Problem 2466: Count Ways To Build Good Strings

Problem Information

Difficulty: Medium
Acceptance Rate: 59.05%
Paid Only: No
Tags: Dynamic Programming

Problem Description

Given the integers `zero`, `one`, `low`, and `high`, we can construct a string by starting with an empty string, and then at each step perform either of the following:

* Append the character `'0'` `zero` times. * Append the character `'1'` `one` times.

This can be performed any number of times.

A **good** string is a string constructed by the above process having a **length** between `low` and `high` (**inclusive**).

Return _the number of**different** good strings that can be constructed satisfying these properties._ Since the answer can be large, return it **modulo** `109 + 7`.

Example 1:

Input: low = 3, high = 3, zero = 1, one = 1 **Output:** 8 **Explanation:** One possible valid good string is "011". It can be constructed as follows: "" -> "0" -> "01" -> "011". All binary strings from "000" to "111" are good strings in this example.

Example 2:

Input: low = 2, high = 3, zero = 1, one = 2 **Output:** 5 **Explanation:** The good strings are "00", "11", "000", "110", and "011".

Constraints:

* `1 <= low <= high <= 105` * `1 <= zero, one <= low`

Code Snippets

C++:

```
class Solution {
public:
int countGoodStrings(int low, int high, int zero, int one) {

}
};
```

Java:

```
class Solution {
public int countGoodStrings(int low, int high, int zero, int one) {

}
}
```

Python3:

```
class Solution:
def countGoodStrings(self, low: int, high: int, zero: int, one: int) -> int:
```