Concepts of Operating System

Assignment 2

Part A

What will the following commands do?

echo "Hello, World!"

**Ans- It will print Hello World on console**

name="Productive"

Ans- Here variable ‘name’ is assigned the string “Productive”

touch file.txt

Ans- It will create a file named ‘file.txt’

ls -a

Ans- lists all files and directories in the current directory even the hidden files.

rm file.txt

Ans –It removes the file from a directory

cp file1.txt file2.txt

Ans- this command copies file1.txt to file2.txt

mv file.txt /path/to/directory/

Ans- here mv enables file.txt to move to different directory

chmod 755 script.sh

Ans- here chmod command changes the permissions of script.sh to ‘755’ which means owner can read,write and execute and others can read and execute

grep "pattern" file.txt

Ans- grep command searches for the string “pattern” inside file.txt and displays it

kill PID

Ans- kill command kills or teminate the process with the ID ie. PID

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

Ans- First mkdir creates a directory mydir . Then using cd command it goes into mydir.Then file.txt is created using touch command in mydir. Then Hello World is written into file.txt using echo command which uses redirection symbol. Then cat file.txt displays the “Hello World” message

ls -l | grep ".txt"

Ans – lists all files in long format and then using piping symbol grep searches .txt files

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

Ans- this command concatenates file.txt and file2.txt . then sorts their content and then removes their duplicates

 ls -l | grep "^d"

Ans- list files in long format and filters the list to show only directories starting with d

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

Ans - Recursively searches for "pattern" in all files within the specified directory and its subdirectories

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

Ans - Concatenates file1.txt and file2.txt, sorts the content, and displays only duplicate lines.

 chmod 644 file.txt

Ans - Changes the permissions of file.txt to 644, which usually means the owner can read and write, and others can only read.

 cp -r source\_directory destination\_directory

Ans - Recursively copies the contents of source\_directory to destination\_directory.

 find /path/to/search -name "\*.txt"

Ans- Searches for all .txt files within the specified directory and its subdirectories.

 chmod u+x file.txt

Ans - Adds execute permission for the user on file.txt.

 echo $PATH

Ans - Prints the current value of the PATH environment variable, which lists directories where the shell looks for executable files.

**Part B**

**Identify True or False:**

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

Ans. **True**

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

Ans. **True**

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

Ans. **False,** because cd is use to change the directory and cp is use to copy the files.

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

Ans. **True ,** it stands for print working directory.

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

Ans. **True**

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

Ans . **True**

1. **mkdir -p directory1/directory2** creates nested directories, creating directory2 inside directory1 if directory1 does not exist.

Ans . **True**

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

Ans. **True**

**Identify the Incorrect Commands:**

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

**Incorrect. (chmod)**

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

**Incorrect. (cp)**

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

**Incorrect. (touchfilename.txt)**

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

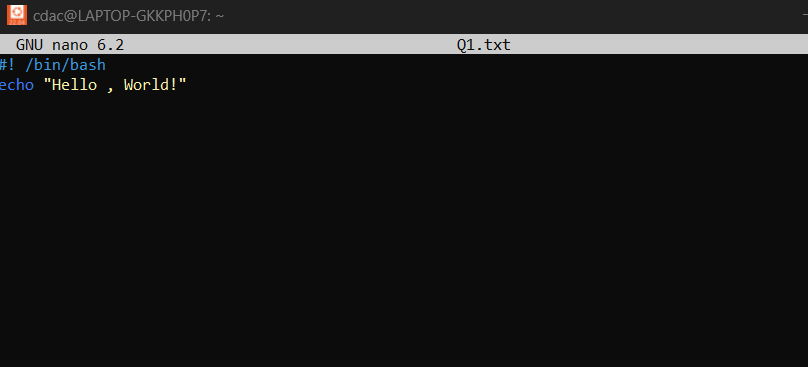
**Incorrect. (cat file1.txt file2.txt > newfile.txt)**

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

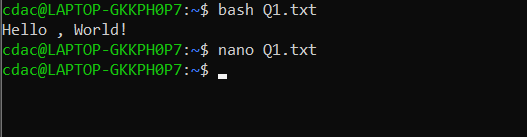
**Incorrect. (cat file1.txt file2.txt > newfile.txt)**

