Problem 2014: Longest Subsequence Repeated k Times

Problem Information

Difficulty: Hard
Acceptance Rate: 71.42%
Paid Only: No
Tags: String, Backtracking, Greedy, Counting, Enumeration

Problem Description

You are given a string `s` of length `n`, and an integer `k`. You are tasked to find the **longest subsequence repeated** `k` times in string `s`.

A **subsequence** is a string that can be derived from another string by deleting some or no characters without changing the order of the remaining characters.

A subsequence `seq` is **repeated** `k` times in the string `s` if `seq * k` is a subsequence of `s`, where `seq * k` represents a string constructed by concatenating `seq` `k` times.

* For example, `"bba"` is repeated `2` times in the string `"bababcba"`, because the string `"bbabba"`, constructed by concatenating `"bba"` `2` times, is a subsequence of the string `"**_b_** a** _bab_** c** _ba_** "`.

Return _the**longest subsequence repeated** _`k` _times in string_`s` _. If multiple such subsequences are found, return the**lexicographically largest** one. If there is no such subsequence, return an **empty** string_.

Example 1:

![example 1](https://assets.leetcode.com/uploads/2021/08/30/longest-subsequence-repeat-k-times.png)

Input: s = "letsleetcode", k = 2 **Output:** "let" **Explanation:** There are two longest subsequences repeated 2 times: "let" and "ete". "let" is the lexicographically largest one.

Example 2:

Input: s = "bb", k = 2 **Output:** "b" **Explanation:** The longest subsequence repeated 2 times is "b".

Example 3:

Input: s = "ab", k = 2 **Output:** "" **Explanation:** There is no subsequence repeated 2 times. Empty string is returned.

Constraints:

* `n == s.length` * `2 <= k <= 2000` * `2 <= n < min(2001, k * 8)` * `s` consists of lowercase English letters.

Code Snippets

C++:

```
class Solution {
public:
string longestSubsequenceRepeatedK(string s, int k) {

}
};
```

Java:

```
class Solution {
public String longestSubsequenceRepeatedK(String s, int k) {

}
}
```

Python3:

```
class Solution:
def longestSubsequenceRepeatedK(self, s: str, k: int) -> str:
```