Assignment Number: 3.5

NAME: N. Nanda Mukesh

H.T.NO: 2303A51410

BATCH: 24

Question 1: Zero-Shot Prompting (Leap Year Check)

Write a zero-shot prompt to generate a Python function that checks

whether a given year is a leap year.

Task:

• Record the AI-generated code.

• Test with years like 1900, 2000, 2024.

• Identify logical flaws or missing conditions.

```python
#write a python program to check if the given year is leap year or not taking user input
year = int(input("Enter a year: "))
if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
    print(year, "is a leap year.")
else:
    print(year, "is not a leap year.")
#user input is only positive integer handle negative and invalid inputs
try:
    year = int(input("Enter a year: "))
    if year < 0:
        print("Please enter a positive integer for the year.")
    else:
        if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
            print(year, "is a leap year.")
        else:
            print(year, "is not a leap year.")
except ValueError:
    print("Invalid input. Please enter a valid integer for the year.")

```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

.py
Enter a year: 2023
2023 is not a leap year.
Enter a year: 2000
2000 is a leap year.
PS C:\Users\yadav\OneDrive\Desktop\AI-LAB>
```

Question 2: One-Shot Prompting (GCD of Two Numbers)

Write a one-shot prompt with one example to generate a Python

function that finds the Greatest Common Divisor (GCD) of two numbers.

Example:

Input: 12, 18 → Output: 6

Task:

• Compare with a zero-shot solution.

• Analyze algorithm efficiency.

```python
#write a well-commented python program to find GCD of two numbers
#example input 12, 18 output 6
def compute_gcd(a, b):
    # Ensure a and b are non-negative integers
    a = abs(a)
    b = abs(b)
    while b:
        a, b = b, a % b
    return a
try:
    num1 = int(input("Enter first number: "))
    num2 = int(input("Enter second number: "))
    if num1 < 0 or num2 < 0:
        print("Please enter non-negative integers only.")
    else:
        gcd = compute_gcd(num1, num2)
        print("The GCD of", num1, "and", num2, "is:", gcd)
except ValueError:
    print("Invalid input. Please enter valid integers.")

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\yadav\OneDrive\Desktop\AI-LAB> & C:/Users/yadav/AppData/Local/Micr
osoft/WindowsApps/python3.13.exe c:/Users/yadav/OneDrive/Desktop/AI-LAB/assign
.py
Enter first number: 24
Enter second number: 12
The GCD of 24 and 12 is: 12
PS C:\Users\yadav\OneDrive\Desktop\AI-LAB>
```

Question 3: Few-Shot Prompting (LCM Calculation)

Write a few-shot prompt with multiple examples to generate a Python

function that computes the Least Common Multiple (LCM).

Examples:

• Input: 4, 6 → Output: 12

• Input: 5, 10 → Output: 10

• Input: 7, 3 → Output: 21

Task:

• Examine how examples guide formula selection.

• Test edge cases

```python
#generate  a well-commented python program to find LCM of two numbers
#example input 4, 6 output 12
#input: 5, 10 output: 10
#ex input 7, 3 output 21
def compute_gcd(a, b):
    # Function to compute GCD using Euclidean algorithm
    a = abs(a)
    b = abs(b)
    while b:
        a, b = b, a % b
    return a
def compute_lcm(a, b):
    # Function to compute LCM using GCD
    if a == 0 or b == 0:
        return 0
    gcd = compute_gcd(a, b)
    lcm = abs(a * b) // gcd
    return lcm
try:
    num1 = int(input("Enter first number: "))
    num2 = int(input("Enter second number: "))
    if num1 < 0 or num2 < 0:
        print("Please enter non-negative integers only.")
    else:
        lcm = compute_lcm(num1, num2)
        print("The LCM of", num1, "and", num2, "is:", lcm)
except ValueError:
    print("Invalid input. Please enter valid integers.")
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