**Part C**

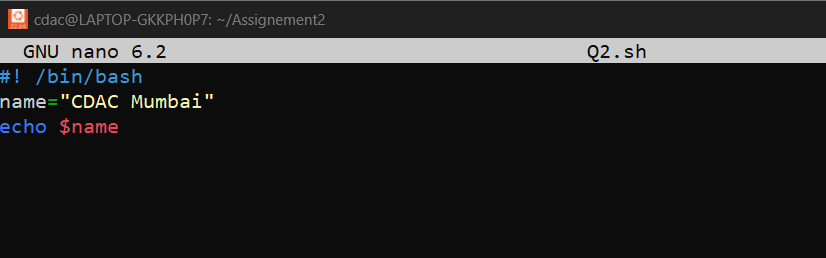
**Question 1: Write a shell script that prints "Hello, World!" to the terminal.**

****

**Output**

****

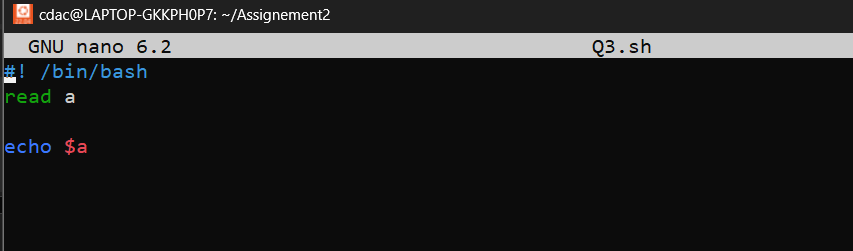
**Question 2: Declare a variable named "name" and assign the value "CDAC Mumbai" to it. Print the value of the variable.**

****

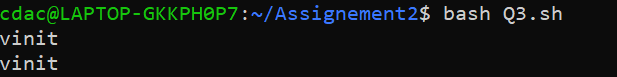
**Output-**

****

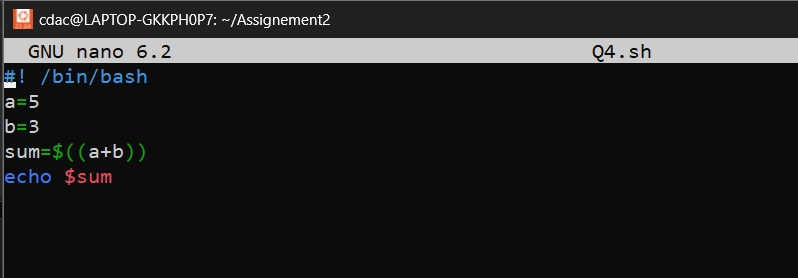
**Question 3: Write a shell script that takes a number as input from the user and prints it.**

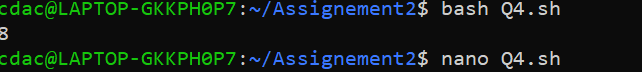
****

**Output-**

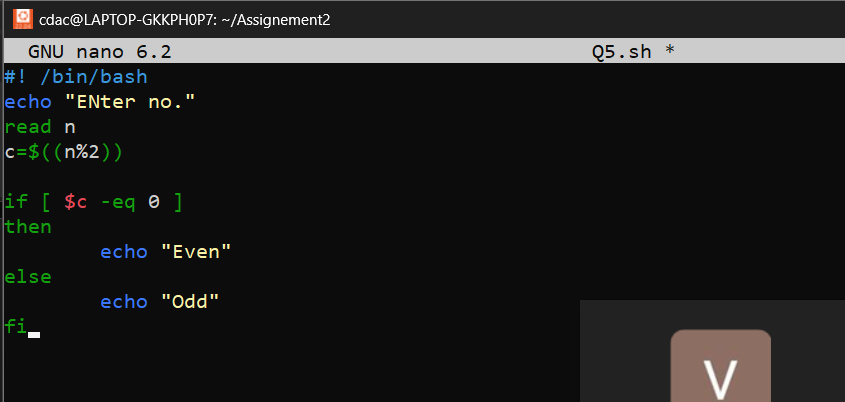
****

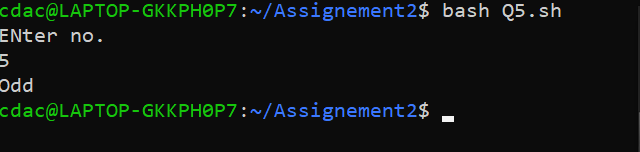
**Question 4: Write a shell script that performs addition of two numbers (e.g., 5 and 3) and prints the result.**

