CDAC MUMBAI

**Concepts of Operating System Assignment 2**

# Part A

## What will the following commands do?

1. **echo "Hello, World!"** : print the Hello, World!
2. **name="Productive"** : assign the value “Productive” to name and we can access it by echo $name
3. **touch file.txt** : create an empty file name file.txt or updates the timestamp of the file if already exists.
4. **ls -a** : lists all files and directories int the current directory including hidden one
5. **rm file.txt** : remove/ deletes the file name file.txt
6. **cp file1.txt file2.txt** : copy the content of file1.txt to file2.txt
7. **mv file.txt /path/to/directory/** : Move the file name file.txt to respective path.
8. **chmod 755 script.sh** : Changes the permissions of script.sh to rwxr-xr-x
9. **grep "pattern" file.txt** : searches for the specified ‘patterb’ in ‘file.txt’ and prints lines that contain it.
10. **kill PID** : Sends a termination signal to the process with specified process ID
11. **mkdir mydir && cd mydir && touch file.txt && echo "Hello, World!" > file.txt && cat file.txt** :
    1. create directory named mydir changes into that directory, creats an empty file named file.txt write Hello World to file,txt and then displays the contents of file.txt.
12. **ls -l | grep ".txt"** : list files in long format and filters the output to show only lines containing .txt which is typically list files with .txt extension
13. **cat file1.txt file2.txt | sort | uniq** : concatenates the contents of file1.txt and file2.txt sort the combined output and remove duplicate lines.
14. **ls -l | grep "^d"** : Lists files in long formant and filters the output to show only directories
15. **grep -r "pattern" /path/to/directory**/ : searches for specified pattern in all files within the specified directory.
16. **cat file1.txt file2.txt | sort | uniq –d** :concatenates the contents of file.txt and file2.txt sort the combined output and shows only duplicate lines.
17. **chmod 644 file.txt** : changes the permissions of file.txt to rw-r—r—giving the owner read and write permissions and others read only permissions
18. **cp -r source\_directory destination\_directory**: copy content from one file to other.
19. **find /path/to/search -name "\*.txt"** : searches for files with th .txt for given path
20. **chmod u+x file.txt** : add execute permissins for the owner of file.txt allowing owner to execute the file.
21. **echo $PATH** : print the value of the path variable

# Part B

## Identify True or False:

1. **ls** is used to list files and directories in a directory. : **True**
2. **mv** is used to move files and directories. :**True**
3. **cd** is used to copy files and directories. :**False**
4. **pwd** stands for "print working directory" and displays the current directory.: **True**
5. **grep** is used to search for patterns in files.: **Ture**
6. **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**
8. **rm -rf file.txt** deletes a file forcefully without confirmation.**True**

# Part C

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

**Answer:** echo "Hello, World!"

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

**Answer:** name="CDAC Mumbai"

echo "$name"

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

**Answer:** echo Enter Number

read num

echo "$num"

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

**Answer** echo Enter Number1

read num1

echo Enter Number2

read num2

num3=`expr $num1 + $num2`

echo "$num3"

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

**Answer:** echo “Enter Number”

read number

if [ $((number % 2)) -eq 0 ];

then

echo "Even"

else

echo "Odd"

fi

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

**Answer:** for i in {1..5}

do

echo “$i”

done

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

**Answer:** i=1

while [ $i -le 5 ]

do

echo $i

i=`expr $i + 1`

done

**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".

**Answer:** if [ -f "file.txt" ];

then

echo "File exists"

else

echo "File does not exist"

fi

**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.

**Answer:** echo Enter the num

read num

if [ $num -gt 10 ]

then

echo "Greater than 10"

else

echo "Not Greater than 10"

fi

**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.

**Answer:** 

**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.

**Answer:**

while true;

do

read -p "Enter a number (negative to stop): " number

if [ $number -lt 0 ];