.py
Enter first number: 7
Enter second number: 3
The LCM of 7 and 3 is: 21
PS C:\Users\yadav\OneDrive\Desktop\AI-LAB>
```

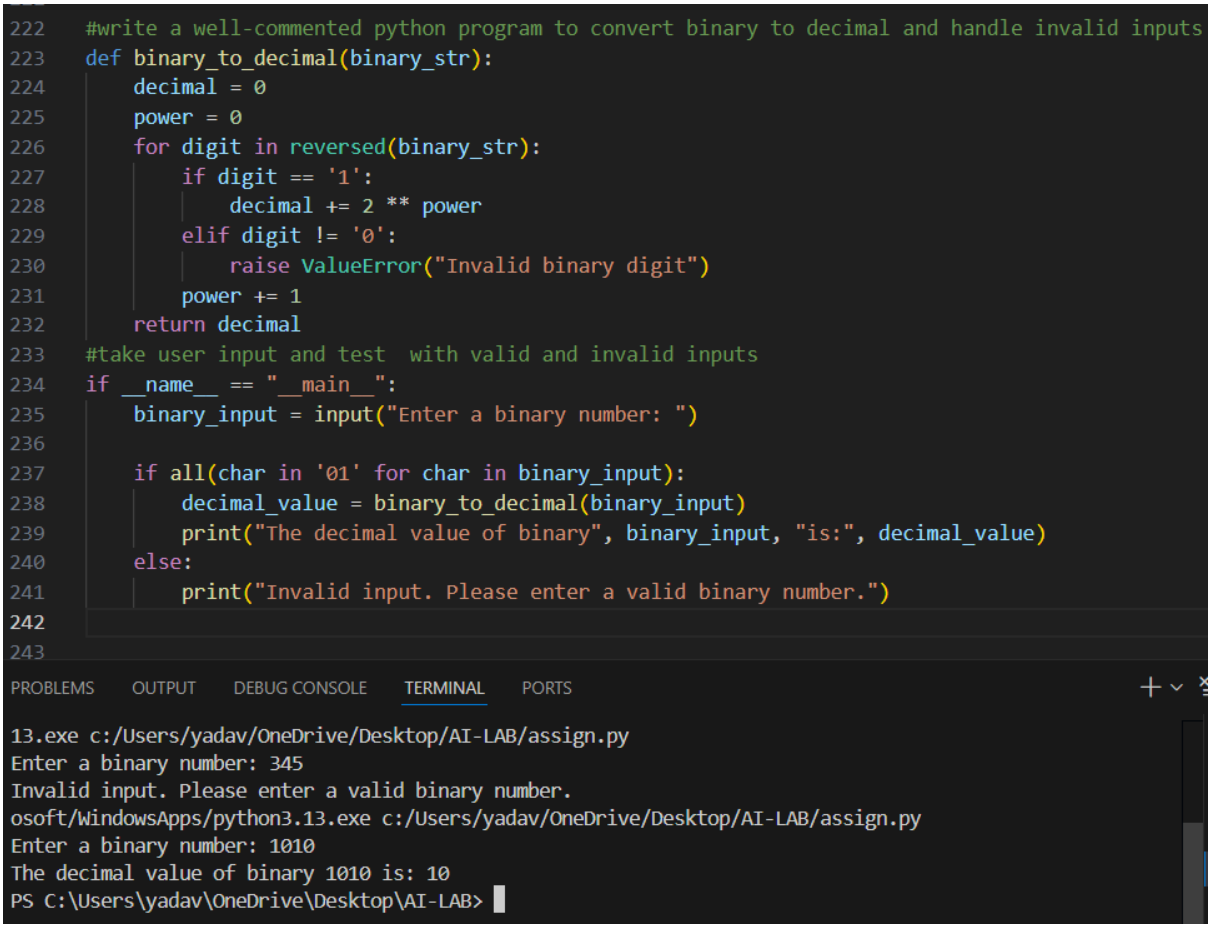
Question 4: Zero-Shot Prompting (Binary to Decimal Conversion)

Write a zero-shot prompt to generate a Python function that converts a binary number to decimal.

Task:

• Test with valid and invalid binary inputs.

• Identify missing validation logic



