num\_list = [3, 4, 5, 1, -44, 5, 10, 12, 33, 1]

k = 3

start\_indexes = list(range(0, len(num\_list) - k + 1))

end\_indexes = list(range(k, len(num\_list) + 1))

result = []

for start\_index, end\_index in zip(start\_indexes, end\_indexes):

    sub\_list = num\_list[start\_index: end\_index]

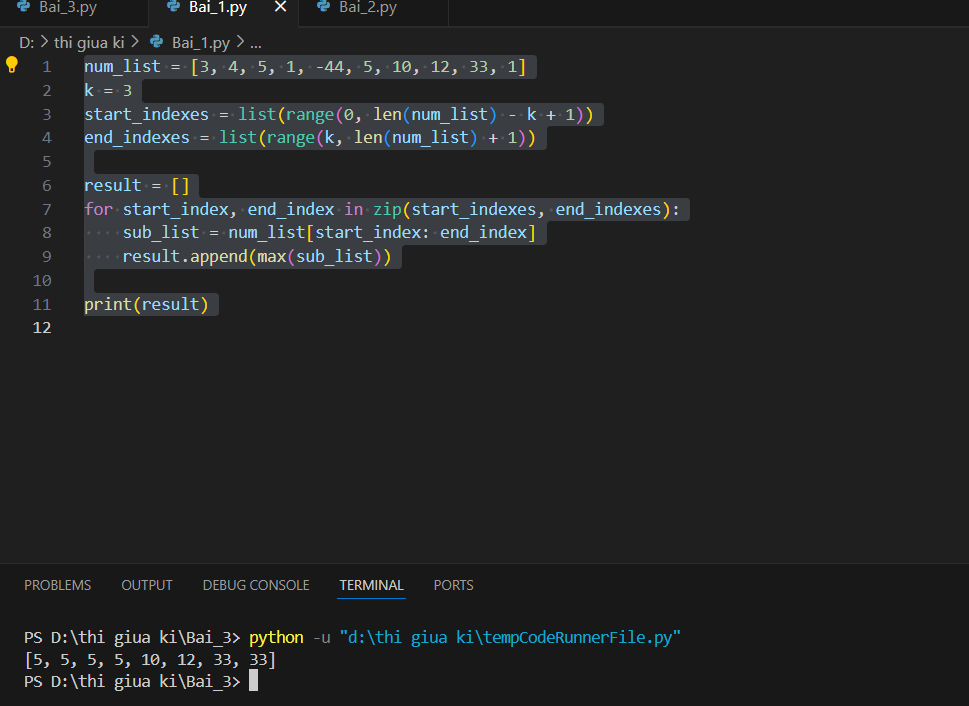
    result.append(max(sub\_list))

print(result)

ĐÁP ÁN:  
PS D:\thi giua ki\Bai\_3> python -u "d:\thi giua ki\tempCodeRunnerFile.py"

[5, 5, 5, 5, 10, 12, 33, 33]

PS D:\thi giua ki\Bai\_3>



BÀI 2:

words = ['happiness', 'smiles']

character\_statistics = {}

for word in words:

    character\_statistics[word] = {}

    for character in word:

        if character in character\_statistics[word]:

            character\_statistics[word][character] += 1

        else:

            character\_statistics[word][character] = 1

for word, stats in character\_statistics.items():

    print(f"Từ: {word}")

    for char, count in stats.items():

        print(f"  '{char}': {count}")

    print()

ĐÁP ÁN:  
PS D:\thi giua ki\Bai\_3> python -u "d:\thi giua ki\tempCodeRunnerFile.py"

