

Assignment-65

String Algorithm

1. Given a list of words followed by two words, the task is to find the minimum distance between the given two words in the list of words.

Example :

Input: S = { "the", "quick", "brown", "fox", "quick"}, word1 = "the", word2 = "fox"

Output: 3

Explanation: Minimum distance between the words "the" and "fox" is 3

2. Given a string Str. The task is to check if it is Pangram or not.

A pangram is a sentence containing every letter in the English Alphabet.

Examples:

Input: "The quick brown fox jumps over the lazy dog"

Output: is a Pangram

Explanation: Contains all the characters from 'a' to 'z']

Input: "The quick brown fox jumps over the dog"

Output: is not a Pangram

Explanation: Doesn't contain all the characters from 'a' to 'z', as 'l', 'z', 'y' are missing

3. Given two strings, the task is to find if a string can be obtained by rotating another string two places.

Example :

Input: string1 = "amazon", string2 = "azonam"

Output: Yes

Explanation: Rotating string1 by 2 places in anti-clockwise gives the string2.

4. Given a dictionary and a string 'str', find the longest string in the dictionary which can be formed by deleting some characters of the given 'str'.

Examples:

Input : dict = {"ale", "apple", "monkey", "plea"}

str = "abpcplea"

Output : apple

5. Given two binary strings, return their sum (also a binary string).

Example:

Input: a = "11", b = "1"
Output: "100"

6. Given a string s, find the length of the longest prefix, which is also a suffix. The prefix and suffix should not overlap.

Examples:

Input : aabcdaabc
Output : 4
The string "aabc" is the longest
prefix which is also a suffix.

Input : abcab
Output : 2

Input : aaaa
Output : 2

7. Given a string S consisting of N, lower case English alphabet, it is also given that a string is encrypted by first replacing every substring of the string consisting of the same character with the concatenation of that character and the hexadecimal representation of the size of the substring and then reversing the whole string, the task is to find the encrypted string.

Note: All Hexadecimal letters should be converted to Lowercase letters.

Examples:

Input: S = "aaaaaaaaaaa"
Output: ba
Explanation:

1. First convert the given string to "a11" i.e. write character along with its frequency.
 2. Then, change "a11" to "ab" because 11 is b in hexadecimal.
 3. Then, finally reverse the string i.e "ba".
8. Given a string of brackets, the task is to find an index k which decides the number of opening brackets is equal to the number of closing brackets.
The string must consist of only opening and closing brackets i.e. '(' and ')'.

An equal point is an index such that the number of opening brackets before it is equal to the number of closing brackets from and after.

Examples:

Input: str = "((()))(

Output: 4

Explanation: After index 4, string splits into (()) and))(. The number of opening brackets in the first part is equal to the number of closing brackets in the second part.

9. Given a string S that consists of only alphanumeric characters and dashes. The string is separated into N + 1 groups by N dashes. Also given an integer K.

We want to reformat the string S, such that each group contains exactly K characters, except for the first group, which could be shorter than K but still must contain at least one character.

Furthermore, a dash must be inserted between two groups, and you should convert all lowercase letters to uppercase.

Return the reformatted string.

Example :

Input: S = "5F3Z-2e-9-w", K = 4

Output: "5F3Z-2E9W"

Explanation: The string S has been split into two parts, each part has 4 characters.

Note that two extra dashes are not needed and can be removed.

Input: S = "2-5g-3-J", K = 2

Output: "2-5G-3J"

Explanation: The string s has been split into three parts, each part has 2 characters except the first part as it could be shorter as mentioned above

10. Given two strings A and B. The task is to find the minimum number of times A has to be repeated such that B is a substring of it. If no such solution exists, print -1.

Example :

Input : A = "abcd", B = "cdababcdab"

Output : 3

Repeating A three times ("abcdabcdabcd"), B is a substring of it. B is not a substring of A when it is repeated less than 3 times.