**Concepts of Operating System**

**Assignment 2**

**Part A**

**Q. What will the following commands do?**

**echo "Hello, World!"**  
This will print Hello, World! In the terminal

**name="Productive”**  
This will print nothing

**touch file.txt**   
This will create a file in you current directory

**ls -a**   
This will list down all the files and directories along with hidden files

**rm file.txt**  
This will remove the file named as file.txt

**cp file1.txt file2.txt**  
This will copy the contents of the file file1.txt to a new file file2.txt. If file2.txt is already existing, it will overwrite with the contents of copy1.txt

**mv file.txt /path/to/directory**  
This will move the file named file.txt to the given path. If this file name is already exisiting in the given directory, it will overwrite.

**chmod 755 script.sh**   
This will change the permission of this file to all users as follows:  
owner: rwx, group: rx, others: rx

**grep "pattern" file.txt**this will print all the lines from the file.txt containing the word “pattern”

**kill PID**This will give an error stating that we need to pass PID of running processes. Eg. Kill 842

**mkdir mydir && cd mydir && touch file.txt && echo "Hello, World!" > file.txt && cat file.txt**This will create a directory called mydir, go into this created directory, create a file named file.txt, run command echo with “Hello, World!” text which will get put into the file file.txt, which gets newly created. Finally, cat command will print the statement of the file.txt.

**ls -l | grep ".txt**  
This will print long list of all the files containing the extention .txt

**cat file1.txt file2.txt | sort | uniq –d**  
This will print all lines which are common in between file1.txt and file2.txt in a sorted fashion.

**chmod 644 file.txt**This will change permission of file.txt for users as follows  
owner: rw, group: r, other: r

**cp -r source\_directory destination\_directory**  
This will copy the entire source directory to the destination directory

**find /path/to/search -name "\*.txt”**  
This will print all the files with .txt extension in the mentioned path

**chmod u+x file.txt**  
This will set the execution permission of the file.txt for the owner/user

**echo $PATH**   
This will list the path where the system can look for the commands that we want to execute. For eg, cat, grep, etc.

**PART B**

**Q. Identify True or False:**

**ls** is used to list files and directories in a directory: True

**mv** is used to move files and directories. : True

**cd** is used to copy files and directories. : False

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

**grep** is used to search for patterns in files. : True

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

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

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

**Q. Identify the Incorrect Commands:**

**chmodx** is used to change file permissions.   
Wrong. It is chmod.

**cpy** is used to copy files and directories.  
Wrong. It is cp

**mkfile** is used to create a new file.  
Wrong. It is touch

**catx** is used to concatenate files.  
Wrong. It is cat file1.txt file2.txt

**rn** is used to rename files.   
Wrong. It is mv name.txt editedname.txt

**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 a  
echo "Your number is $a"

**Question 4: Write a shell script that performs addition of two numbers (e.g., 5 and 3) and prints the result.**   
echo "Enter two numbers"  
read a  
read b  
(( result = a + b ))  
echo "Addition is $result"

**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 a  
if [[ (a%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.**   
for a in {1..5}  
do  
 echo "$a"  
done

**Question 7: Write a shell script that uses a while loop to print numbers from 1 to 5.**   
i=1  
while [ $i -lt 6 ];  
do  
 echo "$i"  
 ((i++))  
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 a  
if [[ ($a -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**a=1  
for a in {1..5}  
do  
 for b in {1..10}  
 do  
 (( result=a\*b ))  
 echo "$a x $b = $result"  
 done  
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.**echo "Enter a number"  
read a  
while [ true ];  
do  
 if [[ a -gt -1 ]]; then  
 (( square=a\*a ))  
 echo "square is $square"  
 echo "Enter number again"  
 read a  
 else  
 break  
 fi  
done