CDAC MUMBAI

Concepts of Operating System

Assignment 2

Name Akshay Janrao

Part A

What will the following commands do?

1. echo "Hello, World!"
   Answer: Will display “Hello World!”
2. name="Productive"

   Answer:

3. touch file.txt
4. Answer: create a blank flie named as file.txt
5. ls -a
   Answer:

6. rm file.txt
7. Answer: delete the file.txt

8. cp file1.txt file2.txt
9. Answer: will copy the contents from file1.txt to file2.txt

10. mv file.txt /path/to/directory/
11. Answer: will move the file1.txt to directory

12. chmod 755 script.sh
13. Answer: will give the owner of file script.sh all the permission and group ,others the read and execute permission

14. grep "pattern" file.txt
15. Answer: will fetch all the files with the specified “pattern”

16. kill PID
17. Answer: will terminate the file with the given PID.

18. mkdir mydir && cd mydir && touch file.txt && echo "Hello, World!" > file.txt && cat file.txt
19. Answer: Will create a directory called “mydir” change the current directory to “mydir” creates a file “file.txt” edite the file.txt to print “Hello,World!”.

20. ls -l | grep ".txt"
21. Answer: will list all the files and through grep will print the flies ending in”.txt”.

22. cat file1.txt file2.txt | sort | uniq
23. Answer: will get the contents and display the reapted lines from file1.txtx and file2.txt

24. ls -l | grep "^d"
25. Answer: will list all the files and by piping method into grep which will filter and display the lines that begin with “d”

26. grep -r "pattern" /path/to/directory/
27. Answer: the command will search all the files within directory that contains the specified patterns

28. cat file1.txt file2.txt | sort | uniq –d
29. Answer: Combines the contents of the files and sorts the combined lines and display the duplicate lines will only allow the files that are repeated more than once

30. chmod 644 file.txt
31. Answer: will give the owner the of file.txt to have read and write permission and will give the group, others the read only permission.

32. cp -r source_directory destination_directory

Answer: Will create copies of the files from the source directory and all of its contents in destination directory.

33. find /path/to/search -name "*.txt"
    Answer: The command will create a list of all files with .txt in directories and all subdirectories.

34. chmod u+x file.txt

    Answer: this will enable the owner to have the execute permission

35. echo $PATH

Answer: the command will print the contents of the path variable

Part B

Identify True or False:

- **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.
  = TRUE
- 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.
  = FALSE
- 8. **rm -rf file.txt** deletes a file forcefully without confirmation.
  = FALSE

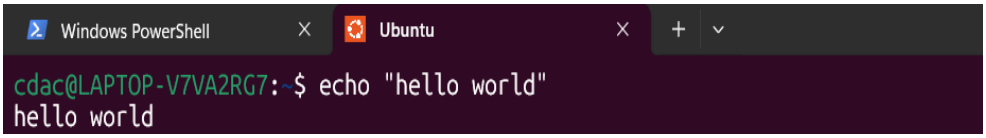
Identify the Incorrect Commands:

1. chmodx is used to change file permissions.
   =Chmod command is used to change the permission
2. cpy is used to copy files and directories.
   =“cp” is used to copy the files
3. mkfile is used to create a new file.
   =This command is used to create a flie with a specified name
4. catx is used to concatenate files.
   =The correct command is “cat”
5. rn is used to rename files.
   =The given command is correct and is used to rename the file.

Part C

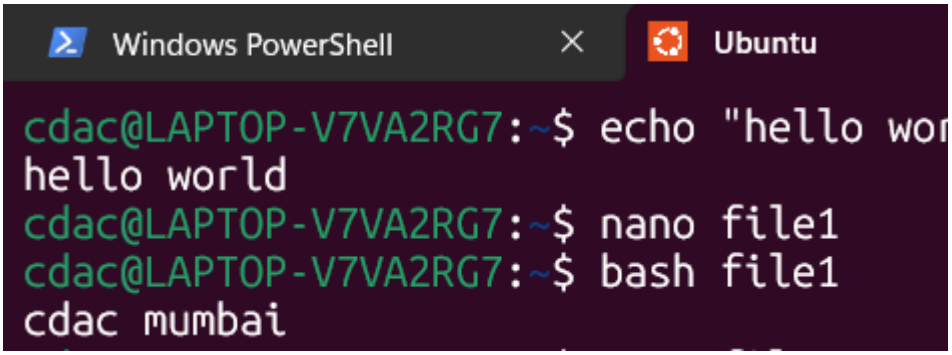
Question 1: Write a shell script that prints "Hello, World!" to the terminal.

