

AI ASSISTED CODING

LAB ASSIGNMENT 3.5

E.Rishi

2303A51890

BATCH - 04

Platform Used: Git hub co-pilot

ZERO SHOT

ONE SHOT

FEW SHOT

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.

PROMPT

Write a Python function that checks whether a given year is a leap year. The function should return True if it is a leap year and False otherwise.

CODE

```
3.5.py > ...
1  def is_leap_year(year):
2      if (year % 400 == 0) or (year % 4 == 0 and year % 100 != 0):
3          return True
4      else:
5          return False
6
7  print(is_leap_year(2000))
8
```

OUTPUT

```
PS D:\3-2 SEM\AI ASSISTED CODING - 1221
True
PS D:\3-2 SEM\AI ASSISTED CODING - 1221
True
PS D:\3-2 SEM\AI ASSISTED CODING - 1221
```

EXPLANATION

A leap year is divisible by 400, or divisible by 4 but not by 100.

The function checks these conditions and returns the appropriate boolean value.

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

PROMPT

Write a Python function to find the Greatest Common Divisor (GCD) of two numbers.

Example:

Input: 12, 18

Output: 6

CODE

```
12  def gcd(a, b):
13  |     while b != 0:
14  |         a, b = b, a % b
15  |     return a
16  print(gcd(15,20))
17
18
```

OUTPUT

```
PS D:\3-2 SEM\AI ASSISTED CODING
5
```

```
● PS D:\3-2 SEM\AI ASSISTED CODING
```

EXPLANATION

This function uses the **Euclidean Algorithm**, which repeatedly replaces the larger number with the remainder until the remainder becomes zero.

The last non-zero value is the GCD.

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.

PROMPT

Write a Python function to compute the Least Common Multiple (LCM) of two numbers

Examples:

Input: 4, 6 → Output: 12

Input: 5, 10 → Output: 10

Input: 7, 3 → Output: 21

CODE

```
21  def lcm(a, b):
22      def gcd(x, y):
23          while y:
24              x, y = y, x % y
25          return x
26      return abs(a * b) // gcd(a, b)
27
28  print(lcm(4, 6))    # 12
29  print(lcm(5, 10))   # 10
30  print(lcm(7, 3))    # 21
31
```

OUTPUT

```
PS D:\3-2 SEM\AI ASSISTED CODING - 1221
12
10
21
PS D:\3-2 SEM\AI ASSISTED CODING - 1221
```

EXPLANATION

LCM is calculated using the formula:

$$\text{LCM}(a, b) = (a \times b) / \text{GCD}(a, b).$$

The examples guide the AI to choose this mathematical formula.

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.

PROMPT

Write a Python function that converts a binary number into its decimal equivalent.

CODE

```
def binary_to_decimal(binary):  
    decimal = 0  
    for digit in binary:  
        decimal = decimal * 2 + int(digit)  
    return decimal  
print(binary_to_decimal("1010")) # 10  
print(binary_to_decimal("111")) # 7
```

OUTPUT

```
PS D:\3-2 SEM\AI ASSISTED CODING  
10  
7  
PS D:\3-2 SEM\AI ASSISTED CODING
```

EXPLANATION

Each binary digit is processed from left to right.

The current value is multiplied by 2 and the digit is added, converting binary to decimal.

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.

PROMPT

Write a Python function to convert a decimal number to binary.

Example:

Input: 10

Output: 1010

CODE

```
def decimal_to_binary(n):
    if n == 0:
        return "0"

    binary = ""
    while n > 0:
        binary = str(n % 2) + binary
        n //= 2
    return binary
print(decimal_to_binary(10)) # 1010
```

OUTPUT

```
PS D:\3-2 SEM\AI ASSISTED CODING
1010
PS D:\3-2 SEM\AI ASSISTED CODING
```

EXPLANATION

The decimal number is repeatedly divided by 2.

The remainders are collected in reverse order to form the binary representation.

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

PROMPT

Write a Python function to check 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

CODE

```
def is_harshad(n):  
    digit_sum = sum(int(d) for d in str(n))  
    if n % digit_sum == 0:  
        return "Harshad Number"  
    else:  
        return "Not a Harshad Number"  
  
print(is_harshad(18)) # Harshad Number  
print(is_harshad(21)) # Harshad Number  
print(is_harshad(19)) # Not a Harshad Number
```

OUTPUT

```
PS D:\3-2 SEM\AI ASSISTED CODING - 1221>
Harshad Number
Harshad Number
Not a Harshad Number
PS D:\3-2 SEM\AI ASSISTED CODING - 1221>
```

EXPLANATION

A Harshad number is divisible by the sum of its digits.

The function calculates the digit sum and checks divisibility.