Concatenation of Array

03 July 2022 06:08

Problem Statement

Given an integer array `nums` of length `n`, you want to create an array `ans` of length `2n` where `ans[i] == nums[i]` and `ans[i + n] == nums[i]` for `0 <= i < n` (**0-indexed**).
Specifically, `ans` is the **concatenation** of two `nums` arrays.
Return *the array* `ans`.

From <https://leetcode.com/problems/concatenation-of-array/>

Example 1:
Input: nums = [1,2,1]
Output: [1,2,1,1,2,1]
Explanation: The array ans is formed as follows:
- ans = [nums[0],nums[1],nums[2],nums[0],nums[1],nums[2]]
- ans = [1,2,1,1,2,1]

Example 2:
Input: nums = [1,3,2,1]
Output: [1,3,2,1,1,3,2,1]
Explanation: The array ans is formed as follows:
- ans = [nums[0],nums[1],nums[2],nums[3],nums[0],nums[1],nums[2],nums[3]]
- ans = [1,3,2,1,1,3,2,1]

Constraints:

- `n == nums.length`
- `1 <= n <= 1000`
- `1 <= nums[i] <= 1000`

From <https://leetcode.com/problems/concatenation-of-array/>

Possible approaches

Solution1(Best)

```
class Solution {
public:
   vector<int> getConcatenation(vector<int>& nums) {
      int n=nums.size();
      for(int i=0;i<n;i++){
         nums.push_back(nums[i]);
      }
      return nums;
   }
};
```

Method using Maths & vector.

```
class Solution {
public:
   vector<int> getConcatenation(vector<int>& nums) {
      int n=nums.size();
      vector<int>s(n*2);
      for(int i=0;i<n;i++){
         s[i]=nums[i];
          s[n*2-(n-i)]=nums[i];
      }
      return s;
   }
};
```

In Method using Maths:
->First we double the size of vector
->In Loop
->Assigning the first value to Vector s from nums vector
->At the same time we also assigning the Nums[i] using n*2-(n-i)
i.e if n=3 and i=0 then it means s[6-(3-0)] which Is s[3].