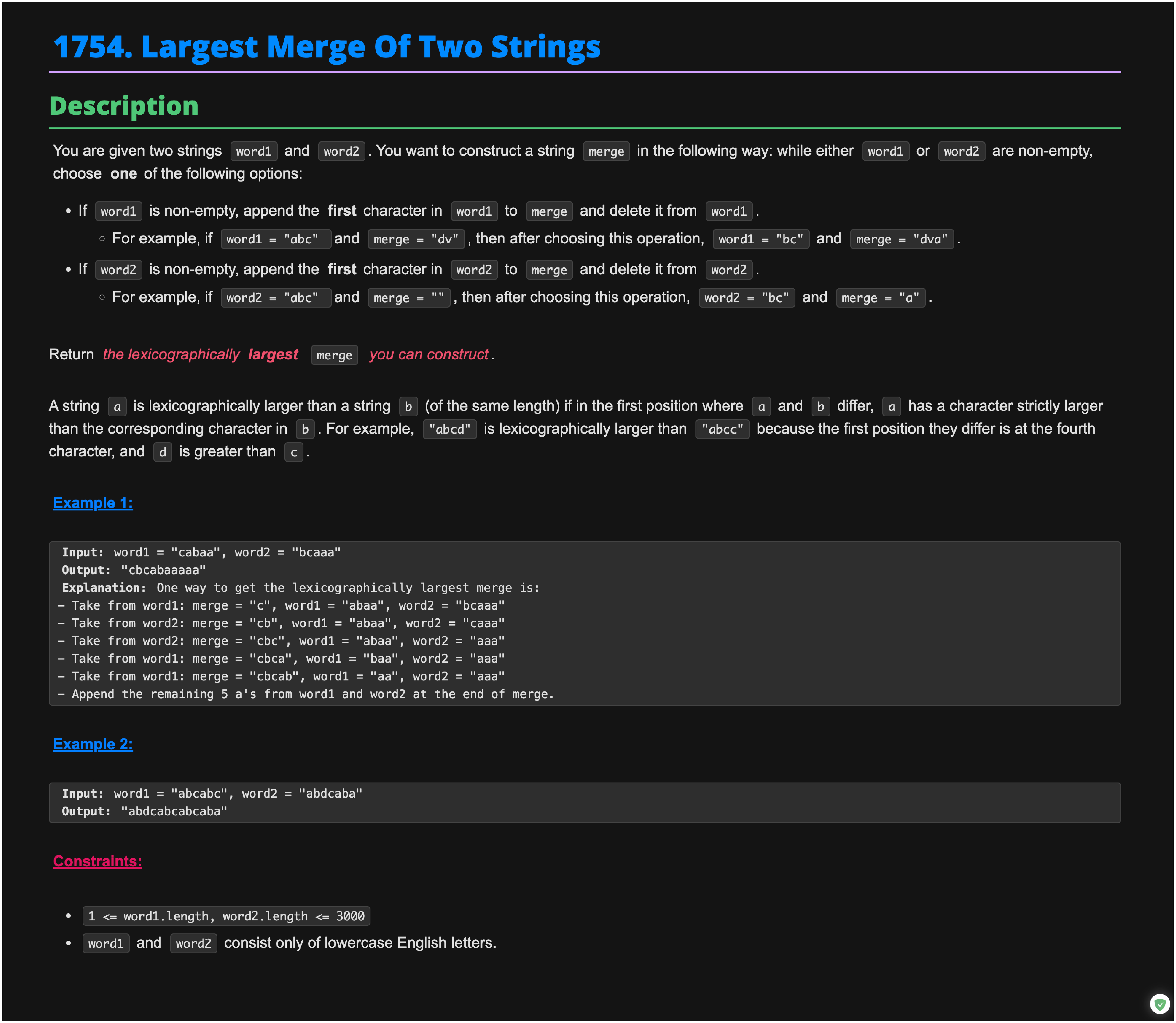

1754. Largest Merge Of Two Strings

Description

You are given two strings `word1` and `word2`. You want to construct a string `merge` in the following way: while either `word1` or `word2` are non-empty, choose **one** of the following options:

- If `word1` is non-empty, append the **first** character in `word1` to `merge` and delete it from `word1`.
    - For example, if `word1 = "abc"` and `merge = "dv"`, then after choosing this operation, `word1 = "bc"` and `merge = "dva"`.
- If `word2` is non-empty, append the **first** character in `word2` to `merge` and delete it from `word2`.
    - For example, if `word2 = "abc"` and `merge = ""`, then after choosing this operation, `word2 = "bc"` and `merge = "a"`.

Return *the lexicographically **largest*** `merge` *you can construct*.

A string `a` is lexicographically larger than a string `b` (of the same length) if in the first position where `a` and `b` differ, `a` has a character strictly larger than the corresponding character in `b`. For example, `"abcd"` is lexicographically larger than `"abcc"` because the first position they differ is at the fourth character, and `d` is greater than `c`.

Example 1:

```
Input: word1 = "cabaa", word2 = "bcaaa"
Output: "cbcabaaaaa"
Explanation: One way to get the lexicographically largest merge is:
- Take from word1: merge = "c", word1 = "abaa", word2 = "bcaaa"
- Take from word2: merge = "cb", word1 = "abaa", word2 = "caaa"
- Take from word2: merge = "cbc", word1 = "abaa", word2 = "aaa"
- Take from word1: merge = "cbca", word1 = "baa", word2 = "aaa"
- Take from word1: merge = "cbcab", word1 = "aa", word2 = "aaa"
- Append the remaining 5 a's from word1 and word2 at the end of merge.
```

Example 2:

```
Input: word1 = "abcabc", word2 = "abdcaba"
Output: "abdcabcabcaba"
```

Constraints:

- `1 <= word1.length, word2.length <= 3000`
- `word1` and `word2` consist only of lowercase English letters.