1.Write a program to reverse a word using loop?(Not to use inbuilt functions)

Sample Input:

String: TEMPLE

Sample Output:

Reverse String: ELPMET

Test cases:SIGN UPAT-LEAST1245!@#$%145*999=144855

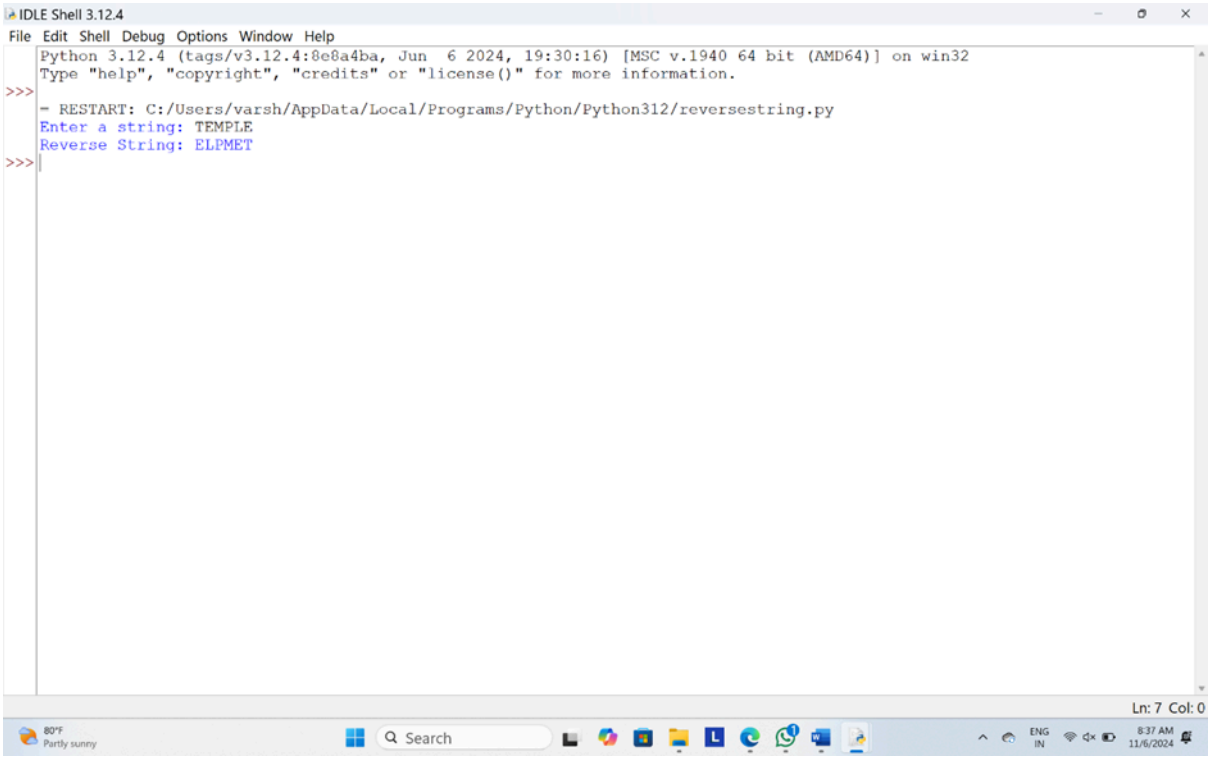
PROGRAM:

```
# Function to reverse a string using a loop
def reverse_string(word):
        reversed_word = ""
        for i in range(len(word) - 1, -1, -1):  # Looping backward
        reversed_word += word[i]
        return reversed_word


# Example usage
input_string = input("Enter a string: ")  # Prompt user for input
output_string = reverse_string(input_string)
print(f"Reverse String: {output_string}")
```

OUTPUT:

Enter a string: TEMPLE

Reverse String: ELPMET



2.Write a program to calculate Pow(x,n), Add(x,n), Sub(x,n), Mul(x,n), Div(x,n)? Get the input and choice from the user.