```python
#write a well-commented python program to convert binary to decimal and handle invalid inputs
def binary_to_decimal(binary_str):
    decimal = 0
    power = 0
    for digit in reversed(binary_str):
        if digit == '1':
            decimal += 2 ** power
        elif digit != '0':
            raise ValueError("Invalid binary digit")
        power += 1
    return decimal
#take user input and test  with valid and invalid inputs
if __name__ == "__main__":
    binary_input = input("Enter a binary number: ")

    if all(char in '01' for char in binary_input):
        decimal_value = binary_to_decimal(binary_input)
        print("The decimal value of binary", binary_input, "is:", decimal_value)
    else:
        print("Invalid input. Please enter a valid binary number.")
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
13.exe c:/Users/yadav/OneDrive/Desktop/AI-LAB/assign.py
Enter a binary number: 345
Invalid input. Please enter a valid binary number.
osoft/WindowsApps/python3.13.exe c:/Users/yadav/OneDrive/Desktop/AI-LAB/assign.py
Enter a binary number: 1010
The decimal value of binary 1010 is: 10
PS C:\Users\yadav\OneDrive\Desktop\AI-LAB>
```

Question 5: One-Shot Prompting (Decimal to Binary Conversion)

Write a one-shot prompt with an example to generate a Python function that converts a decimal number to binary.

Example:

Input: 10 → Output: 1010

Task:

• Compare clarity with zero-shot output.

• Analyze handling of zero and negative numbers.

```python
#write a well-commented python program to convert decimal to binary
#example input 10 output 1010
def decimal_to_binary(n):
    if n < 0:
        raise ValueError("Input must be a non-negative integer.")
    binary_str = ""
    if n == 0:
        return "0"
    while n > 0:
        binary_str = str(n % 2) + binary_str
        n //= 2
    return binary_str
try:
    decimal_input = int(input("Enter a decimal number: "))
    if decimal_input < 0:
        print("Please enter a non-negative integer.")
    else:
        binary_output = decimal_to_binary(decimal_input)
        print("The binary representation of", decimal_input, "is:", binary_output)
except ValueError:
    print("Invalid input. Please enter a valid non-negative integer.")
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\yadav\OneDrive\Desktop\AI-LAB> & C:/Users/yadav/AppData/Local/Microsoft/WindowsApps/python3.
/Users/yadav/OneDrive/Desktop/AI-LAB/assign.py
Enter a decimal number: 1243454
The binary representation of 1243454 is: 100101111100100111110
PS C:\Users\yadav\OneDrive\Desktop\AI-LAB> & C:/Users/yadav/AppData/Local/Microsoft/WindowsApps/python3.
/Users/yadav/OneDrive/Desktop/AI-LAB/assign.py
Enter a decimal number: r536
Invalid input. Please enter a valid non-negative integer.
PS C:\Users\yadav\OneDrive\Desktop\AI-LAB>
```

Question 6: Few-Shot Prompting (Harshad Number Check)

Write a few-shot prompt to generate a Python function that checks

whether a number is a Harshad (Niven) number.

Examples:

• Input: 18 → Output: Harshad Number

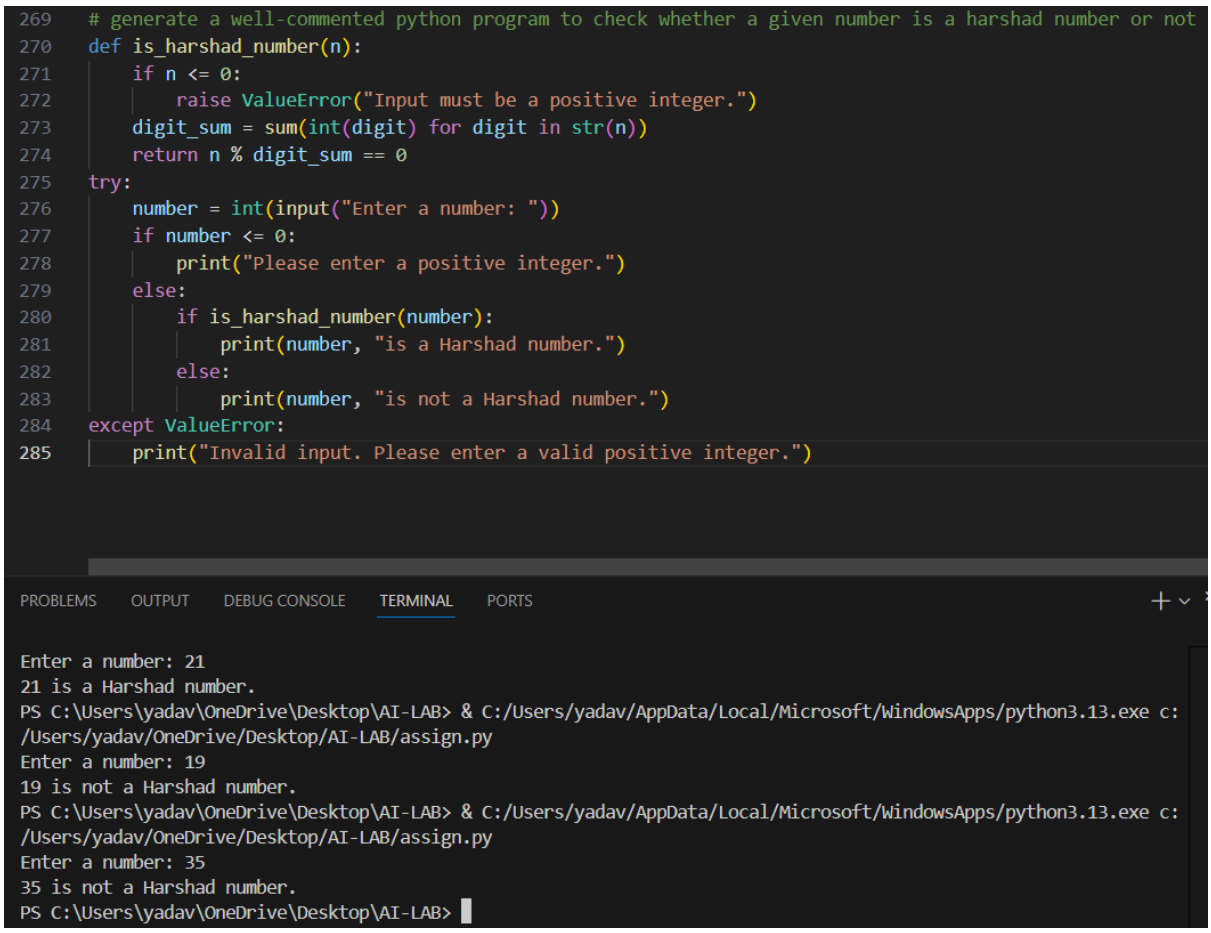
• Input: 21 → Output: Harshad Number

• Input: 19 → Output: Not a Harshad Number

Task:

• Test boundary conditions.

• Evaluate robustness



