Problem 2299: Strong Password Checker II

Problem Information

Difficulty: Easy
Acceptance Rate: 55.21%
Paid Only: No
Tags: String

Problem Description

A password is said to be **strong** if it satisfies all the following criteria:

* It has at least `8` characters. * It contains at least **one lowercase** letter. * It contains at least **one uppercase** letter. * It contains at least **one digit**. * It contains at least **one special character**. The special characters are the characters in the following string: `"!@#$%^&*()-+"`. * It does **not** contain `2` of the same character in adjacent positions (i.e., `"aab"` violates this condition, but `"aba"` does not).

Given a string `password`, return `true` _if it is a**strong** password_. Otherwise, return `false`.

Example 1:

Input: password = "IloveLe3tcode!" **Output:** true **Explanation:** The password meets all the requirements. Therefore, we return true.

Example 2:

Input: password = "Me+You--IsMyDream" **Output:** false **Explanation:** The password does not contain a digit and also contains 2 of the same character in adjacent positions. Therefore, we return false.

Example 3:

Input: password = "1aB!" **Output:** false **Explanation:** The password does not meet the length requirement. Therefore, we return false.

Constraints:

* `1 <= password.length <= 100` * `password` consists of letters, digits, and special characters: `"!@#$%^&*()-+"`.

Code Snippets

C++:

```
class Solution {
public:
bool strongPasswordCheckerII(string password) {

}
};
```

Java:

```
class Solution {
public boolean strongPasswordCheckerII(String password) {

}
}
```

Python3:

```
class Solution:
def strongPasswordCheckerII(self, password: str) -> bool:
```