1218. Longest Arithmetic Subsequence of Given Difference

Given an integer array `arr` and an integer `difference`, return the length of the longest subsequence in `arr` which is an arithmetic sequence such that the difference between adjacent elements in the subsequence equals `difference`.

A **subsequence** is a sequence that can be derived from `arr` by deleting some or no elements without changing the order of the remaining elements.

Example 1:

```
Input: arr = [1,2,3,4], difference = 1
Output: 4
Explanation: The longest arithmetic subsequence is [1,2,3,4].
```

Example 2:

```
Input: arr = [1,3,5,7], difference = 1
Output: 1
Explanation: The longest arithmetic subsequence is any single element.
```

Example 3:

```
Input: arr = [1,5,7,8,5,3,4,2,1], difference = -2
Output: 4
Explanation: The longest arithmetic subsequence is [7,5,3,1].
```

```python
class Solution:
    def longestSubsequence(self, nums: List[int], difference: int) -> int:
        dp = {}
        for el in nums:
            dp[el] = dp.get(el-difference,0)+1
        return max(dp.values())
```