# Advanced Python String Questions

## Basic String Operations

1. Write a function to reverse a string without using slicing (`[::-1]`).

2. Write a function that checks if a string is a palindrome (case-insensitive).

3. How do you remove duplicate characters from a string while maintaining order?

4. Convert 'hello world' to 'Hello World' without using `.title()`.

5. Implement a function that counts vowels and consonants in a string.

## String Formatting & Manipulation

6. Format a string to show a decimal number with 2 decimal places using `format()` and `f-strings`.

7. Replace multiple whitespaces with a single space in a string.

8. Convert a string 'abc123xyz' to 'ABC123XYZ' using `.translate()` instead of `.upper()`.

9. Write a function that splits a string into a list of words without using `.split()`.

10. Find and extract all email addresses from a given text using regex.

## Advanced String Searching

11. Given a sentence, find the word that occurs most frequently.

12. Find all occurrences of a substring in a given string (without using `.count()`).

13. Implement a function that finds the longest common prefix in a list of strings.

14. Check if a given string is a valid IPv4 address (e.g., '192.168.0.1').

15. Write a function to find the longest palindromic substring in a given string.

## Encoding & Decoding

16. Convert a string to Base64 encoding and decode it back.

17. Convert 'hello' to its ASCII values list and back to a string.

18. Implement a function that encrypts a string using Caesar Cipher with a shift of `k`.

19. Compress a string using Run-Length Encoding (RLE) (e.g., 'aaabbc' → 'a3b2c1').

20. Decode an RLE-encoded string back to its original form.

## Regular Expressions & Parsing

21. Extract all numbers from a string using regex.

22. Write a function to find hashtags (e.g., extract `#Python` from 'I love #Python!').

23. Extract and format dates (convert '12/05/2024' to 'May 12, 2024').

24. Validate if a string is a valid hexadecimal color code (e.g., `#AABBCC`).

25. Extract all words of length 4 from a given paragraph.

## Miscellaneous Challenges

26. Implement `strstr()` (find the index of a substring in a string without using `.find()` or `.index()`).

27. Convert a given camelCase string to snake\_case (e.g., 'helloWorld' → 'hello\_world').

28. Given a scrambled string where characters are shuffled, find the original word (you have a dictionary to match with).

29. Write a function that removes duplicate words from a sentence.

30. Implement a function to check if two strings are anagrams of each other.

Extra Learning Concepts Covered

- String searching algorithms

- Regex patterns for data extraction

- Encoding & Decoding techniques

- Optimization of string operations

- Using built-in string functions creatively