

# 3043. Find the Length of the Longest Common Prefix

## Description

You are given two arrays with **positive** integers `arr1` and `arr2`.

A **prefix** of a positive integer is an integer formed by one or more of its digits, starting from its **leftmost** digit. For example, `123` is a prefix of the integer `12345`, while `234` is **not**.

A **common prefix** of two integers `a` and `b` is an integer `c`, such that `c` is a prefix of both `a` and `b`. For example, `5655359` and `56554` have a common prefix `565` while `1223` and `43456` **do not** have a common prefix.

You need to find the length of the **longest common prefix** between all pairs of integers `(x, y)` such that `x` belongs to `arr1` and `y` belongs to `arr2`.

Return *the length of the **longest** common prefix among all pairs*. *If no common prefix exists among them, return `0`*.

### Example 1:

**Input:** `arr1 = [1,10,100]`, `arr2 = [1000]`

**Output:** `3`

**Explanation:** There are 3 pairs `(arr1[i], arr2[j])`:

- The longest common prefix of `(1, 1000)` is `1`.
  - The longest common prefix of `(10, 1000)` is `10`.
  - The longest common prefix of `(100, 1000)` is `100`.
- The longest common prefix is `100` with a length of 3.

### Example 2:

**Input:** `arr1 = [1,2,3]`, `arr2 = [4,4,4]`

**Output:** `0`

**Explanation:** There exists no common prefix for any pair `(arr1[i], arr2[j])`, hence we return `0`.  
Note that common prefixes between elements of the same array do not count.

### Constraints:

- `1 <= arr1.length, arr2.length <= 5 * 104`
- `1 <= arr1[i], arr2[i] <= 108`

