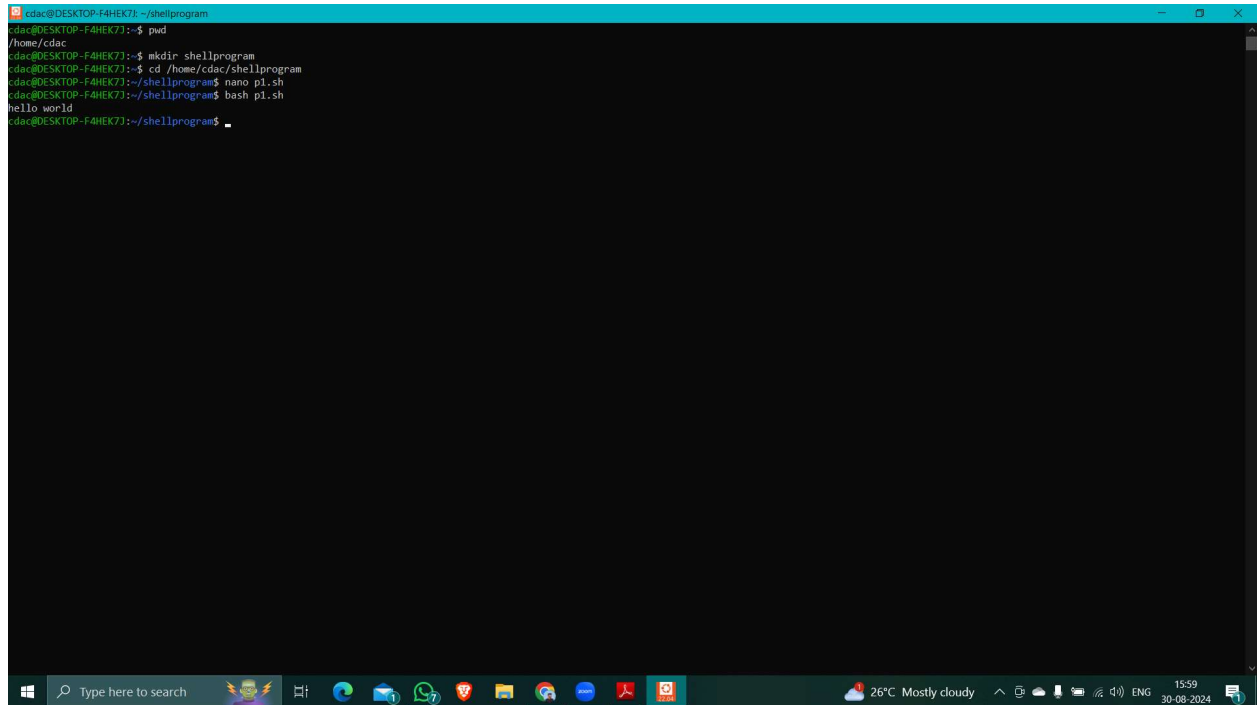


# Assignment -2 OS

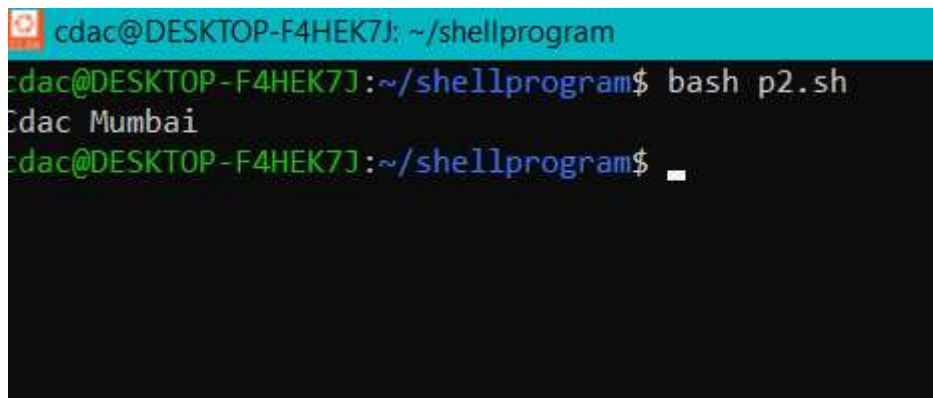
## Part c

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



```
cdac@DESKTOP-F4HEK7J: ~/shellprogram
cdac@DESKTOP-F4HEK7J:~$ pwd
/home/cdac
cdac@DESKTOP-F4HEK7J:~$ mkdir shellprogram
cdac@DESKTOP-F4HEK7J:~$ cd /home/cdac/shellprogram
cdac@DESKTOP-F4HEK7J:~/shellprogram$ nano p1.sh
cdac@DESKTOP-F4HEK7J:~/shellprogram$ bash p1.sh
hello world
cdac@DESKTOP-F4HEK7J:~/shellprogram$
```

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



```
cdac@DESKTOP-F4HEK7J: ~/shellprogram
cdac@DESKTOP-F4HEK7J:~/shellprogram$ bash p2.sh
CDAC Mumbai
cdac@DESKTOP-F4HEK7J:~/shellprogram$
```

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

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

```
cdac@DESKTOP-F4HEK7J: ~/shellprogram
cdac@DESKTOP-F4HEK7J:~/shellprogram$ bash p3.sh
enter number
01082001

cdac@DESKTOP-F4HEK7J:~/shellprogram$ nano p3.sh
cdac@DESKTOP-F4HEK7J:~/shellprogram$ bash p3.sh
enter number
01082001
01082001
cdac@DESKTOP-F4HEK7J:~/shellprogram$ _
```

```
cdac@DESKTOP-F4HEK7J: ~/shellprogram
GNU nano 6.2
#!/bin/bash
echo "enter number 1:"
read number1
echo "enter number 2:"
read number2

result=$((number1+number2))

echo $result
```

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

```
cdac@DESKTOP-F4HEK7J: ~  
cdac@DESKTOP-F4HEK7J:~$ nano p5  
cdac@DESKTOP-F4HEK7J:~$ bash p5  
enter number:  
10  
even  
cdac@DESKTOP-F4HEK7J:~$ bash p5  
enter number:  
5  
odd  
cdac@DESKTOP-F4HEK7J:~$ nano p5  
cdac@DESKTOP-F4HEK7J:~$ _
```

```
cdac@DESKTOP-F4HEK7J: ~  
GNU nano 6.2  
#!/bin/bash  
echo enter number:  
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.

