# 📘 Lab Report

Name: **HIDAYAT ULLAH**

R No: **B24F0050AI099**

Lab No: **02**

Date: **03/10/2025**

Submitted to: **Engineer ABDULLAH SAJID**

## ⌄ Task 1: Vowel Encoding (Data Preprocessing for NLP)

### Scenario:

In natural language processing (NLP), text data often undergoes preprocessing steps like encoding specific characters. You are tasked with encoding vowels in a string by replacing them with numbers: a becomes 1, e becomes 2, i becomes 3, o becomes 4, and u becomes 5. This helps in simple data transformation for further analysis.

### Input:

A string entered by the user.

### Output:

The encoded string where all vowels are replaced by their corresponding numbers.

### Example:

• Input: "hello world" • Output: "h2ll4 w4rld"

### Hints:

• Use the replace() method multiple times for each vowel. • Remember that strings are immutable, so each replace() returns a new string. You can chain these methods or assign the result to a variable each time. • Consider the order of replacement; since the replacements are distinct, order may not matter here.

```python
text = input("Enter a string: ")

encoded_text = (text.replace("a", "1")
                    .replace("e", "2")
                    .replace("i", "3")
                    .replace("o", "4")
                    .replace("u", "5"))

print("Encoded string:", encoded_text)


Enter a string: Hello word
Encoded string: H2ll4 w4rd
```

## ⌄ *Description*

User writes a string and saves it in a variable called text.

The program changes vowels into numbers:

a → 1

e → 2

i → 3

o → 4

u → 5. It makes a new string with these changes and saves it in encoded_text.

Finally, it prints the new encoded string.

### Example:

"hello world" becomes "h2ll4 w4rld".

⌄

⌄

_____

## ⌄ Task 2: Dataset Merging and Analysis (AI Data Management)

### *Scenario:*

In AI, data often comes from multiple sources. You have two lists of numerical values from different experiments. Your task is to merge these lists into a single dataset, sort it to find the overall trend, and identify the highest and lowest values for outlier detection.

### *Input:*

Two lines of input, each containing numbers separated by spaces.

• First line: numbers for List 1.

• Second line: numbers for List 2.

```python
list1 = list(map(int, input("Enter numbers for List 1: ").split()))
list2 = list(map(int, input("Enter numbers for List 2: ").split()))

merged_list = list1 + list2
merged_list.sort()

print("Merged and Sorted List:", merged_list)
print("Lowest Value:", merged_list[0])
print("Highest Value:", merged_list[-1])
```

```
Enter numbers for List 1: 1 3 7 2 4
Enter numbers for List 2: 9 8 12 5 6
Merged and Sorted List: [1, 2, 3, 4, 5, 6, 7, 8, 9, 12]
Lowest Value: 1
Highest Value: 12
```

## ⌄  *Description*

This program takes two lists of numbers from the user.
First, it takes List 1. Then it takes List 2.
It joins both lists into one.
Then it sorts the list from smallest to biggest.
After sorting, it shows the lowest value and the highest value.

⌄

---

## ⌄  Task 3: `Text Statistics (NLP Preprocessing)`

### *Scenario:*

Before feeding text data into an AI model, it is common to compute basic statistics. Given a text string, compute the number of words, the number of characters (with and without space), and convert the text to uppercase for consistency.

### `Input:`

A string entered by the user.

### *Output:*

The number of words, number of characters including spaces, number of characters excluding spaces, and the uppercase version of the string.

```python
taxt = input("enter your taxt: ")

words = taxt.split()
NumberWords = len(words)

Number_of_character_with_space = len(taxt)
Number_of_character_without_space = len(taxt.replace(" ", ""))

uppercase_text = taxt.upper()

print("---RESULTS---")
print("Number of words:", NumberWords)
print("Characters (with spaces):", Number_of_character_with_space)
print("Characters (without spaces):", Number_of_character_without_space)
print("Uppercase version:", uppercase_text)
```

```
enter your taxt: Hello world! AI is fun.
---RESULTS---
Number of words: 5
Characters (with spaces): 23
Characters (without spaces): 19
Uppercase version: HELLO WORLD! AI IS FUN.
```

## Description:

This program takes text from the user.
It finds how many words are in the text.
It finds how many characters with spaces.
It finds how many characters without spaces.
It changes the text into uppercase letters.
Finally, it prints all these results.

## Summary

In this lab we did 3 main tasks.

In Task 1, we worked on string preprocessing where we change vowels into numbers (a=1, e=2, i=3, o=4, u=5). This was to practice string replace and to see immutability.

In Task 2, we merged two lists of numbers, then sorted them, and found the lowest and highest values. This helped us to learn list merging, sorting, and outlier checking.

In Task 3, we calculated text statistics like number of words, number of characters (with and without space), and also changed the text into uppercase. This was for practice of split, len, replace, and upper.

So overall in these tasks we practiced both string and list operations and how they are used in AI preprocessing and data handling.