import requests

import re

def sum (first,last):

    sum = 0

    for i in range(first, last+1):

        if i > 1:

            for j in range(2, int(i\*\*0.5) + 1):

                if i % j == 0:

                    break

            else:

                sum += i

    print("Sum of prime numbers in the range is:", sum)

def Unit\_Conversion(value, direction):

    if direction == "M":

        result = value \* 3.28

        return f"{round(result, 2)} F"

    elif direction == "F":

        result = value / 3.28

        return f"{round(result, 2)} M"

    else:

        print("Invalid ,either enter 'M' or 'F'.")

        return None

def Consonant\_Counter(letter):

    consonant = "dcbfghjkmlnpqrstvwxyzBCDFGHJKLMNPQSQRTVWXZY"

    count = 0

    for char in letter:

        if char in consonant:

            count += 1

    print(f"The number of consonants in the string is: {count}")

def min\_max():

    Data = []

    while True:

        value = input("Enter a number or ( type 'stop' to exit): ")

        if value.lower() == 'stop':

            break

        try:

            num = float(value)

            Data.append(num)

        except ValueError:

            print(" enter a valid number, Please.")

    if Data:

        print("The smallest number you entered is:", min(Data))

        print("The greatest number you entered is:", max(Data))

def pali(p):

    new\_name = p.replace(" ", "").lower()

    return new\_name == new\_name[::-1]

def word(https):

    try:

        reply = requests.get(https)

        reply.raise\_for\_status()

        text = reply.text

        target\_words = ["the", "was", "and"]

        words = re.findall(r'\b\w+\b', text.lower())

        counts = {word: words.count(word) for word in target\_words}

        print("Word counts:")

        for word, count in counts.items():

            print(f"{word}: {count}")

    except reply.exceptions.ReplyException as e:

        print(f"Error downloading the file: {e}")

# Display Menu and Handle User Input

def main():

    while True:

        print("\n--- Menu ---")

        print("1. Calculating the Sum of Prime Numbers")

        print("2. Converting the units of light")

        print("3. Count Consonants in String")

        print("4. Finding the Min-Max Numbers")

        print("5. Palindrome Checker")

        print("6. Word Counter (from URL)")

        print("0. Exit")

        choice = input("Select an option (1-6 or 0 to Exit): ")

        if choice == "1":

            first= int(input("Enter the starting range: "))

            last = int(input("Enter the ending range: "))

            sum(first, last)

        elif choice == "2":

            value = float(input("Enter the length value: "))

            direction = input("Enter the length direction ('M' for meters to feet, 'F' for feet to meters): ").upper()

            new\_value = Unit\_Conversion(value, direction)

            if new\_value:

                print(f"Converted value: {new\_value}")

        elif choice == "3":

            Alphabete = input("Enter a text string: ")

            Consonant\_Counter(Alphabete)

        elif choice == "4":

            min\_max()

        elif choice == "5":

            name = input("Enter the string: ")

            print("Is the string a palindrome?", pali(name))

        elif choice == "6":

            https = input("Enter the URL of the text file: ")

            word(https)

        elif choice == "0":

            print("Exiting the program.")