****

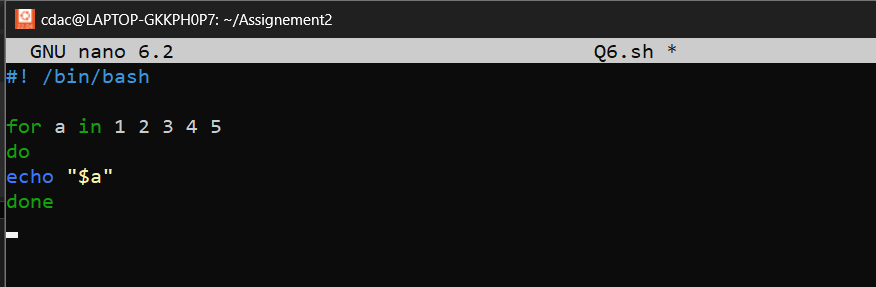
****

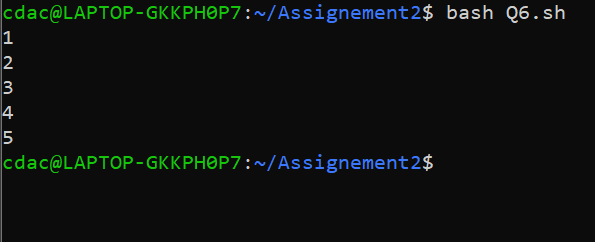
**Question 5: Write a shell script that takes a number as input and prints "Even" if it is even, otherwise prints "Odd".**

****

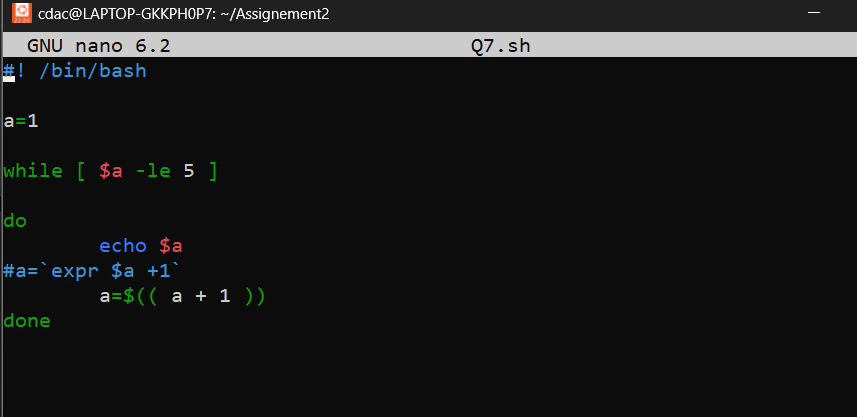
****

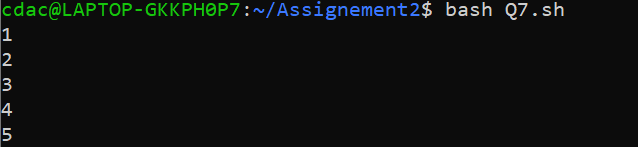
**Question 6: Write a shell script that uses a for loop to print numbers from 1 to 5.**

****

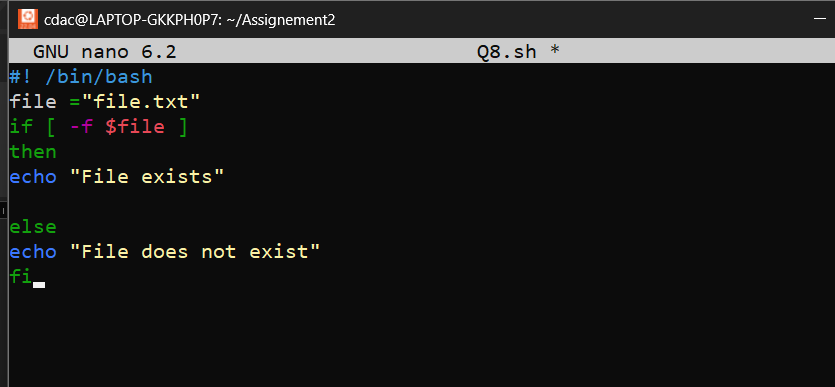
****

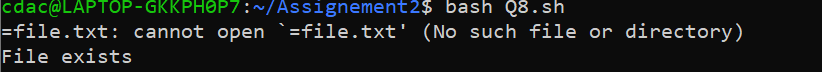
**Question 7: Write a shell script that uses a while loop to print numbers from 1 to 5.**

