Problem 71: Simplify Path

Problem Information

Difficulty: Medium
Acceptance Rate: 49.35%
Paid Only: No
Tags: String, Stack

Problem Description

You are given an _absolute_ path for a Unix-style file system, which always begins with a slash `'/'`. Your task is to transform this absolute path into its **simplified canonical path**.

The _rules_ of a Unix-style file system are as follows:

* A single period `'.'` represents the current directory. * A double period `'..'` represents the previous/parent directory. * Multiple consecutive slashes such as `'//'` and `'///'` are treated as a single slash `'/'`. * Any sequence of periods that does **not match** the rules above should be treated as a **valid directory or** **file****name**. For example, `'...' `and `'....'` are valid directory or file names.

The simplified canonical path should follow these _rules_ :

* The path must start with a single slash `'/'`. * Directories within the path must be separated by exactly one slash `'/'`. * The path must not end with a slash `'/'`, unless it is the root directory. * The path must not have any single or double periods (`'.'` and `'..'`) used to denote current or parent directories.

Return the **simplified canonical path**.

Example 1:

Input: path = "/home/"

Output: "/home"

Explanation:

The trailing slash should be removed.

Example 2:

Input: path = "/home//foo/"

Output: "/home/foo"

Explanation:

Multiple consecutive slashes are replaced by a single one.

Example 3:

Input: path = "/home/user/Documents/../Pictures"

Output: "/home/user/Pictures"

Explanation:

A double period `".."` refers to the directory up a level (the parent directory).

Example 4:

Input: path = "/../"

Output: "/"

Explanation:

Going one level up from the root directory is not possible.

Example 5:

Input: path = "/.../a/../b/c/../d/./"

Output: "/.../b/d"

Explanation:

`"..."` is a valid name for a directory in this problem.

Constraints:

* `1 <= path.length <= 3000` * `path` consists of English letters, digits, period `'.'`, slash `'/'` or `'_'`. * `path` is a valid absolute Unix path.

Code Snippets

C++:

```
class Solution {
public:
string simplifyPath(string path) {

}
};
```

Java:

```
class Solution {
public String simplifyPath(String path) {

}
}
```

Python3:

```
class Solution:
def simplifyPath(self, path: str) -> str:
```