then

break

else

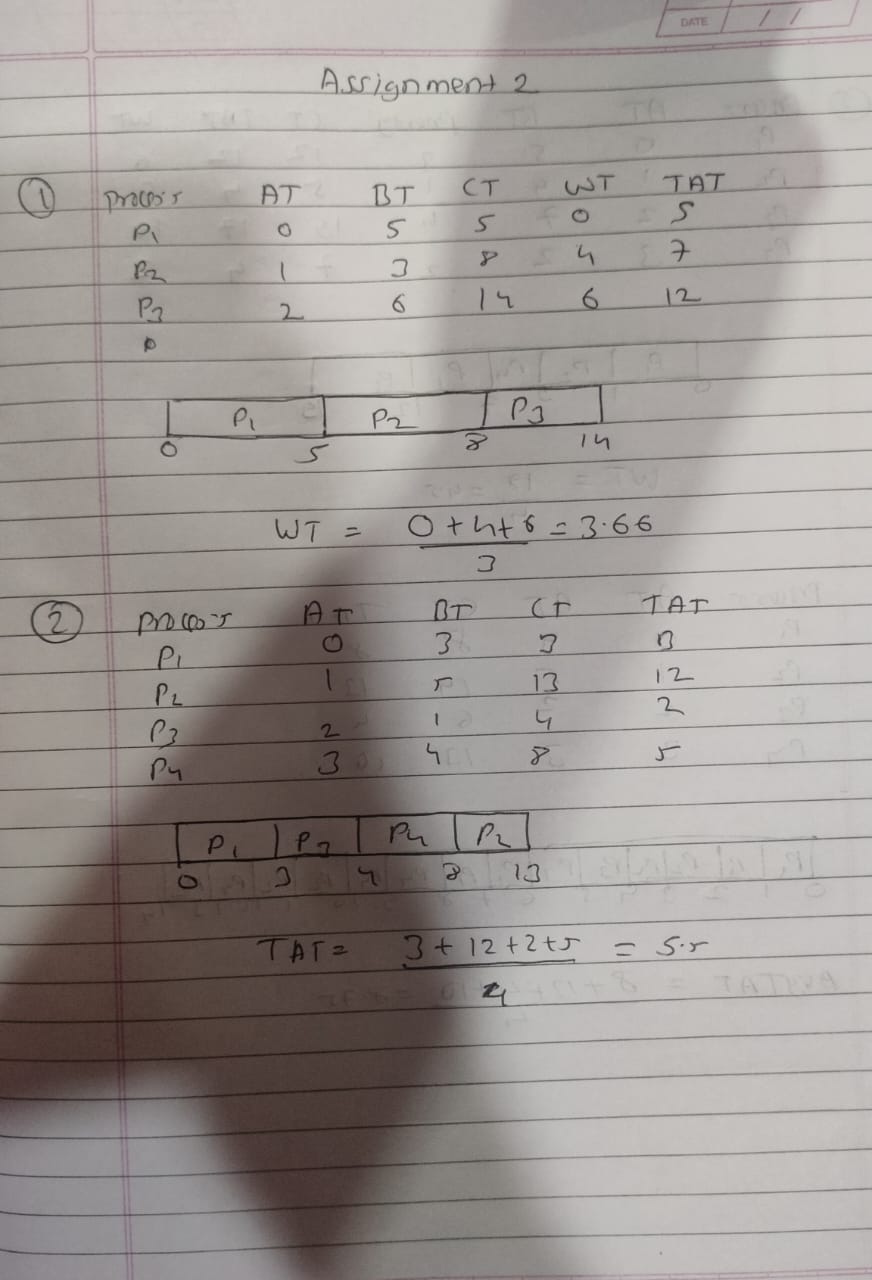
square=$((number \* number))

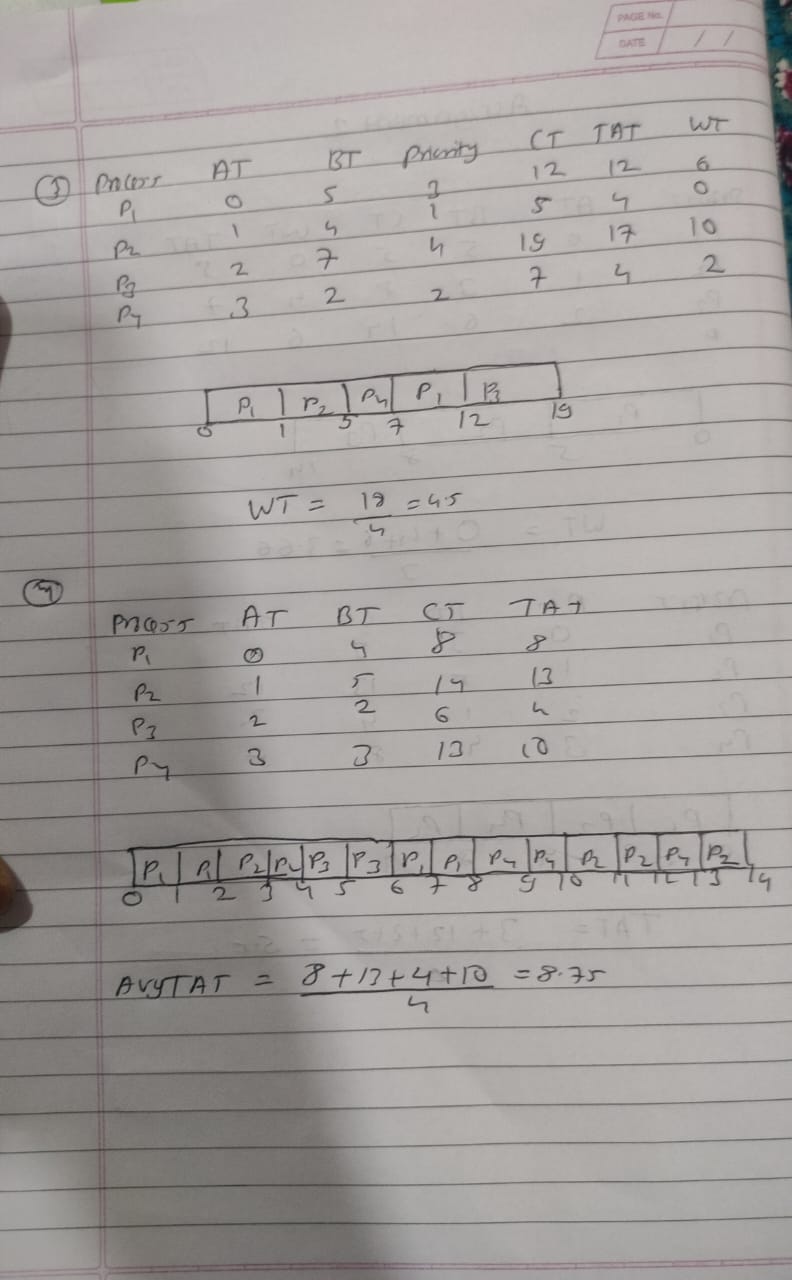
echo "The square of $number is $square"

fi

done

# Part E

****

****