Ans: The command used will be **echo "Hello, World!"**



Question 2: Declare a variable named "name" and assign the value "CDAC Mumbai" to it. Print the value of the variable.
Ans: The command used will be

name="CDAC Mumbai"
echo "$name"



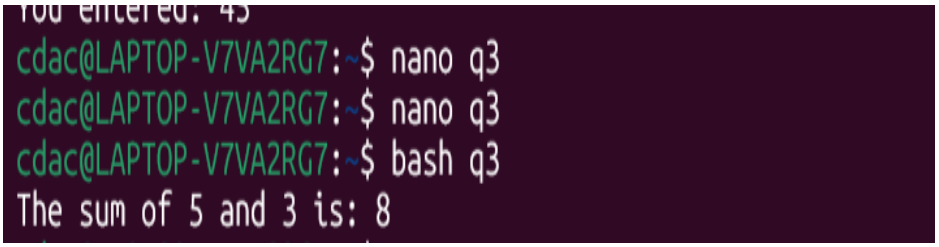
Question 3: Write a shell script that takes a number as input from the user and prints it.
Ans: The command used will be

read -p "Enter a number: " number
echo "You entered: $number"

Question 4: Write a shell script that performs addition of two numbers (e.g., 5 and 3) and prints the result.
Ans: The command used will be

num1=5
num2=3
sum=$((num1 + num2))

echo "The sum of $num1 and $num2 is: $sum"

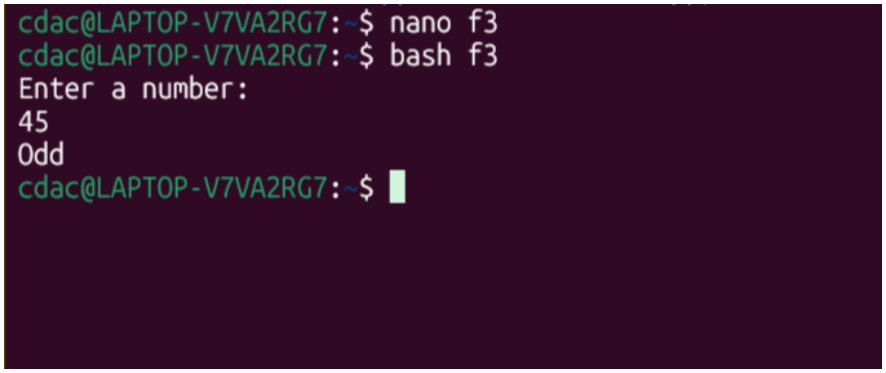


Question 5: Write a shell script that takes a number as input and prints "Even" if it is even, otherwise prints "Odd".
Ans: The command used will be

#!/bin/bash

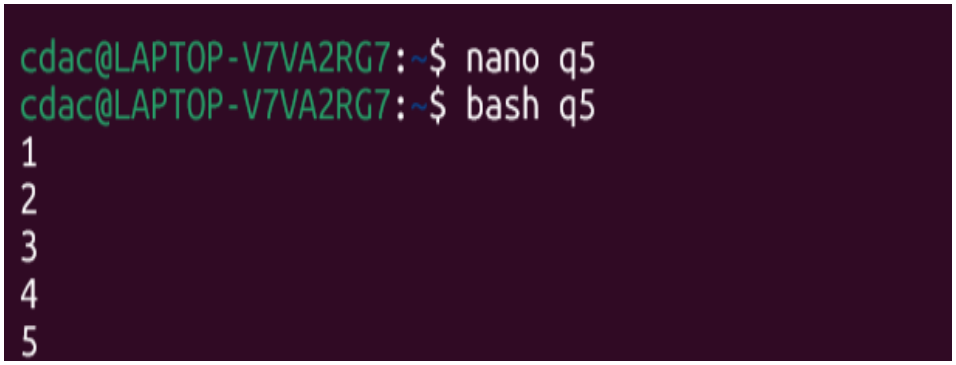
read number
if [$((number % 2)) -eq 0]; then

```
 echo "Even"
else
 echo "Odd"
fi
```



