

Assignment -2

Part B

Identify True or False:

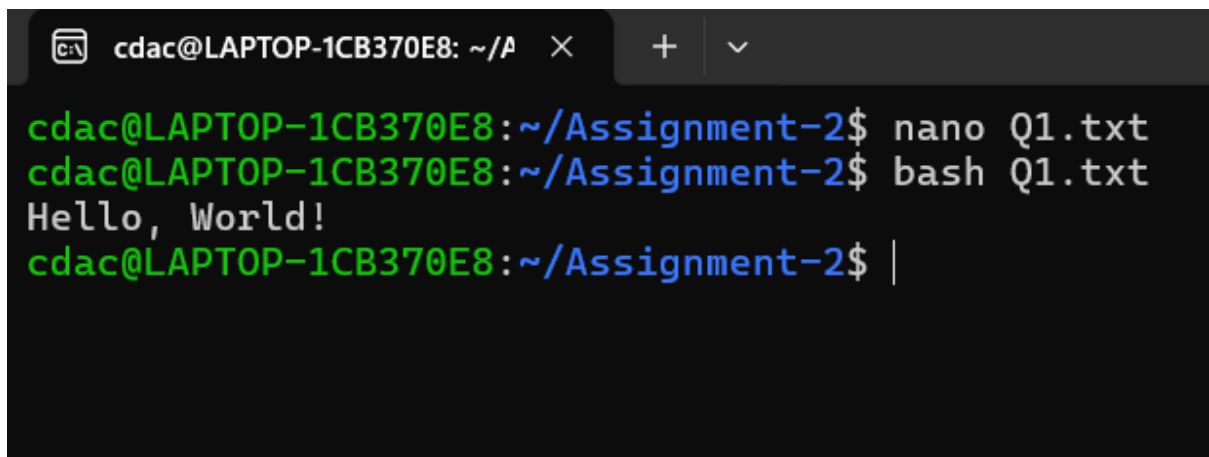
1. ls is used to list files and directories in a directory. - True
2. mv is used to move files and directories. - True
3. cd is used to copy files and directories. – False (cp is used for copy; cd is used to change directory)
4. pwd stands for "print working directory" and displays the current directory. - 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:

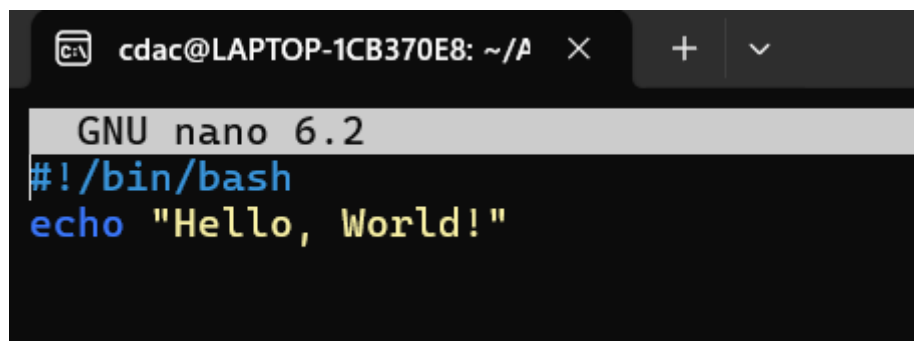
1. chmodx is used to change file permissions. - [chmod](#)
2. cpy is used to copy files and directories. - [cp](#)
3. mkfile is used to create a new file. – [touch/nano](#)
4. catx is used to concatenate files. -[cat](#)
5. rn is used to rename files. - [re](#)

Part C

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



```
cdac@LAPTOP-1CB370E8: ~/A × + v
cdac@LAPTOP-1CB370E8:~/Assignment-2$ nano Q1.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q1.txt
Hello, World!
cdac@LAPTOP-1CB370E8:~/Assignment-2$ |
```



```
GNU nano 6.2
#!/bin/bash
echo "Hello, World!"
```

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

```
cdac@LAPTOP-1CB370E8: ~/A × + v
cdac@LAPTOP-1CB370E8:~/Assignment-2$ ls
Q1.txt Q2.txt Q3.txt Q4.txt Q5.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ nano Q2.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q2.txt
CDAC Mumbai
cdac@LAPTOP-1CB370E8:~/Assignment-2$ |
```

```
cdac@LAPTOP-1CB370E8: ~/A × + v
GNU nano 6.2
#!/bin/bash
name = "CDAC Mumbai"
echo $name
```

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

```
cdac@LAPTOP-1CB370E8: ~/A × + ∨  
cdac@LAPTOP-1CB370E8:~/Assignment-2$ ls  
Q1.txt Q2.txt Q3.txt Q4.txt Q5.txt  
cdac@LAPTOP-1CB370E8:~/Assignment-2$ nano Q3.txt  
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q3.txt  
Enter any number of your choice:  
3  
The number you typed is: 3  
cdac@LAPTOP-1CB370E8:~/Assignment-2$ |
```

```
cdac@LAPTOP-1CB370E8: ~/A × + ∨  
GNU nano 6.2  
#!/bin/bash  
echo "Enter any number of your choice: "  
read number  
echo "The number you typed is: $number"
```

