# Problem 1805: Number of Different Integers in a String

## Problem Information

**Difficulty:** Easy
**Acceptance Rate:** 39.63%
**Paid Only:** No
**Tags:** Hash Table, String

## Problem Description

You are given a string `word` that consists of digits and lowercase English letters.

You will replace every non-digit character with a space. For example, `"a123bc34d8ef34"` will become `" 123 34 8 34"`. Notice that you are left with some integers that are separated by at least one space: `"123"`, `"34"`, `"8"`, and `"34"`.

Return _the number of**different** integers after performing the replacement operations on _`word`.

Two integers are considered different if their decimal representations **without any leading zeros** are different.

**Example 1:**

**Input:** word = "a _123_ bc _34_ d _8_ ef _34_ " **Output:** 3 **Explanation:** The three different integers are "123", "34", and "8". Notice that "34" is only counted once.

**Example 2:**

**Input:** word = "leet _1234_ code _234_ " **Output:** 2

**Example 3:**

**Input:** word = "a _1_ b _01_ c _001_ " **Output:** 1 **Explanation:** The three integers "1", "01", and "001" all represent the same integer because the leading zeros are ignored when comparing their decimal values.

**Constraints:**

* `1 <= word.length <= 1000` * `word` consists of digits and lowercase English letters.

## Code Snippets

**C++:**

```
class Solution {
public:
int numDifferentIntegers(string word) {

}
};
```

**Java:**

```
class Solution {
public int numDifferentIntegers(String word) {

}
}
```

**Python3:**

```
class Solution:
def numDifferentIntegers(self, word: str) -> int:
```