Assignment 2: Word Length Histogram with MapReduce

Topic: MapReduce Examples (Lecture 9)
Time: 30-40 minutes
Objective: Categorize words in a text by length and count them using a MapReduce-like approach.

Problem Statement:
Given a text dataset, categorize each word into "Tiny" (1 letter), "Small" (2-4 letters), "Medium" (5-9 letters), or "Big" (10+ letters) and count the occurrences in each category. Simulate this using a MapReduce process in Jupyter Notebook and visualize the results.

Dataset:
A text string, e.g.,

python

```
text = "a cat runs fast and the elephant is enormous in this vast world"
```

Requirements:

- Implement map_function to emit (category, 1) pairs based on word length.
- Implement reduce_function to sum counts per category.
- Output a dictionary with categories and their counts.
- Create a bar chart with matplotlib.

Solution Outline:

1. Define a helper function to categorize word length.
2. map_function: Split text into words, categorize each, and emit (category, 1).
3. reduce_function: Aggregate counts by category.
4. Process the text and apply MapReduce simulation.
5. Plot the results.

Sample Code Starter:

python
CollapseWrapCopy

```
import matplotlib.pyplot as plt


text = "a cat runs fast and the elephant is enormous in this vast world"


def categorize_word(word):
    length = len(word)
```

```
    if length == 1:
        return "Tiny"
    elif 2 <= length <= 4:
        return "Small"
    elif 5 <= length <= 9:
        return "Medium"
    else:
        return "Big"

def map_function(text_chunk):
    # Students implement
    pass

def reduce_function(mapped_data):
    # Students implement
    pass

# Students complete the rest
```