CDAC MUMBAI

Concepts of Operating System

Assignment 2

Part A – What will the following commands do?

Question: echo "Hello, World!"
Answer: Prints the text Hello, World! on the terminal.

Question: name="Productive"
Answer: Creates a shell variable 'name' with value Productive.

Question: touch file.txt
Answer: Creates an empty file file.txt or updates its timestamp.

Question: ls -a
Answer: Lists all files including hidden ones.

Question: rm file.txt
Answer: Deletes the file file.txt.

Question: cp file1.txt file2.txt
Answer: Copies file1.txt into file2.txt.

Question: mv file.txt /path/to/directory/
Answer: Moves file.txt to the given directory or renames it.

Question: chmod 755 script.sh
Answer: Gives rwx to owner, rx to group/others.

Question: grep "pattern" file.txt
Answer: Searches for 'pattern' in file.txt.

Question: kill PID
Answer: Terminates the process with the given PID.

Question: mkdir mydir && cd mydir && touch file.txt && echo "Hello, World!" > file.txt && cat file.txt
Answer: Creates dir, enters it, creates file.txt, writes 'Hello, World!' inside, then displays it.

Question: ls -l | grep ".txt"
Answer: Lists files in long format and shows only .txt files.

Question: cat file1.txt file2.txt | sort | uniq
Answer: Merges both files, sorts and removes duplicates.

Question: ls -l | grep "^d"
Answer: Lists only directories.

Question: grep -r "pattern" /path/to/directory/
Answer: Recursively searches for 'pattern' in the directory.

Question: cat file1.txt file2.txt | sort | uniq -d
Answer: Shows duplicate lines only

Question: chmod 644 file.txt
Answer: Owner can read/write, group and others only read.

Question: cp -r source_directory destination_directory
Answer: Recursively copies one directory to another.

Question: find /path/to/search -name "*.txt"
Answer: Finds all .txt files under the given directory.

Question: chmod u+x file.txt
Answer: Gives execute permission to the file's owner.

Question: echo $PATH
Answer: Displays the PATH environment variable.

Part B – True/False

1. ls is used to list files and directories → True
2. mv is used to move files and directories → True
3. cd is used to copy files and directories → False (cd changes directory)
4. pwd stands for print working directory → True
5. grep is used to search for patterns in files → True
6. chmod 755 file.txt gives rwx to owner, rx to group/others → True
7. mkdir -p creates nested directories → True
8. rm -rf file.txt deletes a file forcefully → True

Part B – Incorrect Commands

1. chmodx ■ → Correct: chmod
2. cpy ■ → Correct: cp
3. mkfile ■ → Correct: touch
4. catx ■ → Correct: cat
5. rn ■ → Correct: mv

Part C – Shell Script Questions

Q1. Write a shell script that prints 'Hello, World!' to the terminal.

```
#!/bin/bash
echo "Hello, World!"
```

Q2. Declare a variable named 'name' and assign the value 'CDAC Mumbai' to it. Print the value.

```
#!/bin/bash
name="CDAC Mumbai"
echo $name
```

Q3. Write a shell script that takes a number as input from the user and prints it.

```
#!/bin/bash
read -p "Enter a number: " num
echo "You entered: $num"
```

Q4. Write a shell script that performs addition of two numbers (5 and 3)

```
#!/bin/bash
a=5
b=3
sum=$((a+b))
echo "Sum: $sum"
```

Q5. Write a shell script that checks if a number is even or odd.

```
#!/bin/bash
read -p "Enter a number: " num
if (( num % 2 == 0 ))
then
   echo "Even"
else
   echo "Odd"
fi
```

Q6. Write a shell script that uses a for loop to print numbers from 1 to 5.

```
#!/bin/bash
for i in {1..5}
do
   echo $i
done
```

Q7. Write a shell script that uses a while loop to print numbers from 1 to 5.

```
#!/bin/bash
i=1
while [ $i -le 5 ]
do
   echo $i
   ((i++))
done
```

Q8. Write a shell script that checks if 'file.txt' exists in the current directory.

```
#!/bin/bash
if [ -f file.txt ]
then
   echo "File exists"
else
   echo "File does not exist"
fi
```

Q9. Write a shell script that checks if a number is greater than 10.

```
#!/bin/bash
read -p "Enter a number: " num
if [ $num -gt 10 ]
then
   echo "Number is greater than 10"
else
   echo "Number is 10 or less"
fi
```

Q10. Write a shell script that prints multiplication table (1 to 5).

```
#!/bin/bash
for i in {1..5}
do
   for j in {1..5}
   do
      printf "%4d" $((i*j))
   done
   echo
done
```

Q11. Write a shell script that reads numbers until a negative number is entered.

```
#!/bin/bash
while true
do
   read -p "Enter a number: " num
   if [ $num -lt 0 ]
   then
      break
```

```
    fi
    echo "Square: $((num*num))"
done
```