file1.txt file2.txt**: Combines the contents of file1.txt and file2.txt.

 **| sort**: Sorts the combined content in ascending order.

 **| uniq**: Removes duplicate lines from the sorted content.

** ls -l | grep "^d"**

* this used to filter and display information about directories from a list of files and directories in a directory

 **ls -l**: Lists detailed information about files and directories.

 **| grep "^d"**: Filters the list to show only directories, as directories are indicated by lines starting with the character d

** grep -r "pattern" /path/to/directory/**

* this used to search for a specific pattern recursively through files within a specified directory and its subdirectories

 **Searches Recursively**: grep -r "pattern" /path/to/directory/ will look for occurrences of "pattern" in every file within /path/to/directory/ and its subdirectories.

 **Displays Matching Lines**: The command will output lines from the files that match the specified pattern, along with the filenames and line numbers where the matches are found

** cat file1.txt file2.txt | sort | uniq –d**

* this used to process and filter text data from multiple files, specifically to identify and display duplicate lines across those files

 **cat file1.txt file2.txt | sort | uniq -d**: Combines the contents of file1.txt and file2.txt, sorts them, and then filters out and displays only the lines that appear more than once.

 **Purpose**: Useful for identifying duplicate lines that are present across multiple files, helping in data analysis, cleanup, or verification tasks

** chmod 644 file.txt**

* this used to set specific file permissions for the file named file.txt

 **Owner (User)**: Read and write (rw-)

 **Group**: Read-only (r--)

 **Others**: Read-only (r--)

** cp -r source\_directory destination\_directory**

* this used to copy a directory and its contents from one location to another

 **cp -r source\_directory destination\_directory**: Copies source\_directory and all its contents (including subdirectories and files) to destination\_directory.

 **Usage**: Useful for backing up or moving entire directory structures while preserving the hierarchy and contents

** find /path/to/search -name "\*.txt"**

* this used to search for files with a specific name pattern within a specified directory and its subdirectories

 **find /path/to/search -name "\*.txt"**: Searches for files with a .txt extension in the specified directory and all its subdirectories.

 **Usage**: Useful for locating all text files within a given directory hierarchy, especially when dealing with large directory structures or needing to find files by their extension

** chmod u+x file.txt**

* this used to modify the permissions of a file
* **chmod u+x file.txt**: Adds execute permissions for the owner (user) of file.txt.

** echo $PATH \* th**is used to display the value of the PATH environment variable in Unix-like operating systems (such as Linux and macOS).

* **echo $PATH**: Prints the current value of the PATH environment variable.

**----------------------------------------------------------------------------------------------------------------------**

**Part B**

**Identify True or False:**

1. **ls is used to list files and directories in a directory.**

True

1. **mv is used to move files and directories.**

True

1. **cd is used to copy files and directories.**

**False**

The cd command is not used to copy files and directories. Instead, cd stands for "change directory"

1. **pwd stands for "print working directory" and displays the current directory.**

True

1. **grep is used to search for patterns in files.**

True

1. **chmod 755 file.txt gives read, write, and execute permissions to the owner, and read and execute permissions to group and others.**

True

**7. mkdir -p directory1/directory2 creates nested directories, creating directory2 inside directory1**

**if directory1 does not exist.**

True

1. **rm -rf file.txt deletes a file forcefully without confirmation.**

True

**Identify the Incorrect Commands:**

1. **chmodx is used to change file permissions.**

 **chmodx**: Incorrect; should be chmod.

1. **cpy is used to copy files and directories.**

 **cpy**: Incorrect; should be cp.

1. **mkfile is used to create a new file.**

 **mkfile**: Incorrect for general file creation; touch is more commonly used.

1. **catx is used to concatenate files.**

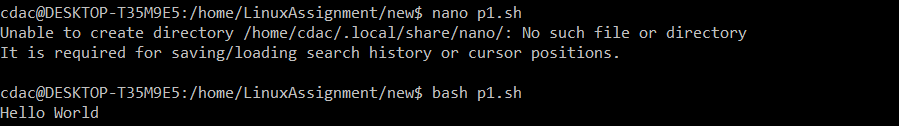
**catx**: Incorrect; should be cat.

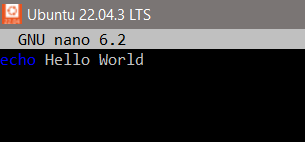
1. **rn is used to rename files.**

 **rn**: Incorrect; should be mv.

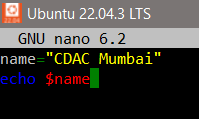
**Part C**

**Question 1: Write a shell script that prints "Hello, World!" to the terminal.**



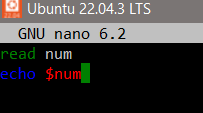


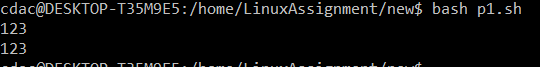
**Question 2: Declare a variable named "name" and assign the value "CDAC Mumbai" to it. Print the value of the variable.**



