# 1055. Shortest Way to Form String

## Description

A **subsequence** of a string is a new string that is formed from the original string by deleting some (can be none) of the characters without disturbing the relative positions of the remaining characters. (i.e., `"ace"` is a subsequence of `"abcde"` while `"aec"` is not).

Given two strings `source` and `target`, return *the minimum number of* ***subsequences*** *of* `source` *such that their concatenation equals* `target`. If the task is impossible, return `-1`.

### Example 1:

```
Input: source = "abc", target = "abcbc"
Output: 2
Explanation: The target "abcbc" can be formed by "abc" and "bc", which are subsequences of source "abc".
```

### Example 2:

```
Input: source = "abc", target = "acdbc"
Output: -1
Explanation: The target string cannot be constructed from the subsequences of source string due to the character "d" in target string.
```

### Example 3:

```
Input: source = "xyz", target = "xzyxz"
Output: 3
Explanation: The target string can be constructed as follows "xz" + "y" + "xz".
```

### Constraints:

- `1 <= source.length, target.length <= 1000`
- `source` and `target` consist of lowercase English letters.