Problem 1432: Max Difference You Can Get From Changing an Integer

Problem Information

Difficulty: Medium
Acceptance Rate: 48.82%
Paid Only: No
Tags: Math, Greedy

Problem Description

You are given an integer `num`. You will apply the following steps to `num` **two** separate times:

* Pick a digit `x (0 <= x <= 9)`. * Pick another digit `y (0 <= y <= 9)`. Note `y` can be equal to `x`. * Replace all the occurrences of `x` in the decimal representation of `num` by `y`.

Let `a` and `b` be the two results from applying the operation to `num` _independently_.

Return _the max difference_ between `a` and `b`.

Note that neither `a` nor `b` may have any leading zeros, and **must not** be 0.

Example 1:

Input: num = 555 **Output:** 888 **Explanation:** The first time pick x = 5 and y = 9 and store the new integer in a. The second time pick x = 5 and y = 1 and store the new integer in b. We have now a = 999 and b = 111 and max difference = 888

Example 2:

Input: num = 9 **Output:** 8 **Explanation:** The first time pick x = 9 and y = 9 and store the new integer in a. The second time pick x = 9 and y = 1 and store the new integer in b. We have now a = 9 and b = 1 and max difference = 8

Constraints:

* `1 <= num <= 108`

Code Snippets

C++:

```
class Solution {
public:
int maxDiff(int num) {

}
};
```

Java:

```
class Solution {
public int maxDiff(int num) {

}
}
```

Python3:

```
class Solution:
def maxDiff(self, num: int) -> int:
```