Từ: happiness

'h': 1

'a': 1

'p': 2

'i': 1

'n': 1

'e': 1

's': 2

Từ: smiles

's': 2

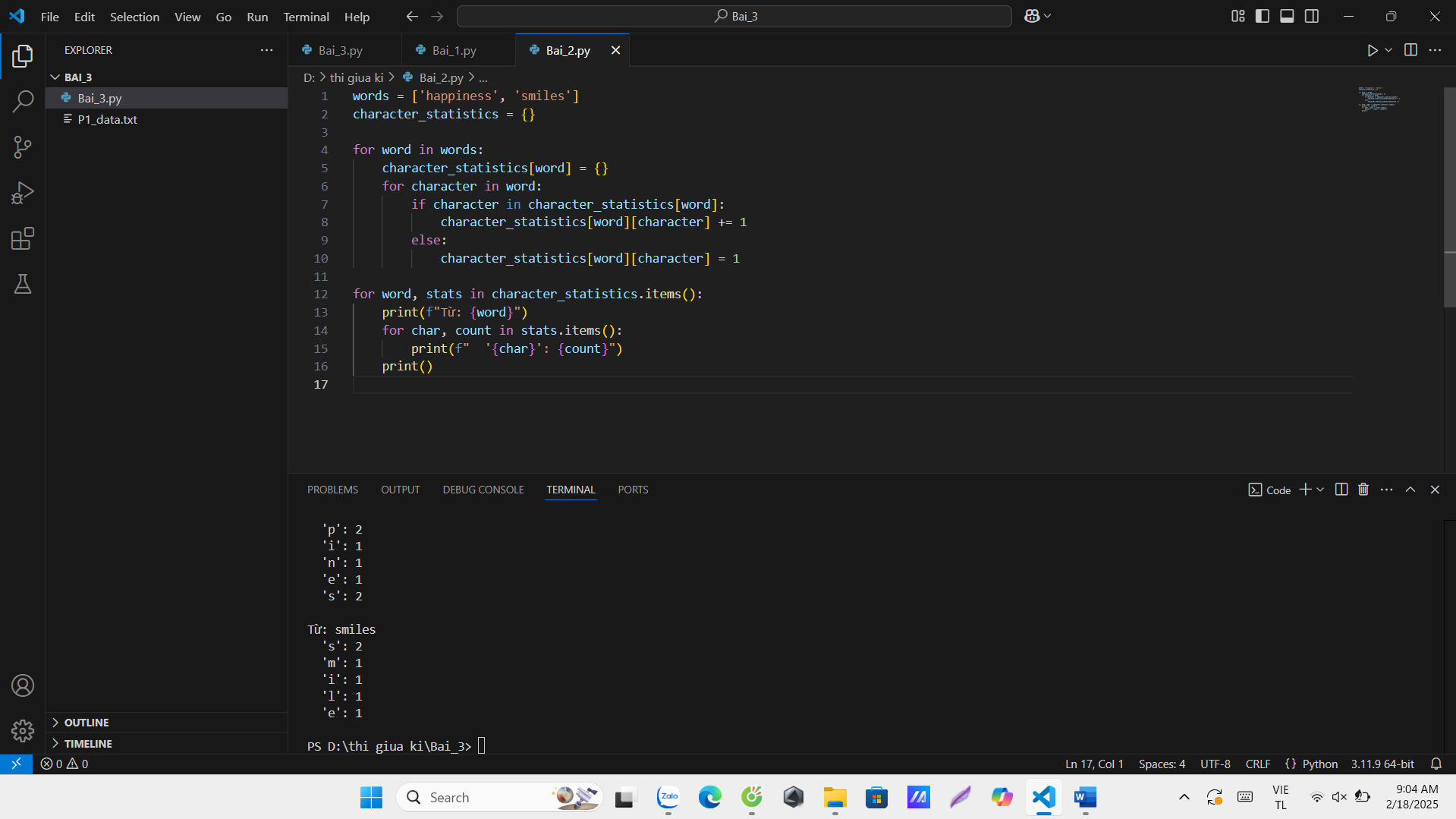
'm': 1

'i': 1

'l': 1

'e': 1

PS D:\thi giua ki\Bai\_3>



BÀI 3:

counter = {}

for sentence in open("P1\_data.txt").read().split():