Sample Input:

X = 2

N = 4

Choice : 2

Sample Output:

Add(X,N) = 6

Test cases:X = 0 , N = 4X = 5 , N = 0X = -3 , N = 3X = 0 , N = 0X = 123, N = 123

PROGRAM:

```
# Function definitions for various operations

def power(x, n):

        return x ** n


def add(x, n):
```

```
        return x + n


def subtract(x, n):
        return x - n


def multiply(x, n):
        return x * n


def divide(x, n):
        if n != 0:
        return x / n
        else:
        return "Division by zero is not allowed"


# Menu and input from the user
def menu():
        print("Choose an operation:")
        print("1. Power (Pow(x, n))")
        print("2. Addition (Add(x, n))")
        print("3. Subtraction (Sub(x, n))")
        print("4. Multiplication (Mul(x, n))")
        print("5. Division (Div(x, n))")


# Main program
def main():
        # Getting inputs
```

```
        x = float(input("Enter value for X: "))
        n = float(input("Enter value for N: "))

        # Display menu
        menu()
        choice = int(input("Enter your choice (1-5): "))

        # Perform the chosen operation
        if choice == 1:
    print(f"Pow({x}, {n}) = {power(x, n)}")
        elif choice == 2:
    print(f"Add({x}, {n}) = {add(x, n)}")
        elif choice == 3:
    print(f"Sub({x}, {n}) = {subtract(x, n)}")
        elif choice == 4:
    print(f"Mul({x}, {n}) = {multiply(x, n)}")
        elif choice == 5:
    print(f"Div({x}, {n}) = {divide(x, n)}")
        else:
    print("Invalid choice! Please choose a number between 1 and 5.")

# Run the program
if __name__ == "__main__":
        main()
```

OUTPUT:

Enter value for X: 2

Enter value for N: 4

Choose an operation:

1. Power (Pow(x, n))

2. Addition (Add(x, n))

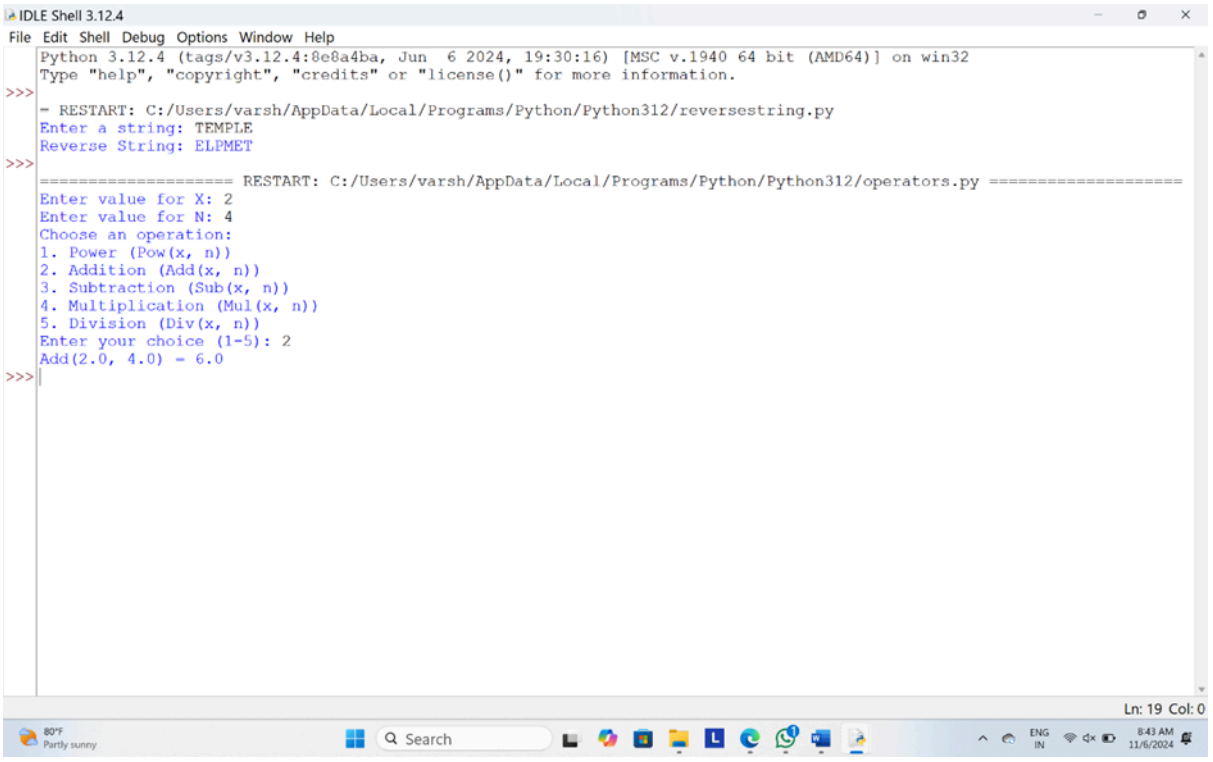
3. Subtraction (Sub(x, n))

4. Multiplication (Mul(x, n))

5. Division (Div(x, n))

Enter your choice (1-5): 2

Add(2.0, 4.0) = 6.0



3.Write a program to count all the prime and composite numbers entered by the user.

Sample Input:

Enter the numbers

4

54

29

71

7

59

98

23

Sample Output:

Composite number:3

Prime number:5

Test cases:33, 41, 52, 61,73,90TEN, FIFTY, SIXTY-ONE, SEVENTY-SEVEN, NINE45, 87, 09, 5.0 ,2.3, 0.4-54, -76, -97, -23, -33, -9845, 73, 00, 50, 67, 44

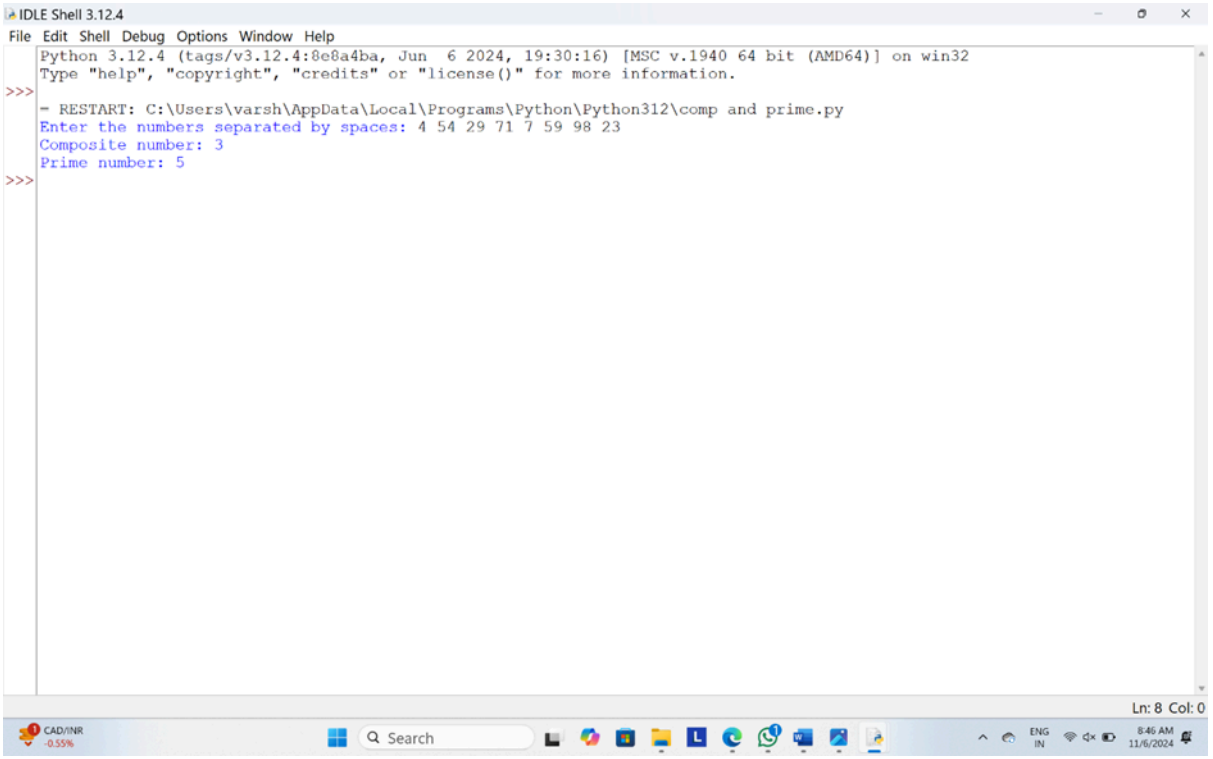
PROGRAM:

```
# Function to check if a number is prime
def is_prime(num):
    if num <= 1:
    return False  # Numbers less than 2 are not prime
    for i in range(2, int(num ** 0.5) + 1):
    if num % i == 0:
    return False
    return True


# Main function to count prime and composite numbers
def count_prime_and_composite(numbers):
    prime_count = 0
    composite_count = 0


    for num in numbers:
```

```
        if num > 1:  # Only consider numbers greater than 1 for prime/composite
            if is_prime(num):
                prime_count += 1
            else:
                composite_count += 1

        return prime_count, composite_count

# Taking input from the user
numbers = list(map(int, input("Enter the numbers separated by spaces: ").split()))

# Count primes and composites
prime_count, composite_count = count_prime_and_composite(numbers)

# Display the results
print(f"Composite number: {composite_count}")
print(f"Prime number: {prime_count}")
```

OUTPUT:

```
Enter the numbers separated by spaces: 4 54 29 71 7 59 98 23
Composite number: 3
Prime number: 5
```



4.Write a program to check the entered user name is valid or not. Get both the inputs from the user.

Sample Input:

Enter the user name: Saveetha@789

Reenter the user name: Saveetha@123

Sample Output:

User name is Invalid

PROGRAM:

```
# Function to check if the two usernames match

def check_username(username, reentered_username):

    if username == reentered_username:

    return "User name is Valid"

    else:

    return "User name is Invalid"


# Taking inputs from the user
```