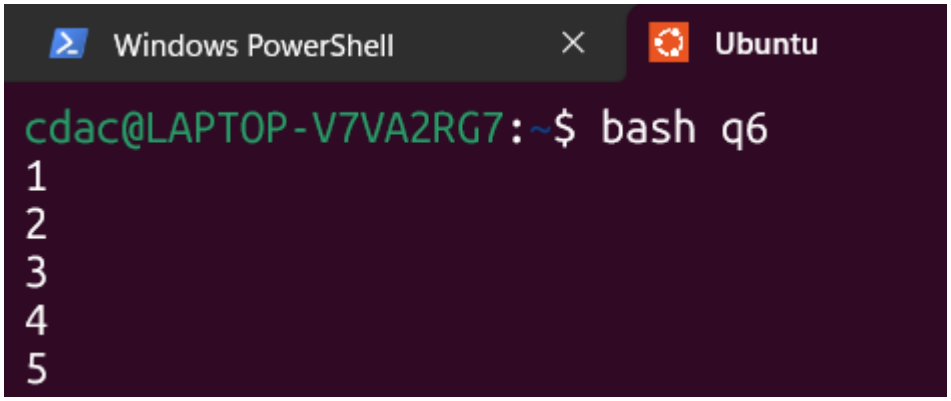
Question 6: Write a shell script that uses a for loop to print numbers from 1 to 5.
Ans: The command used will be

```
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.
Ans: The command used will be

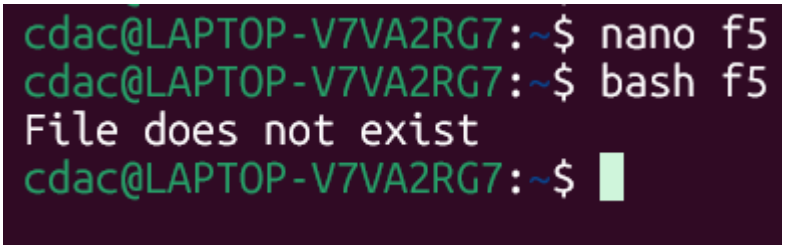
```
i=1
while [ $i -le 5 ]; do
 echo "$i"
 i=$((i + 1))
done
```



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".

Ans: The command used will be

```
if [ -f "file.txt" ]; then
  echo "File exists"
else
  echo "File does not exist"
fi
```

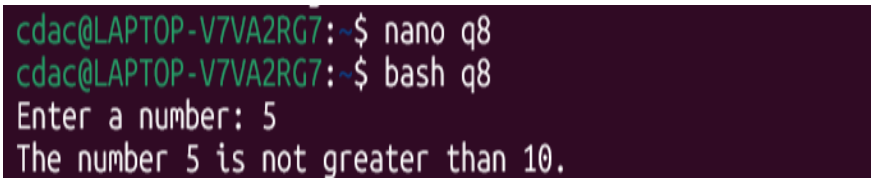


Question 9: Write a shell script that uses the if statement to check if a number is greater than 10 and prints a message accordingly.

Ans: The command used will be

```
read -p "Enter a number: " number

if (( number > 10 )); then
  echo "The number $number is greater than 10."
else
  echo "The number $number is 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.

