1)

Print the pattern

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

Code:

def print\_pattern(rows):

for i in range(1, rows + 1):

for j in range(i):

print("\*", end=" ")

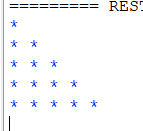
print()

# Sample Input

rows = 5

# Print the pattern

print\_pattern(rows)

Output:  


2)

Write a program to print the number of vowels and number of consonants in the given

statement and which is maximum?

Sample Input: Saveetha School of Engineering Sample Output:

Number of vowels = 12 Number of Consonants = 15

Code:

def count\_vowels\_and\_consonants(statement):

vowels = "aeiouAEIOU"

num\_vowels = 0

num\_consonants = 0

for char in statement:

if char.isalpha():

if char in vowels:

num\_vowels += 1

else:

num\_consonants += 1

return num\_vowels, num\_consonants

# Sample Input

input\_statement = "Saveetha School of Engineering"

# Count vowels and consonants

vowels\_count, consonants\_count = count\_vowels\_and\_consonants(input\_statement)

# Determine which count is maximum

if vowels\_count > consonants\_count:

max\_type = "vowels"

max\_count = vowels\_count

else:

max\_type = "consonants"

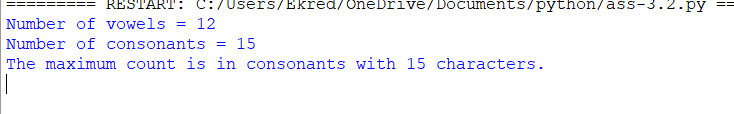
max\_count = consonants\_count

# Sample Output

print(f"Number of vowels = {vowels\_count}")

print(f"Number of consonants = {consonants\_count}")

print(f"The maximum count is in {max\_type} with {max\_count} characters.")

output:  


3)

c Get the input and choice from the user.

Sample Input:

X = 2

N = 4

Choice : 2 Sample Output:

Add(X,N) = 6

Test cases:

1. X = 0 , N = 4

2. X = 5 , N = 0

3. X = -3 , N = 3

4. X = 0 , N = 0

5. X = 123, N = 123"

Code:

def perform\_operation(X, N, choice):

if choice == 1:

result = X + N

operation = "Add"

elif choice == 2:

result = X - N

operation = "Subtract"

elif choice == 3:

result = X \* N

operation = "Multiply"

elif choice == 4:

if N == 0:

print("Error: Division by zero is not allowed.")

return

result = X / N

operation = "Divide"

else:

print("Invalid choice. Please choose a number between 1 and 4.")

return

print(f"{operation}({X},{N}) = {result}")

# Sample Input

X = int(input("Enter X: "))

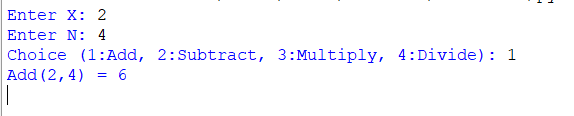
N = int(input("Enter N: "))

choice = int(input("Choice (1:Add, 2:Subtract, 3:Multiply, 4:Divide): "))

# Perform operation based on user choice

perform\_operation(X, N, choice)

output:



4)

Write a program to read the numbers until -1 is encountered. Find the average of positive numbers and negative numbers entered by the user.

Sample Input:

Enter -1 to exit…

Enter the number: 7

Enter the number: -2

Enter the number: 9

Enter the number: -8

Enter the number: -6

Enter the number: -4

Enter the number: 10

Enter the number: -1

Sample Output:

The average of negative numbers is: -5.0

The average of positive numbers is : 8.66666667

Test cases:

1. -1,43, -87, -29, 1, -9

2. 73, 7-6,2,10,28,-1

3. -5, -9, -46,2,5,0

4. 9, 11, -5, 6, 0,-1

5. -1,-1,-1,-1,-1"

Code:

def calculate\_average(numbers):

positive\_count = 0

positive\_sum = 0

negative\_count = 0

negative\_sum = 0

for num in numbers:

if num > 0:

positive\_count += 1

positive\_sum += num

elif num < 0:

negative\_count += 1

negative\_sum += num

if positive\_count > 0:

positive\_average = positive\_sum / positive\_count

else:

positive\_average = 0 # Handle case where there are no positive numbers

if negative\_count > 0:

negative\_average = negative\_sum / negative\_count

else:

negative\_average = 0 # Handle case where there are no negative numbers

return positive\_average, negative\_average

# Initialize an empty list to store numbers

numbers = []

# Input numbers until -1 is encountered

while True:

num = int(input("Enter the number (-1 to exit): "))

if num == -1:

break

numbers.append(num)

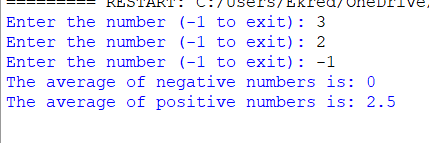
# Calculate averages

positive\_avg, negative\_avg = calculate\_average(numbers)

# Print results

print(f"The average of negative numbers is: {negative\_avg}")

print(f"The average of positive numbers is: {positive\_avg}")

Output:  


5)

Write a program to Reverse Words in a String

Given an input string s, reverse the order of the words.

Input: s = ""the sky is blue""

Output: ""blue is sky the"""

Code:

def reverse\_words(s):

# Split the string into words

words = s.split()

# Reverse the list of words

reversed\_words = words[::-1]

# Join the reversed words into a single string

reversed\_string = " ".join(reversed\_words)

return reversed\_string

# Sample Input

input\_string = "the sky is blue"

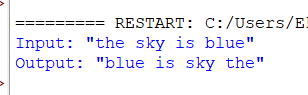
# Reverse words in the string

output\_string = reverse\_words(input\_string)

# Output

print(f"Input: \"{input\_string}\"")

print(f"Output: \"{output\_string}\"")

output:  


6)

"Write a program to print the number of vowels and number of consonants in the given

statement and which is maximum?