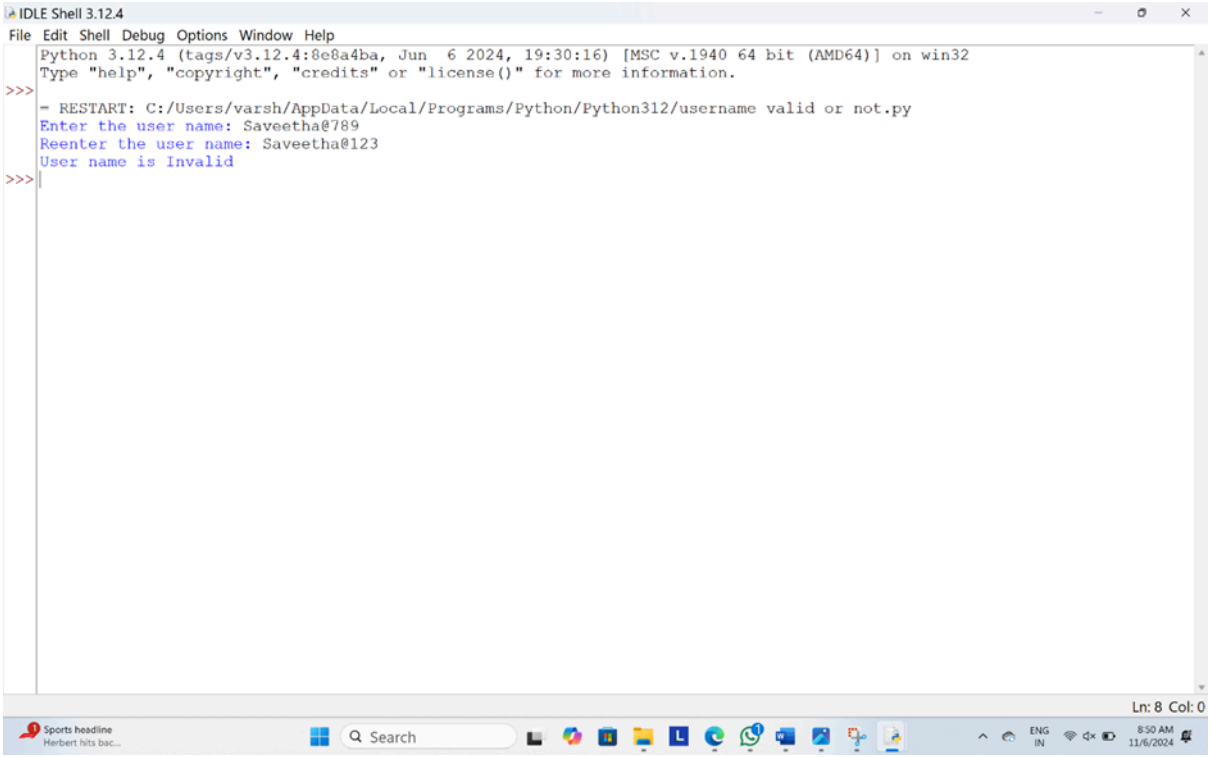
```
username = input("Enter the user name: ")

reentered_username = input("Reenter the user name: ")


# Checking if the username is valid

result = check_username(username, reentered_username)


# Display the result

print(result)
```

OUTPUT:

Enter the user name: Saveetha@789

Reenter the user name: Saveetha@123

User name is Invalid



5.Find the Mth maximum number and Nth minimum number in an array and then find the sum of it and difference of it.

Sample Input:

Array of elements = {14, 16, 87, 36, 25, 89, 34}

M = 1

N = 3

Sample Output:

1st Maximum Number = 89

3rd Minimum Number = 25

Sum = 114

Difference = 64

Test cases:{16, 16, 16 16, 16}, M = 0, N = 1{0, 0, 0, 0}, M = 1, N = 2{-12, -78, -35, -42, -85}, M = 3 , N = 3{15, 19, 34, 56, 12}, M = 6 , N = 3{85, 45, 65, 75, 95}, M = 5 , N = 7