****

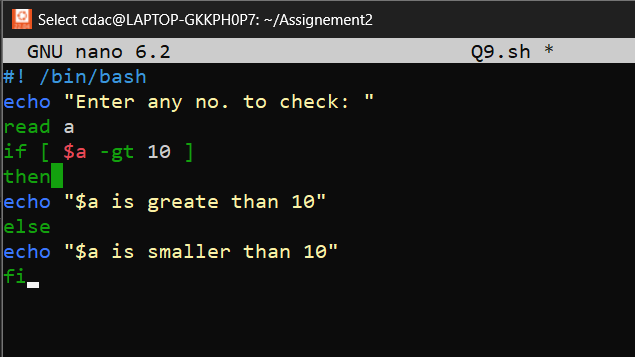
****

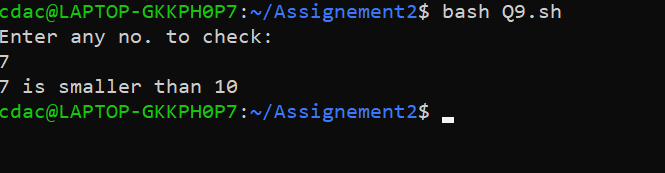
**Question 8: Write a shell script that checks if a file named "file.txt" exists in the current directory. If it does, print "File exists", otherwise, print "File does not exist".**

****

****

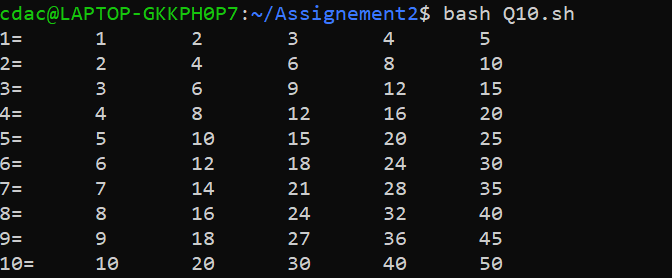
**Question 9: Write a shell script that uses the if statement to check if a number is greater than 10 and prints a message accordingly.**

****

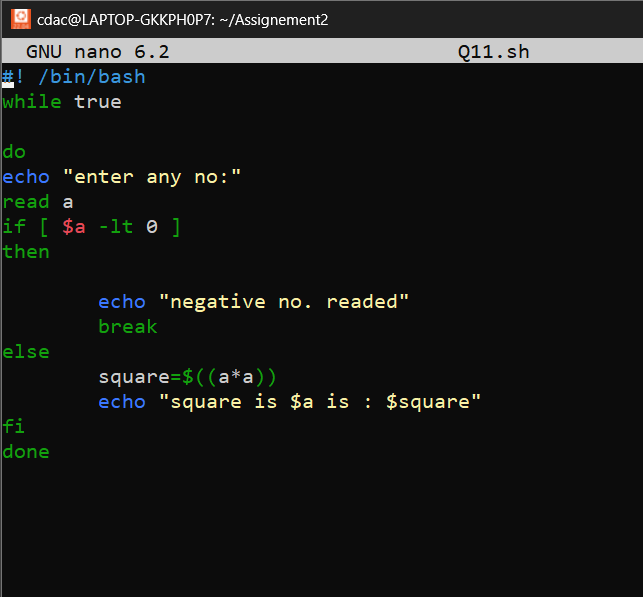
****

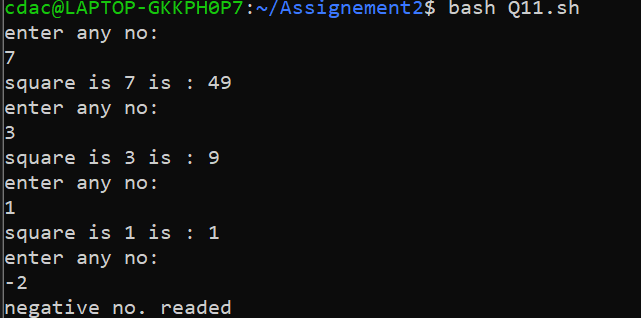
**Question 10: Write a shell script that uses nested for loops to print a multiplication table for numbers from 1 to 5. The output should be formatted nicely, with each row representing a number and each column representing the multiplication result for that number.**

****

****

**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.**

****

****

PART E