Sample Input:

Saveetha School of Engineering Sample Output:

Number of vowels = 12 Number of Consonants = 15"

Code:  
def count\_vowels\_and\_consonants(statement):

vowels = "aeiouAEIOU"

num\_vowels = 0

num\_consonants = 0

for char in statement:

if char.isalpha():

if char in vowels:

num\_vowels += 1

else:

num\_consonants += 1

return num\_vowels, num\_consonants

# Sample Input

input\_statement = "Saveetha School of Engineering"

# Count vowels and consonants

vowels\_count, consonants\_count = count\_vowels\_and\_consonants(input\_statement)

# Determine which count is maximum

if vowels\_count > consonants\_count:

max\_type = "vowels"

max\_count = vowels\_count

elif consonants\_count > vowels\_count:

max\_type = "consonants"

max\_count = consonants\_count

else:

max\_type = "both vowels and consonants"

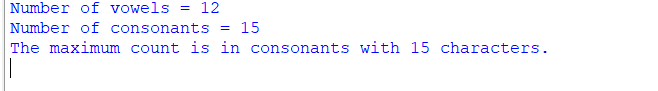
max\_count = vowels\_count # or consonants\_count, since they are equal

# Sample Output

print(f"Number of vowels = {vowels\_count}")

print(f"Number of consonants = {consonants\_count}")

print(f"The maximum count is in {max\_type} with {max\_count} characters.")

Output:  


7)

Write a program to calculate tax given the following conditions:

If income is less than or equal to 1,50,000 then no tax

If taxable income is 1,50,001 – 3,00,000 the charge 10% tax

If taxable income is 3,00,001 – 5,00,000 the charge 20% tax

If taxable income is above 5,00,001 then charge 30% tax

Sample Input:

Enter the income:200000

Sample Output:

Tax= 20000

Test cases:a) 400700

b) 2789239

c) 150000

d) 00000

e) -125486

code:

def calculate\_tax(income):

if income <= 150000:

tax = 0

elif income <= 300000:

tax = (income - 150000) \* 0.1

elif income <= 500000:

tax = 15000 + (income - 300000) \* 0.2

else:

tax = 55000 + (income - 500000) \* 0.3

return tax

# Sample Input

income = int(input("Enter the income: "))

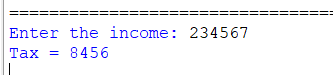
# Calculate tax

tax = calculate\_tax(income)

# Sample Output

print(f"Tax = {int(tax)}")

output:



8)

Write a program to count number of space, line, vowels and consonants in a file

Test case “ Hello,txt”

“Welcome to saveetha School of Engineering”

Code:

def count\_spaces\_lines\_vowels\_consonants(filename):

vowels = "aeiouAEIOU"

space\_count = 0

line\_count = 0

vowel\_count = 0

consonant\_count = 0

try:

with open(filename, 'r') as file:

for line in file:

line\_count += 1

space\_count += line.count(' ')

for char in line:

if char.isalpha():

if char in vowels:

vowel\_count += 1

else:

consonant\_count += 1

except FileNotFoundError:

print(f"Error: File '{filename}' not found.")

return

return space\_count, line\_count, vowel\_count, consonant\_count

# Sample Input

filename = "Hello.txt"

# Count spaces, lines, vowels, and consonants in the file

space\_count, line\_count, vowel\_count, consonant\_count = count\_spaces\_lines\_vowels\_consonants(filename)

# Sample Output

print(f"Number of spaces: {space\_count}")

print(f"Number of lines: {line\_count}")

print(f"Number of vowels: {vowel\_count}")

print(f"Number of consonants: {consonant\_count}")

9)

**Print the pattern,**

**0.1**

**0.1 0.2**

**0.1 0.2 0.3**

**0.1 0.2 0.3 0.4**

Code:

def print\_pattern(rows):

for i in range(1, rows + 1):

for j in range(1, i + 1):

print(f"{j / 10:.1f}", end=" ")

print()

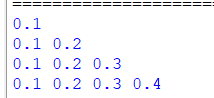
# Sample Input

rows = 4

# Print the pattern

print\_pattern(rows)

output:



10)

Write a Program to find row, column and diagonal sum in Matrix

a = [[1, 2, 3],

[4, 5, 6],

[7, 8, 9]]

o/p:

Sum of 1 row: 6

Sum of 2 row: 15

Sum of 3 row: 24

Sum of 1 column: 12

Sum of 2 column: 15

Sum of 3 column: 18

Diagonal sum 15"

Code:

def matrix\_sum(matrix):

rows = len(matrix)

cols = len(matrix[0]) if rows > 0 else 0

row\_sums = [0] \* rows

col\_sums = [0] \* cols

diagonal\_sum = 0

# Calculate row sums and column sums

for i in range(rows):

for j in range(cols):

row\_sums[i] += matrix[i][j]

col\_sums[j] += matrix[i][j]

if i == j:

diagonal\_sum += matrix[i][j]

# Print row sums

for i in range(rows):

print(f"Sum of {i + 1} row: {row\_sums[i]}")

# Print column sums

for j in range(cols):

print(f"Sum of {j + 1} column: {col\_sums[j]}")

# Print diagonal sum

print(f"Diagonal sum: {diagonal\_sum}")

# Sample Matrix

a = [

[1, 2, 3],

[4, 5, 6],

[7, 8, 9]

]

# Calculate and print sums

matrix\_sum(a)

output:

