CDAC MUMBAI
Concepts of Operating System
Assignment 2

ANSWERS

Part A

What will the following commands do?

 echo "Hello, World!"
Prints the text Hello, World! to the terminal.

 name="Productive"
Creates a shell variable named name and assigns it the value Productive.

 touch file.txt
Creates an empty file named file.txt or updates it.

 ls -a
Lists all files

 rm file.txt
Deletes the file named file.txt.

 cp file1.txt file2.txt
Copies file1.txt to a new file named file2.txt.

 mv file.txt /path/to/directory/
Moves file.txt into the given directory

 chmod 755 script.sh
Changes permissions of script.sh so that owner can read/write/execute, others can read/execute.

 grep "pattern" file.txt
Searches for the word "pattern" inside file.txt and displays matching lines.

 kill PID
Terminates the process with the given process ID.

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

Make a directory mydir, move into it, create file.txt, write "Hello, World!" into file.txt, display its contents

 ls -l | grep ".txt"
Lists files in long format and filters results to show only files with .txt in their name.

 cat file1.txt file2.txt | sort | uniq
Combines the contents of file1.txt and file2.txt, sorts them, and removes duplicate lines.

 ls -l | grep "^d"
Lists files in long format, then shows only directories lines starting with d

 grep -r "pattern" /path/to/directory/
Recursively searches for "pattern" inside all files under the given directory.

 cat file1.txt file2.txt | sort | uniq –d
Combines file1.txt and file2.txt, sorts them, and shows only the duplicate lines.

 chmod 644 file.txt
Sets permissions of file.txt → owner can read/write, others can only read.

 cp -r source_directory destination_directory
Recursively copies a directory and its contents to a new location.

 find /path/to/search -name "*.txt"
Finds all files ending with .txt inside /path/to/search.

 chmod u+x file.txt
Adds execute permission for the file's owner.

 echo $PATH
Displays the system's PATH environment 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.
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.
TRUE
8. rm -rf file.txt deletes a file forcefully without confirmation.
TRUE

Identify the Incorrect Commands:

1. chmodx is used to change file permissions.
chmod
2. cpy is used to copy files and directories.
cp
3. mkfile is used to create a new file.
touch
4. catx is used to concatenate files.
cat
5. rn is used to rename files.
mv

Part C

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

```
echo "Hello, World!"
```

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

```
name="CDAC Mumbai"
echo $name
```

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

```
echo "Enter a number:"
read num
echo "You entered: $num"
```

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

```
a=5
b=3
sum=$((a + b))
echo "Sum = $sum"
```

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

```
echo "Enter a number:"
read num

if (( num % 2 == 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.

```
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.

```
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".

```
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.

```
echo "Enter a number:"
read num

if [ $num -gt 10 ]; then
   echo "Number is greater than 10"
else
   echo "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.

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

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.

```
while true
do
   echo "Enter a number (negative to stop):"
   read num
   if [ $num -lt 0 ]; then
      break
   fi
   echo "Square: $((num * num))"
done
```