```python
# generate a well-commented python program to check whether a given number is a harshad number or not
def is_harshad_number(n):
    if n <= 0:
        raise ValueError("Input must be a positive integer.")
    digit_sum = sum(int(digit) for digit in str(n))
    return n % digit_sum == 0
try:
    number = int(input("Enter a number: "))
    if number <= 0:
        print("Please enter a positive integer.")
    else:
        if is_harshad_number(number):
            print(number, "is a Harshad number.")
        else:
            print(number, "is not a Harshad number.")
except ValueError:
    print("Invalid input. Please enter a valid positive integer.")
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Enter a number: 21
21 is a Harshad number.
PS C:\Users\yadav\OneDrive\Desktop\AI-LAB> & C:/Users/yadav/AppData/Local/Microsoft/WindowsApps/python3.13.exe c:
/Users/yadav/OneDrive/Desktop/AI-LAB/assign.py
Enter a number: 19
19 is not a Harshad number.
PS C:\Users\yadav\OneDrive\Desktop\AI-LAB> & C:/Users/yadav/AppData/Local/Microsoft/WindowsApps/python3.13.exe c:
/Users/yadav/OneDrive/Desktop/AI-LAB/assign.py
Enter a number: 35
35 is not a Harshad number.
PS C:\Users\yadav\OneDrive\Desktop\AI-LAB>
```