# PART – A

## Task 1 : Spelling Correction Application

### Objective:

Develop a Spelling Correction application that provides users with the ability to input misspelled words or upload text files for automatic correction. The application should include both a web-based interface and backend processing using Flask and Python libraries.

### Requirements:

**Web Interface (3 Marks)**

**Front-end Development:**

* Create a web-based front-end using HTML and JavaScript.
* Include an input field for users to manually enter misspelled words.
* Provide a file upload option allowing users to upload a text file for batch correction.

**User Input:**

* Display the corrected word immediately when manually entered.
* Correct spelling mistakes in the uploaded text file and generate a corrected version.

**Spelling Correction Application (3 Marks)**

**Backend Implementation:**

1. Use Flask to develop the backend for handling user requests and responses.

2. Use any Python library, to create spelling correction with given text as corpus.

3. Use text from below source as corpus for calculating probabilities.

4. Use the 'Research Paper Text' column in the alldata\_1\_for\_kaggle.csv available in this link: https://www.kaggle.com/datasets/falgunipatel19/biomedical-text-publication-classification

**Integration: (2 Marks)**

* 1. Integrate the front-end and back-end components to ensure seamless functionality.
  2. Process user input effectively, perform necessary corrections, and present results in a clear and user-friendly manner on the web page.

## Task 2 : Enhancement Plan : 2 Marks

Provide a detailed documentation outlining the step-by-step process to enhance your existing Python-based spelling correction application with phonetic similarity algorithms. Discuss implementation details, algorithm selection, and performance evaluation criteria for integrating phonetic-based corrections.

# 

# 

# 

# 

# 

# 

# 

# 

# 

# PART – B

**Objective**: Conduct a literature survey to explore the evolution of techniques used for part-of-speech tagging in natural language processing (NLP).

**Requirements:**

1. **Research Historical Approaches (2 Marks)**:
   * Investigate early approaches to POS tagging, such as rule-based methods (e.g., handwritten linguistic rules) and statistical models (e.g., Hidden Markov Models (HMMs) and Maximum Entropy Models).
   * Summarize their key principles, methodologies, and limitations.
2. **Survey Modern Approaches (2 Marks)**:
   * Explore advancements in POS tagging using machine learning and deep learning techniques.
   * Discuss methods like Conditional Random Fields (CRFs), BiLSTMs, and transformer-based models (e.g., BERT, GPT).
   * Highlight how these methods address challenges like ambiguity and out-of-vocabulary words.
3. **Critical Analysis (1 Mark)**:
   * Compare historical and modern techniques.
   * Highlight trade-offs between complexity, accuracy, and computational requirements in different approaches.

--------------------------------------------------------------------------------------------------------------------------

# Deliverables:

**PART - A**

* A well-documented code (Python and frontend) for the Spelling Correction application.
* Instructions for running the application locally.
* A set screenshots that explains the entire flow of the application to be included in the report for all input cases.
* Task-2 to be submitted as a .pdf document.

**PART – B**

* Submit 2-3 page well-documented literature review completing all the tasks in PART - B as a **single .pdf document only**