Problem 2028: Find Missing Observations

Problem Information

Difficulty: Medium
Acceptance Rate: 57.38%
Paid Only: No
Tags: Array, Math, Simulation

Problem Description

You have observations of `n + m` **6-sided** dice rolls with each face numbered from `1` to `6`. `n` of the observations went missing, and you only have the observations of `m` rolls. Fortunately, you have also calculated the **average value** of the `n + m` rolls.

You are given an integer array `rolls` of length `m` where `rolls[i]` is the value of the `ith` observation. You are also given the two integers `mean` and `n`.

Return _an array of length_ `n` _containing the missing observations such that the_**average value** of the _`n + m` _rolls is_**exactly** _`mean`. If there are multiple valid answers, return _any of them_. If no such array exists, return _an empty array_.

The **average value** of a set of `k` numbers is the sum of the numbers divided by `k`.

Note that `mean` is an integer, so the sum of the `n + m` rolls should be divisible by `n + m`.

Example 1:

Input: rolls = [3,2,4,3], mean = 4, n = 2 **Output:** [6,6] **Explanation:** The mean of all n + m rolls is (3 + 2 + 4 + 3 + 6 + 6) / 6 = 4.

Example 2:

Input: rolls = [1,5,6], mean = 3, n = 4 **Output:** [2,3,2,2] **Explanation:** The mean of all n + m rolls is (1 + 5 + 6 + 2 + 3 + 2 + 2) / 7 = 3.

Example 3:

Input: rolls = [1,2,3,4], mean = 6, n = 4 **Output:** [] **Explanation:** It is impossible for the mean to be 6 no matter what the 4 missing rolls are.

Constraints:

* `m == rolls.length` * `1 <= n, m <= 105` * `1 <= rolls[i], mean <= 6`

Code Snippets

C++:

```cpp
class Solution {
public:
vector<int> missingRolls(vector<int>& rolls, int mean, int n) {

}
};
```

Java:

```java
class Solution {
public int[] missingRolls(int[] rolls, int mean, int n) {

}
}
```

Python3:

```python
class Solution:
def missingRolls(self, rolls: List[int], mean: int, n: int) -> List[int]:
```