Ans: The command used will be

```
echo -e "   1\t2\t3\t4\t5" # Header row
echo "-----------------------------------"

for i in {1..5}; do
  echo -n "$i |"
  for j in {1..5}; do
    result=$((i * j))
    echo -en "\t$result
  done
  echo ""
done
```

```
cdac@LAPTOP-V7VA2RG7:~$ nano q9
cdac@LAPTOP-V7VA2RG7:~$ bash q9
   1    2       3       4       5
-------------------------------------
1 |     1       2       3       4       5
2 |     2       4       6       8       10
3 |     3       6       9       12      15
4 |     4       8       12      16      20
5 |     5       10      15      20      25
```

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.
Ans: The command used will be

```
while true; do
 read -p "Enter a number (negative to exit): " number

 if (( number < 0 )); then
  break
 fi

 if (( number >= 0 )); then
  square=$((number * number))
  echo "Square of $number is: $square"
 fi

done

echo "Exiting..."
```

```
cdac@LAPTOP-V7VA2RG7:~$ nano q9
cdac@LAPTOP-V7VA2RG7:~$ nano q10
cdac@LAPTOP-V7VA2RG7:~$ bash q10
Enter a number (negative to exit): 54
Square of 54 is: 2916
Enter a number (negative to exit): 12
Square of 12 is: 144
Enter a number (negative to exit): -4
Exiting...
cdac@LAPTOP-V7VA2RG7:~$
```

Part E

1. Consider the following processes with arrival times and burst times:

Process	Arrival Time	Burst Time
P1	0	5
P2	1	3
P3	2	6

Calculate the average waiting time using First-Come, First-Served (FCFS) scheduling**.**

ANS

As the p1 came at "**0 sec**" it will get allocated instantly hence the waiting time for p1 is**"0".**
Therefore **w1=0**
The p2 came at "**1 sec" the task will be executed after completion of "p1"** therefore the waiting of **"p2"** will be **"w2= burst time of p1 -arrival p2" is "5-1 =4"**
Similarly waiting time for p3 "**burst time – arrival time** "is **"8 -2 =6**"
Avg waiting time **10/3 = 3.33 sec**

2. Consider the following processes with arrival times and burst times:

Process	Arrival Time	Burst Time
P1	0	3
P2	1	5
P3	2	1
P4	3	4

Calculate the average turnaround time using Shortest Job First (SJF) scheduling.

ANS

	arrival	**burst**	**wait**	**response**	**TAT**
P1	**0**	**3**	**0**	**0**	**3**
P2	**1**	**5**	**7**	**8**	**12**
P3	**2**	**1**	**1**	**3**	**2**
P4	**3**	**4**	**9**	**12**	**13**

AVG TAT 7.5

3. Consider the following processes with arrival times, burst times, and priorities (lower number indicates higher priority):

| Process | Arrival Time | Burst Time | Priority |

|---------|--------------|------------|----------|

| P1 | 0 | 6 | 3 |

| P2 | 1 | 4 | 1 |

| P3 | 2 | 7 | 4 |

| P4 | 3 | 2 | 2 |

Calculate the average waiting time using Priority Scheduling

ANS **ARRIVAL** **BURST** **PRIORITY** **WAIT**

P1	**0**	**6**	**3**	**7**
P2	**1**	**4**	**1**	**0**
P3	**2**	**7**	**4**	**11**
P4	**3**	**2**	**2**	**2**

AVG = 5 SEC

4. Consider the following processes with arrival times and burst times, and the time quantum for
Round Robin scheduling is 2 units:

Process	Arrival Time	Burst Time
P1	0	4
P2	1	5
P3	2	2
P4	3	3

Calculate the average turnaround time using Round Robin scheduling

ANS **ARRIVAL** **BURST** **WAIT** **TAT**

P1	**0**	**4**	**10**	**10**
P2	**1**	**5**	**14**	**13**
P3	**2**	**2**	**6**	**4**
P4	**3**	**3**	**12**	**9**

AVG =9 SEC

GNATT CHART

TIME	**0**	**2**	**4**	**6**	**8**	**10**	**11**	**12**	**14**
PROCESS	**P1**	**P2**	**P3**	**P4**	**P1**	**P2**	**P4**	**P2**	**P2**

6. Consider a program that uses the fork () system call to create a child process. Initially, the parent process has a variable x with a value of 5. After forking, both the parent and child processes increment the value of x by 1. What will be the final values of x in the parent and child processes after the fork () call?

   ANS

   The parent class increase by 1, the parent process becomes “X+1= “
   i.e. 5+1=6.

   Also, the child process increases by 1 hence the child class will be i.e. 5+1=6.