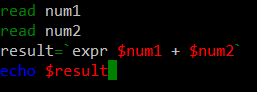


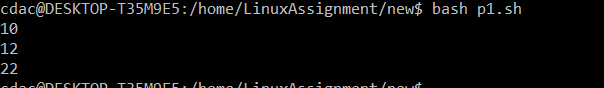
**Question 3: Write a shell script that takes a number as input from the user and prints it.**



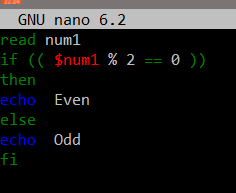


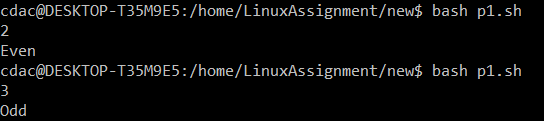
**Question 4: Write a shell script that performs addition of two numbers (e.g., 5 and 3) and prints the result.**



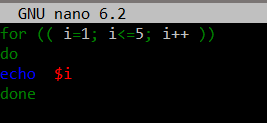


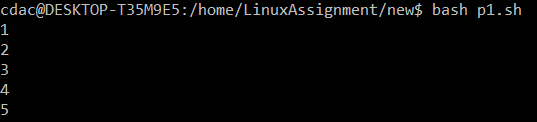
**Question 5: Write a shell script that takes a number as input and prints "Even" if it is even, otherwise prints "Odd".**



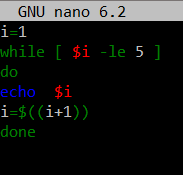


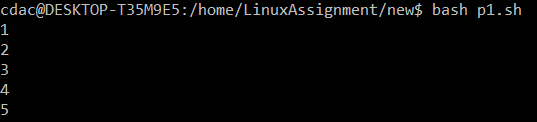
**Question 6: Write a shell script that uses a for loop to print numbers from 1 to 5.**



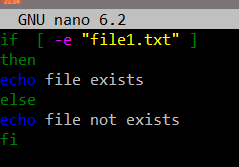


**Question 7: Write a shell script that uses a while loop to print numbers from 1 to 5.**



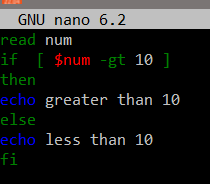


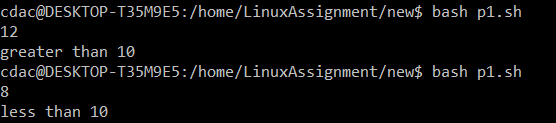
**Question 8: Write a shell script that checks if a file named "file.txt" exists in the current directory. If it does, print "File exists", otherwise, print "File does not exist".**





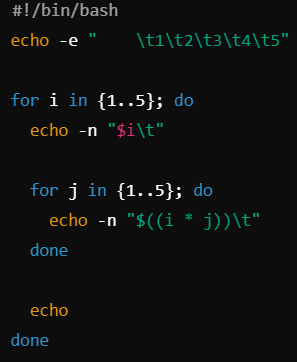
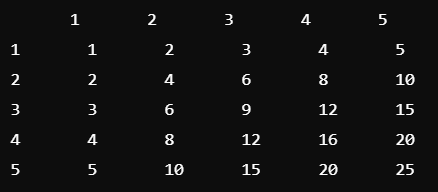
**Question 9: Write a shell script that uses the if statement to check if a number is greater than 10 and prints a message accordingly.**

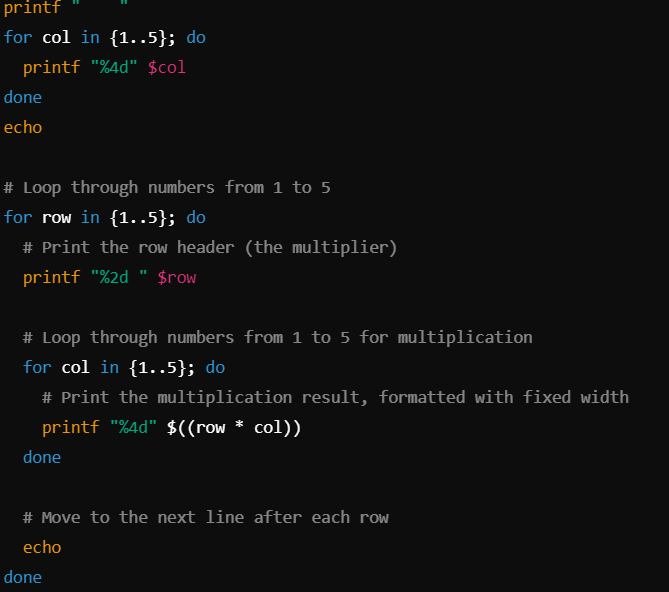




**Question 10: Write a shell script that uses nested for loops to print a multiplication table for numbers from 1 to 5. The output should be formatted nicely, with each row representing a number and each**

**column representing the multiplication result for that number.**



**Question 11: Write a shell script that uses a while loop to read numbers from the user until the user enters a negative number. For each positive number entered, print its square. Use the break statement to exit the**

**loop when a negative number is entered.** 