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

```
cdac@LAPTOP-1CB370E8: ~/A × + v
cdac@LAPTOP-1CB370E8:~/Assignment-2$ ls
Q1.txt Q2.txt Q3.txt Q4.txt Q5.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ nano Q4.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ nano Q4.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q4.txt
Enter first number:
3
Enter second number:
4
Sum of both numbers is: 7
cdac@LAPTOP-1CB370E8:~/Assignment-2$ |
```

```
GNU nano 6.2
#!/bin/bash

echo "Enter first number: "
read num1
echo "Enter second number: "
read num2
Result=`expr $num1 + $num2`
echo "Sum of both numbers is: $Result"
```

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

```
cdac@LAPTOP-1CB370E8: ~/A × + v
cdac@LAPTOP-1CB370E8:~/Assignment-2$ ls
Q1.txt Q2.txt Q3.txt Q4.txt Q5.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ nano Q5.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q5.txt
Enter a number:
8
Even
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q5.txt
Enter a number:
7
Odd
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q5.txt
Enter a number:
0
Even
cdac@LAPTOP-1CB370E8:~/Assignment-2$ |
```

```
GNU nano 6.2
#!/bin/bash

# Prompt the user to enter a number
echo "Enter a number: "
read number

# Check if the number is even or odd using modulus operation
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.

```
cdac@LAPTOP-1CB370E8: ~/A × + v
cdac@LAPTOP-1CB370E8:~/Assignment-2$ ls
Q1.txt Q2.txt Q3.txt Q4.txt Q5.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ nano Q6.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ nano Q6.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q6.txt
1
2
3
4
5
cdac@LAPTOP-1CB370E8:~/Assignment-2$ |
```

```
GNU nano 6.2 Q6.txt
#!/bin/bash
a=0
for a in 1 2 3 4 5
do
    echo $a
done
```


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

```
cdac@LAPTOP-1CB370E8: ~/A × + v
cdac@LAPTOP-1CB370E8:~/Assignment-2$ ls
Q1.txt Q2.txt Q3.txt Q4.txt Q5.txt Q6.txt Q7.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ nano Q7.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q7.txt
1
2
3
4
5
cdac@LAPTOP-1CB370E8:~/Assignment-2$ |
```

```
GNU nano 6.2 Q7.txt
#!/bin/bash

count=1

while [ $count -le 5 ]
do
    echo $count

    # Increment
    count=$((count + 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".

```
cdac@LAPTOP-1CB370E8: ~/Assignment-2$ ls
Q1.txt Q2.txt Q3.txt Q4.txt Q5.txt Q6.txt Q7.txt Q8.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ nano Q8.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q8.txt
File is not present
cdac@LAPTOP-1CB370E8:~/Assignment-2$ touch file.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ ls
Q1.txt Q2.txt Q3.txt Q4.txt Q5.txt Q6.txt Q7.txt Q8.txt file.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q8.txt
File is present
cdac@LAPTOP-1CB370E8:~/Assignment-2$ |
```

```
GNU nano 6.2 Q8.txt
#!/bin/bash

if [ -f "file.txt" ];
then
echo "File is present"
else
echo "File is not present"
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.

```
cdac@LAPTOP-1CB370E8: ~/A × + v
cdac@LAPTOP-1CB370E8:~/Assignment-2$ ls
Q1.txt  Q3.txt  Q5.txt  Q7.txt  Q9.txt
Q2.txt  Q4.txt  Q6.txt  Q8.txt  file.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ nano Q9.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q9.txt
Enter a number
9
This number is less or equal to 10
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q9.txt
Enter a number
10
This number is less or equal to 10
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q9.txt
Enter a number
11
This number is greater than 10
cdac@LAPTOP-1CB370E8:~/Assignment-2$ |
```

```
cdac@LAPTOP-1CB370E8: ~/A × + v
GNU nano 6.2 Q9.txt
#!/bin/bash
echo "Enter a number"
read number
if [ $number -gt 10 ];
then
    echo "This number is greater than 10"
else
    echo "This number is less or equal to 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.