            break

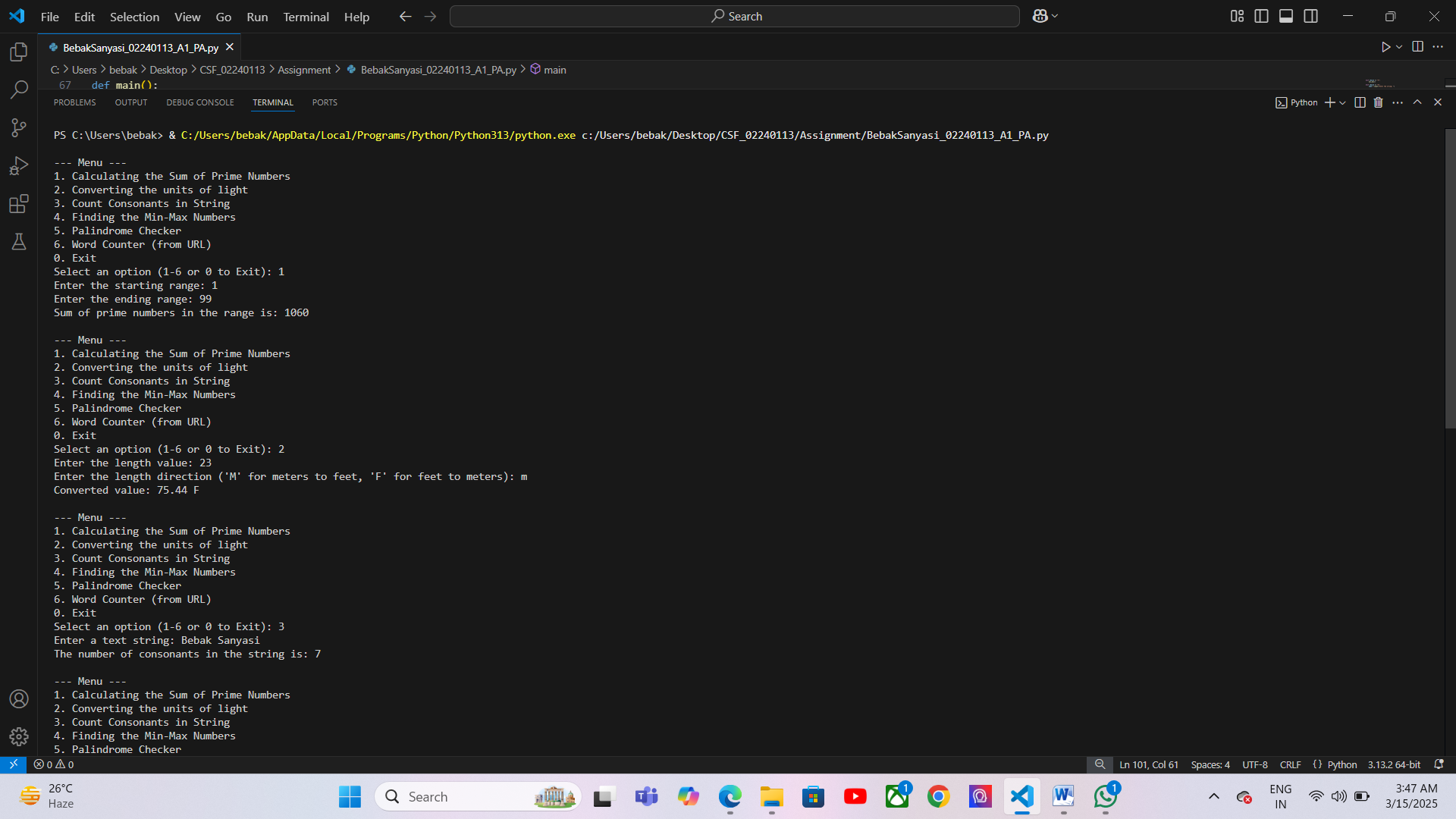
        else:

