Cutting Binary String

You are given a binary string **s** consisting only of characters **'0'** and **'1'**. Your task is to split this string into the **minimum number** of non-empty **substrings** such that:

- Each substring represents a **power of 5** in decimal (e.g., 1, 5, 25, 125, ...).

- No substring should have **leading zeros**.

Return the **minimum number** of such pieces the string can be divided into.
Note: If it is **not possible** to split the string in this way, return **-1**.

Examples:

Input: s **=** "101101101"
Output: 3
Explanation: The string can be split into three substrings: "101", "101", and "101", each of which is a power of 5 with no leading zeros.

Input: s = "1111101"

Output: 1

Explanation: The string can be split into one binary string "1111101" which is 125 in decimal and a power of 5 with no leading zeros.

Input: s = "00000"
Output: -1
Explanation: There is no substring that can be split into power of **5.**

Constraints:
1 ≤ s.size() ≤ 30