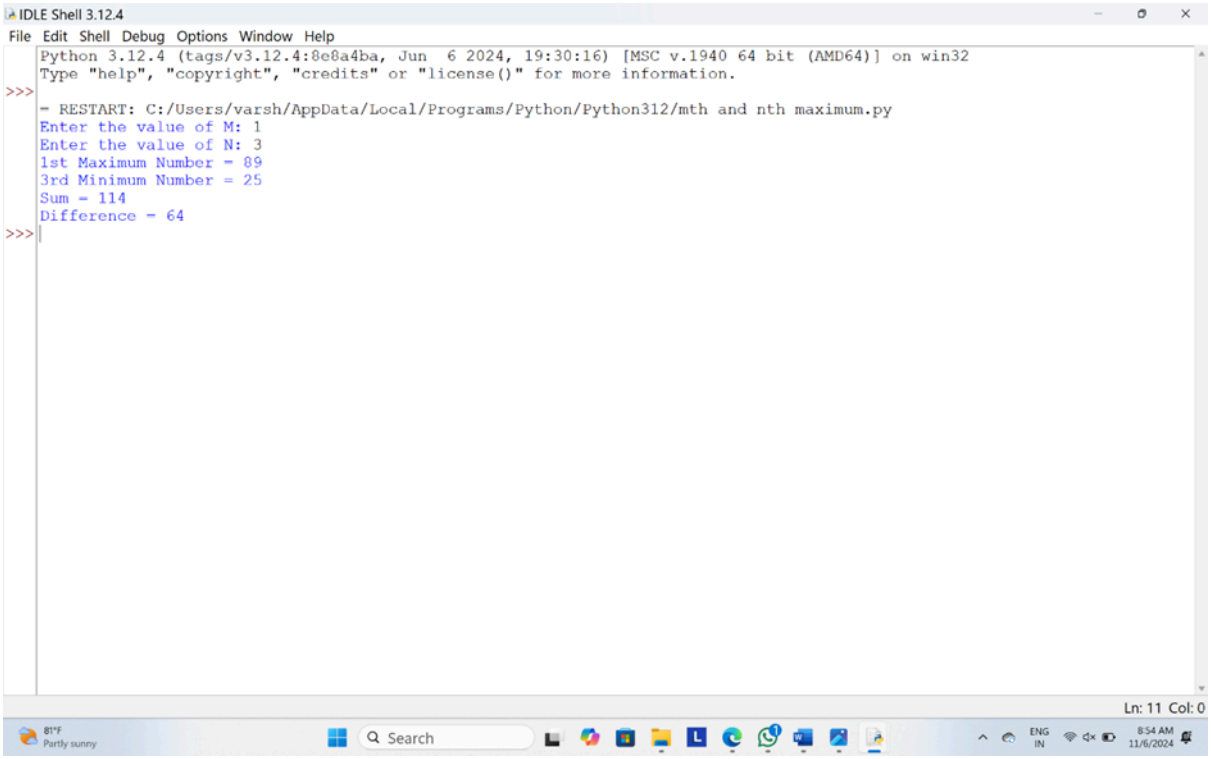
PROGRAM:

```
# Function to find the Mth maximum and Nth minimum

def find_mth_max_and_nth_min(arr, M, N):

        arr_sorted = sorted(arr)  # Sort the array in ascending order

        nth_min = arr_sorted[N - 1]  # Nth minimum is at index N-1

        mth_max = arr_sorted[-M]  # Mth maximum is at index -M in a sorted array

        return mth_max, nth_min


# Taking inputs

arr = [14, 16, 87, 36, 25, 89, 34]  # Array of elements

M = int(input("Enter the value of M: "))  # Mth maximum

N = int(input("Enter the value of N: "))  # Nth minimum


# Finding Mth maximum and Nth minimum
```

```
mth_max, nth_min = find_mth_max_and_nth_min(arr, M, N)


# Calculating sum and difference
sum_result = mth_max + nth_min
difference_result = mth_max - nth_min


# Displaying results
print(f"{M}st Maximum Number = {mth_max}")
print(f"{N}rd Minimum Number = {nth_min}")
print(f"Sum = {sum_result}")
print(f"Difference = {difference_result}")
```

OUTPUT:

```
Enter the value of M: 1
Enter the value of N: 3
1st Maximum Number = 89
3rd Minimum Number = 25
Sum = 114
Difference = 64
```


IDLE Shell 3.12.4
File Edit Shell Debug Options Window Help
Python 3.12.4 (tags/v3.12.4:8e8a4ba, Jun 6 2024, 19:30:16) [MSC v.1940 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/varsh/AppData/Local/Programs/Python/Python312/mth and nth maximum.py
Enter the value of M: 1
Enter the value of N: 3
1st Maximum Number = 89
3rd Minimum Number = 25
Sum = 114
Difference = 64
>>>
Ln: 11 Col: 0
81°F
Partly sunny
Search
ENG
IN
8:54 AM
11/6/2024