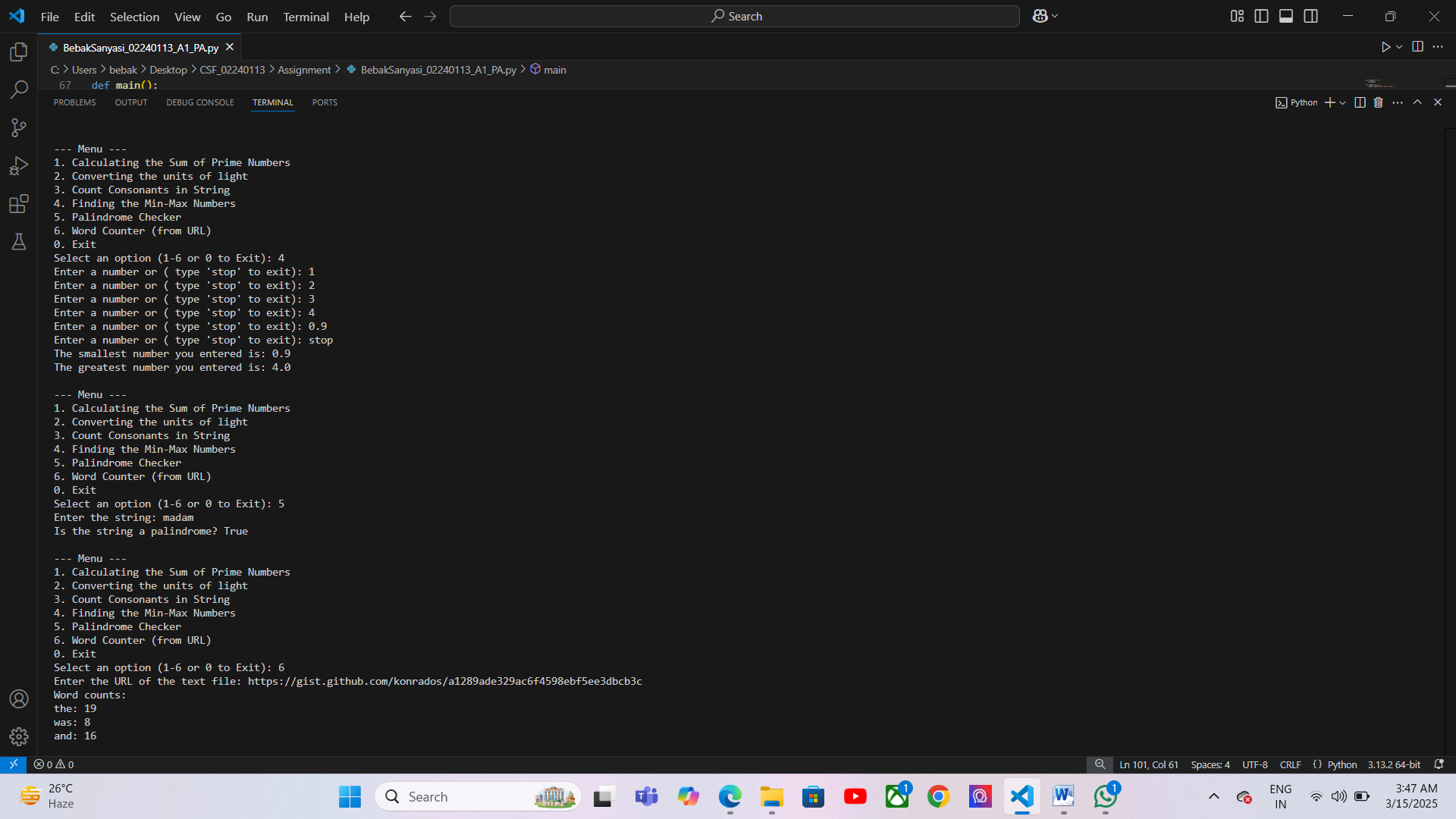
            print("Invalid choice. Please try again.")

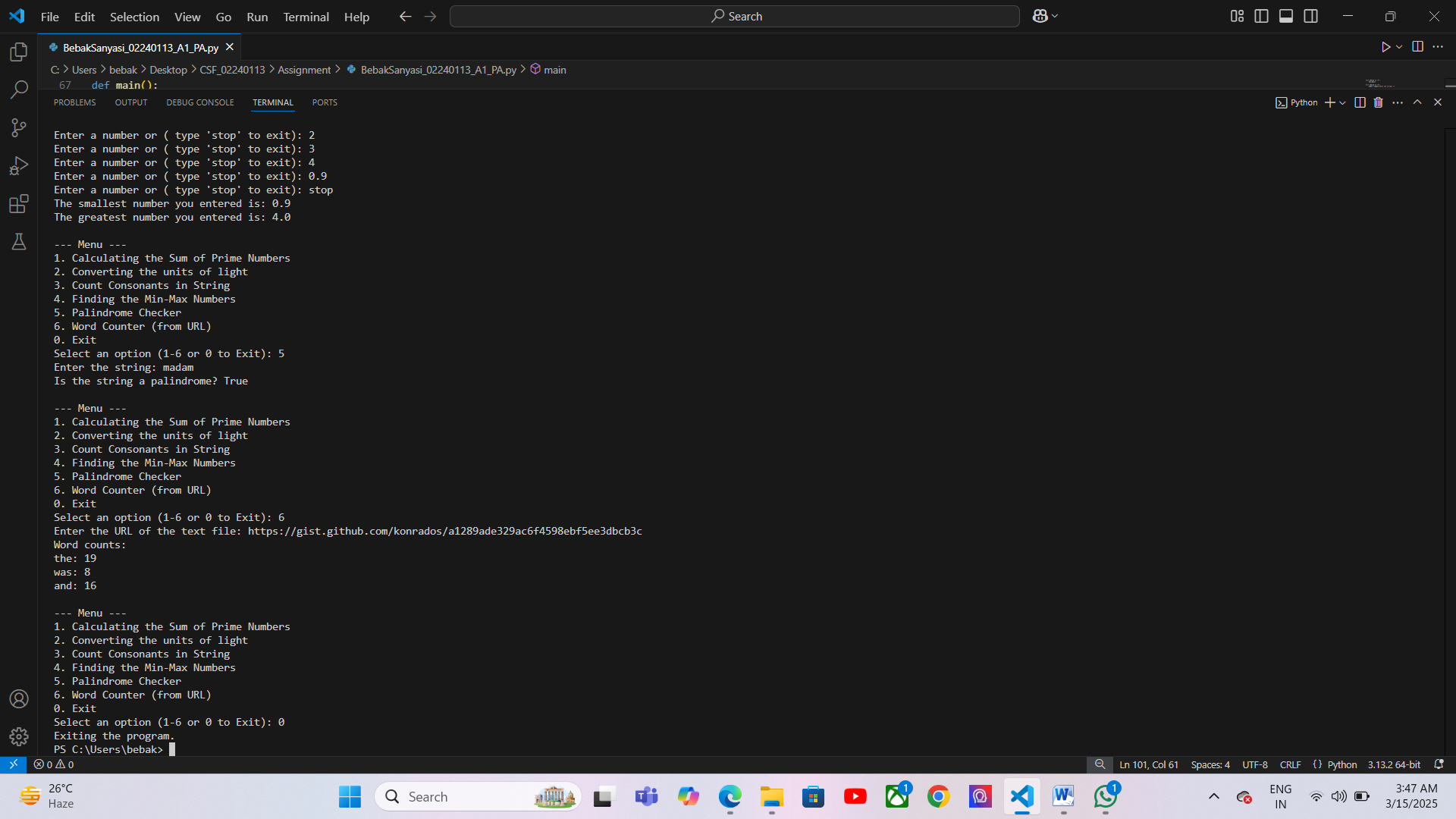
if \_\_name\_\_ == "\_\_main\_\_":

