Problem 2834: Find the Minimum Possible Sum of a Beautiful Array

Problem Information

Difficulty: Medium
Acceptance Rate: 35.45%
Paid Only: No
Tags: Math, Greedy

Problem Description

You are given positive integers `n` and `target`.

An array `nums` is **beautiful** if it meets the following conditions:

* `nums.length == n`. * `nums` consists of pairwise **distinct** **positive** integers. * There doesn't exist two **distinct** indices, `i` and `j`, in the range `[0, n - 1]`, such that `nums[i] + nums[j] == target`.

Return _the**minimum** possible sum that a beautiful array could have modulo _`109 + 7`.

Example 1:

Input: n = 2, target = 3 **Output:** 4 **Explanation:** We can see that nums = [1,3] is beautiful. - The array nums has length n = 2. - The array nums consists of pairwise distinct positive integers. - There doesn't exist two distinct indices, i and j, with nums[i] + nums[j] == 3. It can be proven that 4 is the minimum possible sum that a beautiful array could have.

Example 2:

Input: n = 3, target = 3 **Output:** 8 **Explanation:** We can see that nums = [1,3,4] is beautiful. - The array nums has length n = 3. - The array nums consists of pairwise distinct positive integers. - There doesn't exist two distinct indices, i and j, with nums[i] + nums[j] == 3. It can be proven that 8 is the minimum possible sum that a beautiful array could have.

Example 3:

Input: n = 1, target = 1 **Output:** 1 **Explanation:** We can see, that nums = [1] is beautiful.

Constraints:

* `1 <= n <= 109` * `1 <= target <= 109`

Code Snippets

C++:

```
class Solution {
public:
int minimumPossibleSum(int n, int target) {

}
};
```

Java:

```
class Solution {
public int minimumPossibleSum(int n, int target) {

}
}
```

Python3:

```
class Solution:
def minimumPossibleSum(self, n: int, target: int) -> int:
```