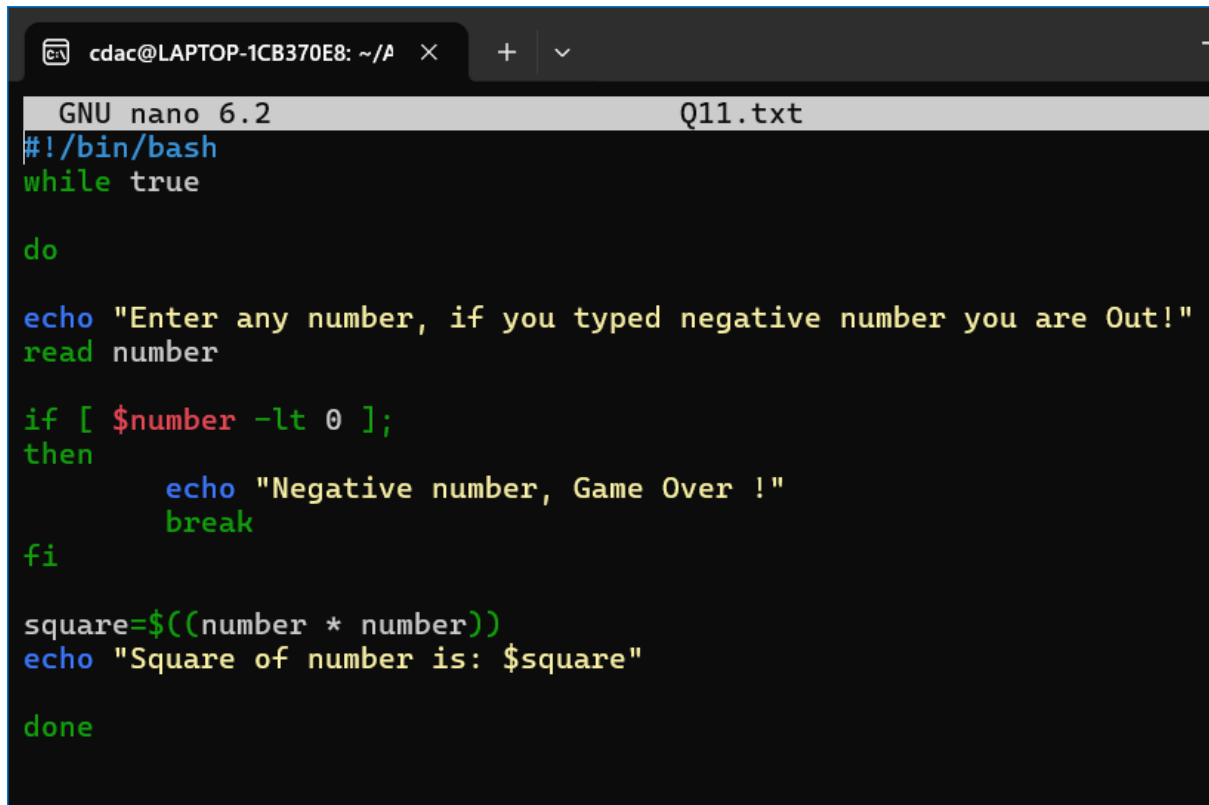
```
cdac@LAPTOP-1CB370E8: ~/Assignment-2$ ls
Q1.txt  Q11.txt  Q3.txt  Q5.txt  Q7.txt  Q9.txt
Q10.txt Q2.txt   Q4.txt  Q6.txt  Q8.txt  file.txt
cdac@LAPTOP-1CB370E8: ~/Assignment-2$ nano Q10.txt
cdac@LAPTOP-1CB370E8: ~/Assignment-2$ bash Q10.txt
  1   2   3   4   5   6   7   8   9  10
  2   4   6   8  10  12  14  16  18  20
  3   6   9  12  15  18  21  24  27  30
  4   8  12  16  20  24  28  32  36  40
  5  10  15  20  25  30  35  40  45  50
cdac@LAPTOP-1CB370E8: ~/Assignment-2$ |
```

```
GNU nano 6.2 Q10.txt
#Table
#!/bin/bash

for i in 1 2 3 4 5
do
    for j in 1 2 3 4 5 6 7 8 9 10
    do
        table=$((i*j))
        printf "%4d" $table
    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.

A screenshot of a terminal window with a dark background. The window title bar shows 'cdac@LAPTOP-1CB370E8: ~/A' and a tab icon. Below the title bar, the text 'GNU nano 6.2' and 'Q11.txt' are visible. The main area contains a shell script written in a syntax-highlighted font. The script uses a 'while true' loop to repeatedly prompt the user for a number. If the number is negative, it prints 'Negative number, Game Over !' and uses 'break' to exit the loop. Otherwise, it calculates the square of the number and prints it.

```
#!/bin/bash
while true
do
    echo "Enter any number, if you typed negative number you are Out!"
    read number

    if [ $number -lt 0 ];
    then
        echo "Negative number, Game Over !"
        break
    fi

    square=$((number * number))
    echo "Square of number is: $square"
done
```



cdac@LAPTOP-1CB370E8: ~/A



```
cdac@LAPTOP-1CB370E8:~/Assignment-2$ ls
Q1.txt  Q2.txt  Q4.txt  Q6.txt  Q8.txt  file.txt
Q11.txt Q3.txt  Q5.txt  Q7.txt  Q9.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ nano Q11.txt
cdac@LAPTOP-1CB370E8:~/Assignment-2$ bash Q11.txt
Enter any number, if you typed negative number you are Out!
2
Square of number is: 4
Enter any number, if you typed negative number you are Out!
3
Square of number is: 9
Enter any number, if you typed negative number you are Out!
5
Square of number is: 25
Enter any number, if you typed negative number you are Out!
-1
Negative number, Game Over !
cdac@LAPTOP-1CB370E8:~/Assignment-2$ |
```