    main()

Following are the output for above coding







Below ones are the coding for the part B of the assignment

import random

def guess\_game():

    print("Guess Number Game")

    correct\_num = 5

    num = int(input("Enter the number from 1 to 10: "))

    if num == correct\_num:

        print("Hurray, your guess is right, wanna play another game?")

        print("Your luck did favour, try your luck in another game.")

    else:

        print("your gess is wrong,sorry.")

        print("Don't worry,try another game!")

def second\_game():

    print("Rock Paper Scissors Game")

    options = ["rock", "paper", "scissors"]

    Player\_choice = input("Enter your choice (rock, paper, or scissors): ").lower()

    ai\_choice = random.choice(options)

    print(f"Computer's choice: {ai\_choice}")

    if Player\_choice == ai\_choice:

        print("It's a draw!")

    elif (Player\_choice == "rock" and ai\_choice == "scissors")or \

     (Player\_choice == "scissors" and ai\_choice == "paper")or \

     (Player\_choice == "paper" and ai\_choice == "rock"):

        print("You win!")

    else:

        print("You lose!")

def main():

    while True:

        game\_choosing = input("Which game would you like to play? (1) Guess Number or (2) Rock Paper Scissors (Enter 1 or 2, or 'quit' to exit): ").lower()

        if game\_choosing == "1":

            guess\_game()

        elif game\_choosing == "2":

            second\_game()

        elif game\_choosing == "leave":

            print("Thank you for playing! bye.")