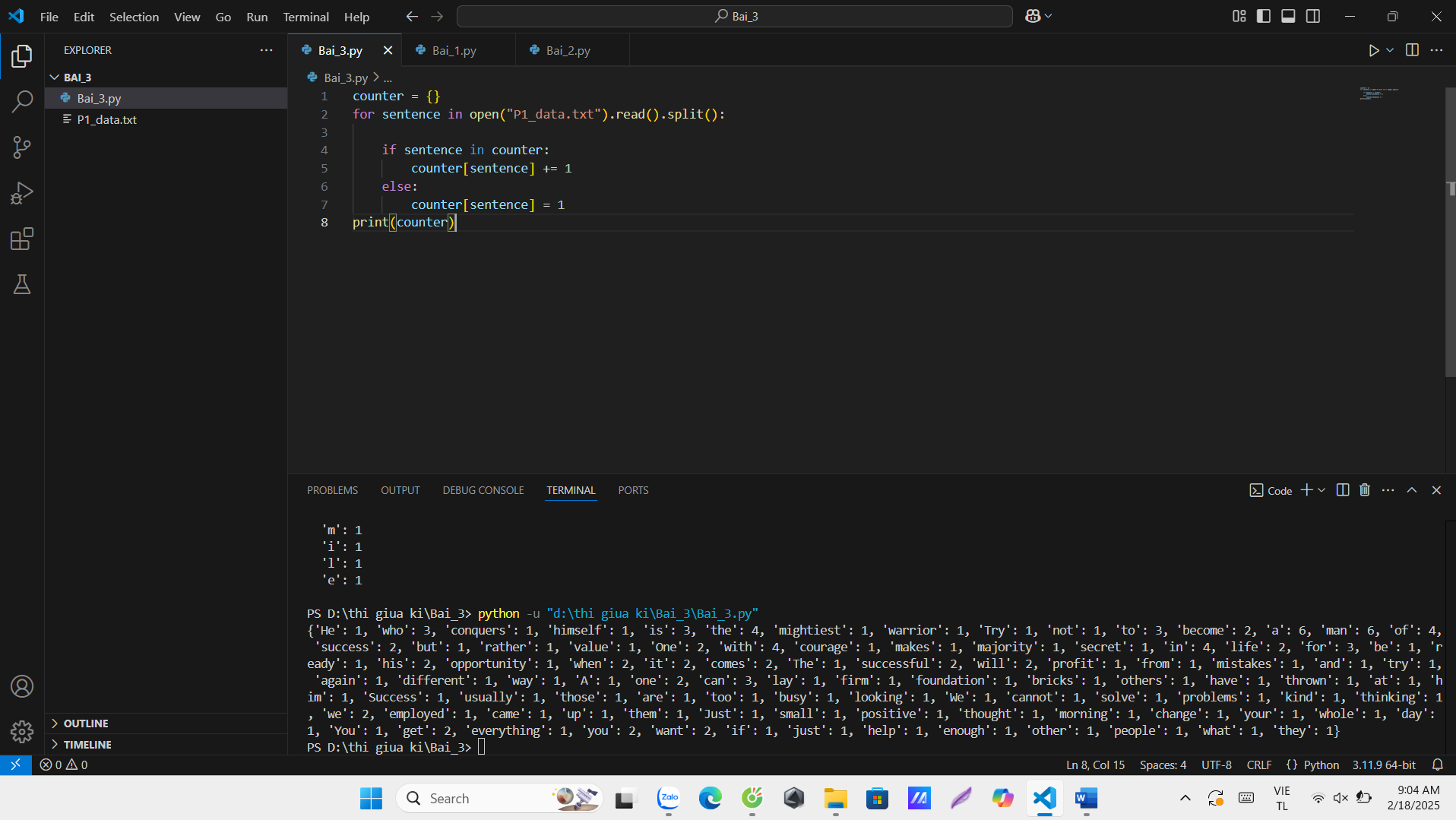
    if sentence in counter:

