You are given a **0-indexed** string street. Each character in street is either 'H' representing a house or '.' representing an empty space.

You can place buckets on the **empty spaces** to collect rainwater that falls from the adjacent houses. The rainwater from a house at index i is collected if a bucket is placed at index i - 1 **and/or** index i + 1. A single bucket, if placed adjacent to two houses, can collect the rainwater from **both** houses.

Return *the****minimum****number of buckets needed so that for****every****house, there is****at least****one bucket collecting rainwater from it, or*-1*if it is impossible.*

**Example 1:**

**Input:** street = "H..H"

**Output:** 2

**Explanation:**

We can put buckets at index 1 and index 2.

"H..H" -> "HBBH" ('B' denotes where a bucket is placed).

The house at index 0 has a bucket to its right, and the house at index 3 has a bucket to its left.

Thus, for every house, there is at least one bucket collecting rainwater from it.

**Example 2:**

**Input:** street = ".H.H."

**Output:** 1

**Explanation:**

We can put a bucket at index 2.

".H.H." -> ".HBH." ('B' denotes where a bucket is placed).

The house at index 1 has a bucket to its right, and the house at index 3 has a bucket to its left.

Thus, for every house, there is at least one bucket collecting rainwater from it.

**Example 3:**

**Input:** street = ".HHH."

**Output:** -1

**Explanation:**

There is no empty space to place a bucket to collect the rainwater from the house at index 2.

Thus, it is impossible to collect the rainwater from all the houses.

**Example 4:**

**Input:** street = "H"

**Output:** -1

**Explanation:**

There is no empty space to place a bucket.

Thus, it is impossible to collect the rainwater from the house.

**Example 5:**

**Input:** street = "."

**Output:** 0

**Explanation:**

There is no house to collect water from.

Thus, 0 buckets are needed.

**Constraints:**

* 1 <= street.length <= 105
* street[i] is either'H' or '.'.