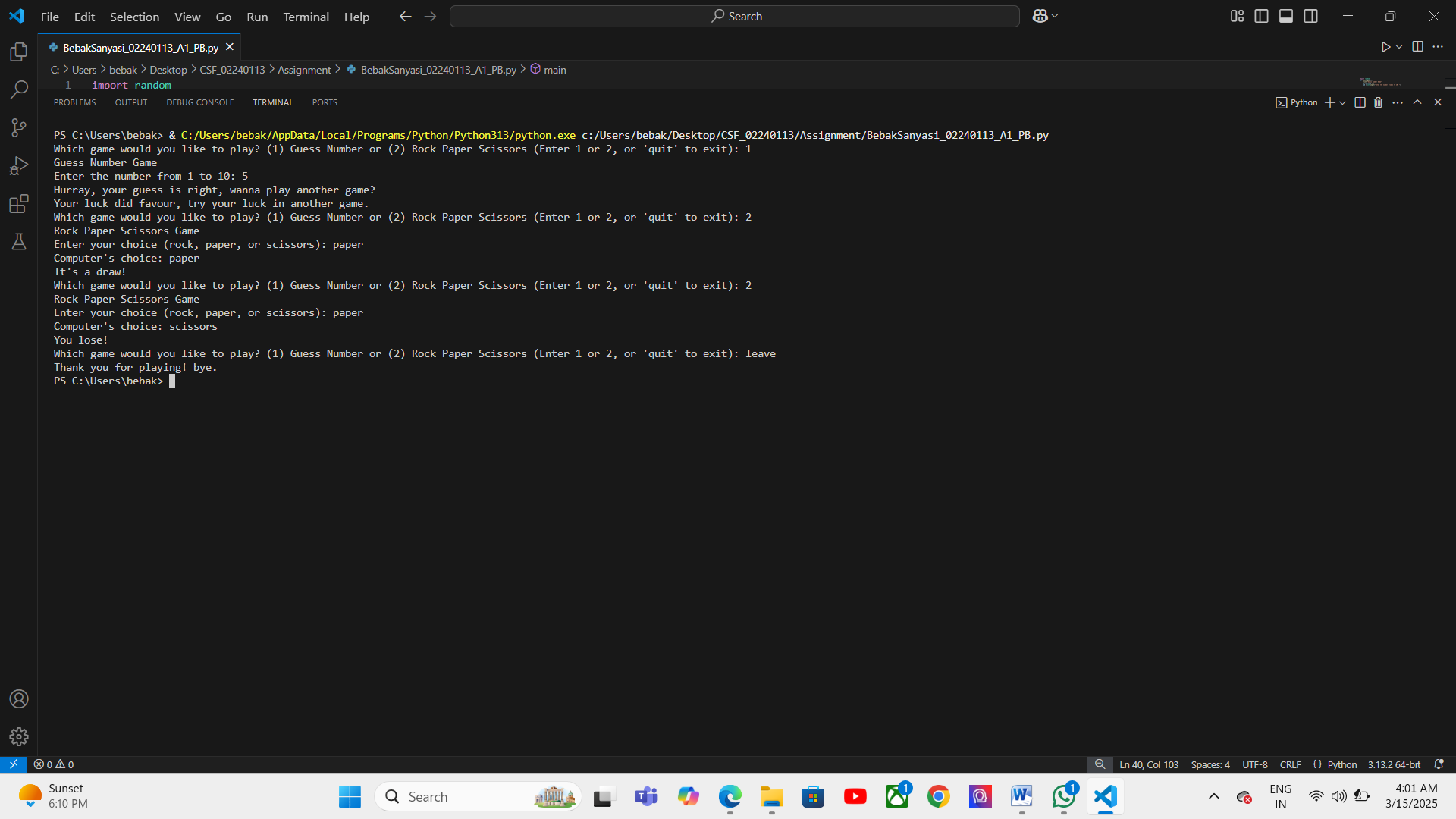
            break

        else:

            print("Invalid choice. Please enter 1 for Guess game and 2 for second game, or type 'leave' to exit.")

main()

below ones are the output for the guess game



Citation

"GeeksforGeeks." (2025). *Python program to find the sum of prime numbers*. Retrieved from <https://www.geeksforgeeks.org/>

"W3Schools." (2025). *Python Rock Paper Scissors game*. Retrieved from <https://www.w3schools.com/>

 Python Software Foundation. *Python 3 Documentation*. 2025, <https://docs.python.org/3/>.

 Requests Contributors. *Requests: HTTP for Humans*. 2025, <https://requests.readthedocs.io/en/latest/>.

 "GeeksforGeeks." "Python Program to Find the Sum of Prime Numbers." 2025, https://www.geeksforgeeks.org/python-program-to-find-the-sum-of-prime-numbers/.

 Downey, Allen B. *Think Python: How to Think Like a Computer Scientist.* O'Reilly Media, 2015.

 **Python Software Foundation**. *Python 3 Documentation*. 2025, <https://docs.python.org/3/>

 **Python Software Foundation**. *random — Generate pseudo-random numbers*. 2025, <https://docs.python.org/3/library/random.html>

 **W3Schools**: *Python Rock Paper Scissors Game.* 2025, https://www.w3schools.com/python/python\_rps.asp

 **GeeksforGeeks**: *Python Program to Create Guess the Number Game.* 2025, https://www.geeksforgeeks.org/python-program-to-create-guess-the-number-game/