384. Shuffle an Array

QuestionEditorial Solution

My Submissions

- Total Accepted: **4963**
- Total Submissions: **10990**
- Difficulty: **Medium**

Shuffle a set of numbers without duplicates.

Example:

// Init an array with set 1, 2, and 3.

int[] nums = {1,2,3};

Solution solution = new Solution(nums);

// Shuffle the array [1,2,3] and return its result. Any permutation of [1,2,3] must equally likely to be returned.

solution.shuffle();

// Resets the array back to its original configuration [1,2,3].

solution.reset();

// Returns the random shuffling of array [1,2,3].

solution.shuffle();

```
class Solution {
private:
vector<int> origin;
vector<int> sol;
public:
    Solution(vector<int> nums) {
        this->origin=nums;
        this->sol=nums;
    }

    /** Resets the array to its original configuration and return it. */
    vector<int> reset() {
        return this->origin;
    }

    /** Returns a random shuffling of the array. */
    vector<int> shuffle() {
        int n=origin.size()-1;
        for(int j=0;j<=n;j++)
        {
            int i=rand()%(n-j+1);
            swap(sol[i],sol[n-j]);
        }
        return sol;
    }
};

/**
 * Your Solution object will be instantiated and called as such:
 * Solution obj = new Solution(nums);
 * vector<int> param_1 = obj.reset();
 * vector<int> param_2 = obj.shuffle();
 */
```