        counter[sentence] += 1

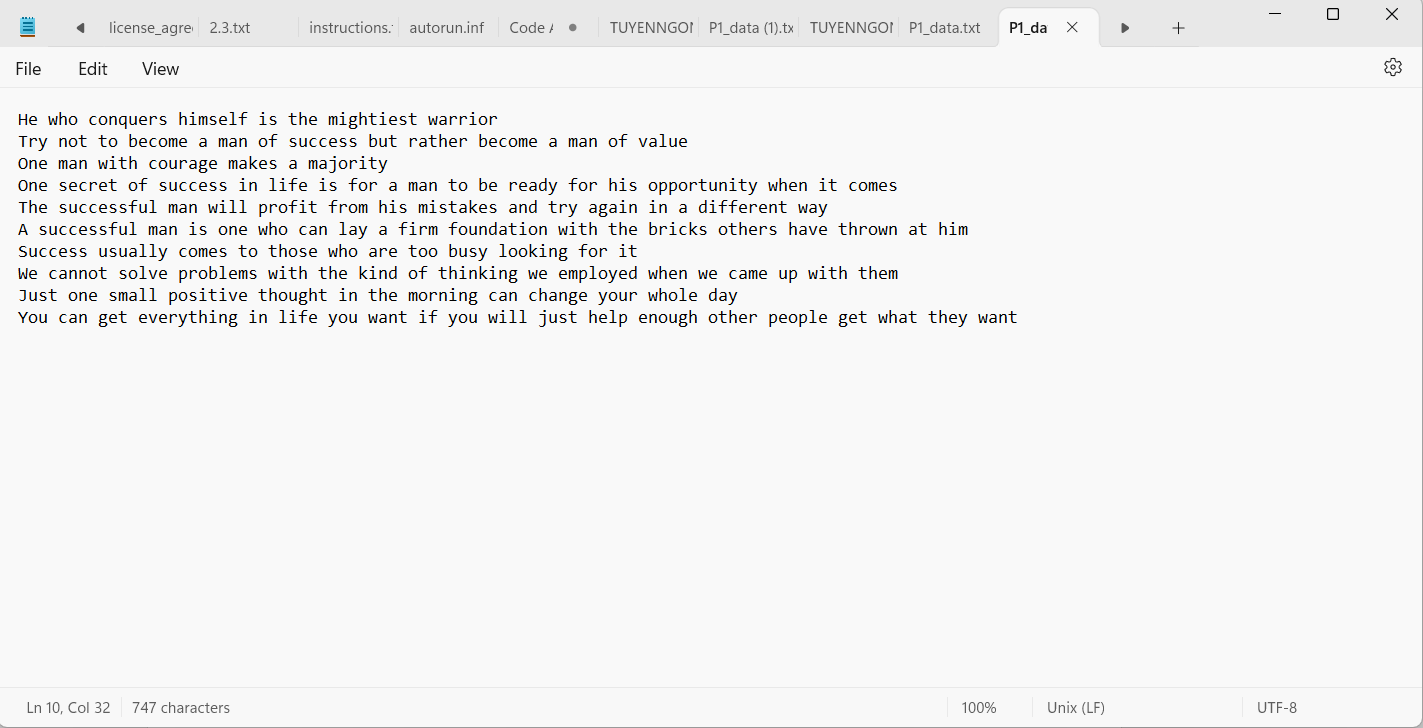
    else:

        counter[sentence] = 1

print(counter)



FILE\_TEXT:



BÀI 4:

def levenshtein\_distance(s, t):

    m, n = len(s), len(t)

    dp = [[0] \* (n + 1) for \_ in range(m + 1)]

    for i in range(m + 1):

        for j in range(n + 1):

            if i == 0:

                dp[i][j] = j

            elif j == 0:

                dp[i][j] = i

            else:

                cost = 0 if s[i-1] == t[j-1] else 1

                dp[i][j] = min(dp[i-1][j] + 1,  # Xóa

                               dp[i][j-1] + 1,  # Chèn

                               dp[i-1][j-1] + cost)  # Thay thế

    return dp[m][n]

# Nhập hai chuỗi từ người dùng

s = input("Nhập chuỗi thứ nhất: ")

t = input("Nhập chuỗi thứ hai: ")

# Tính và hiển thị khoảng cách Levenshtein

print(f"Khoảng cách Levenshtein giữa '{s}' và '{t}' là: {levenshtein\_distance(s, t)}")

KẾT QUẢ  
PS D:\thi giua ki\Bai\_3> python -u "d:\thi giua ki\Bai\_4.py"

Nhập chuỗi thứ nhất: manh

Nhập chuỗi thứ hai: man

Khoảng cách Levenshtein giữa 'manh' và 'man' là: 1

PS D:\thi giua ki\Bai\_3>