```
cdac@DESKTOP-F4HEK7J: ~/shellprog  
GNU nano 6.2  
#!/bin/bash  
  
for n in {1..5..1}  
do  
    echo $n  
done
```

```
cdac@DESKTOP-F4HEK7J: ~/shellprogram  
cdac@DESKTOP-F4HEK7J:~$ cd /home/cdac/shellprogram  
cdac@DESKTOP-F4HEK7J:~/shellprogram$ ls  
p1.sh p2.sh p3.sh p4.sh  
cdac@DESKTOP-F4HEK7J:~/shellprogram$ nano p5.sh  
cdac@DESKTOP-F4HEK7J:~/shellprogram$ bash p5.sh  
p5.sh: line 3: syntax error near unexpected token `in[1..5..1]'  
p5.sh: line 3: `for n in[1..5..1]'  
cdac@DESKTOP-F4HEK7J:~/shellprogram$ nano p5.sh  
cdac@DESKTOP-F4HEK7J:~/shellprogram$ bash p5.sh  
1  
2  
3  
4  
5  
cdac@DESKTOP-F4HEK7J:~/shellprogram$
```

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

```
cdac@DESKTOP-F4HEK7J: ~/shellprogram
cdac@DESKTOP-F4HEK7J:~$ cd /home/cdac/shellprogram
cdac@DESKTOP-F4HEK7J:~/shellprogram$ ls
p1.sh p2.sh p3.sh p4.sh p5.sh
cdac@DESKTOP-F4HEK7J:~/shellprogram$ nano p6.sh
cdac@DESKTOP-F4HEK7J:~/shellprogram$ bash p6.sh
1
2
3
4
5
cdac@DESKTOP-F4HEK7J:~/shellprogram$ nano p6.sh
cdac@DESKTOP-F4HEK7J:~/shellprogram$
```

```
cdac@DESKTOP-F4HEK7J: ~/shellprogram
GNU nano 6.2
#!/bin/bash

a=1

while [ $a -le 5 ]
do
    echo $a
    a=$((a+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@DESKTOP-F4HEK7J: ~/shellprogram
cdac@DESKTOP-F4HEK7J:~/shellprogram$ bash p8.sh
file exist
cdac@DESKTOP-F4HEK7J:~/shellprogram$ nano p8.sh
cdac@DESKTOP-F4HEK7J:~/shellprogram$
```

```
cdac@DESKTOP-F4HEK7J: ~/shellprogram
GNU nano 6.2
#!/bin/bash

if [ -f "p6.sh" ]
then
    echo file exist
else
    echo file not there
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@DESKTOP-F4HEK7J: ~/shellprogram
cdac@DESKTOP-F4HEK7J:~/shellprogram$ bash p7.sh
Enter number:
14
14 is greater then 10
cdac@DESKTOP-F4HEK7J:~/shellprogram$
```

```
cdac@DESKTOP-F4HEK7J: ~/shellprogram
GNU nano 6.2
#!/bin/bash

echo "Enter number:"
read num

if [ $num -gt 10 ]
then
    echo "$num is greater then 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@DESKTOP-F4HEK7J: ~/shellprogram
cdac@DESKTOP-F4HEK7J:~/shellprogram$ nano p9.sh
cdac@DESKTOP-F4HEK7J:~/shellprogram$ bash p9.sh
.....1.....table.....
1
2
3
4
5
6
7
8
9
10
.....2.....table.....
2
4
6
8
10
12
14
16
18
20
.....3.....table.....
3
6
9
12
15
18
21
24
27
30
.....4.....table.....
4
8
12
16
20
24
28
32
36
40
.....5.....table.....
5
10
15
20
25
30
35
40
45
```



```
#!/bin/bash

for n in {1..5..1}
do
    echo "....."$n".....table....."
    for m in {1..10..1}
    do
        echo $(( $n * $m ))
    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.

```
cdac@DESKTOP-F4HEK7J: ~/shellprogram
cdac@DESKTOP-F4HEK7J:~$ cd /home/cdac/shellprogram
cdac@DESKTOP-F4HEK7J:~/shellprogram$ ls
p1.sh p2.sh p3.sh p4.sh p5.sh p6.sh p7.sh p8.sh p9.sh
cdac@DESKTOP-F4HEK7J:~/shellprogram$ nano p10.sh
cdac@DESKTOP-F4HEK7J:~/shellprogram$ bash p10.sh
Enter Number:
10
100
Enter Number:
2
4
Enter Number:
-1
1
cdac@DESKTOP-F4HEK7J:~/shellprogram$ nano p10.sh
cdac@DESKTOP-F4HEK7J:~/shellprogram$
```

```
#!/bin/bash

while [ 1 ]
do
    echo "Enter Number:"
    read num
    echo $(( $num * $num ))
    if [ $num -lt 0 ]
    then
        break
    fi
done
```

