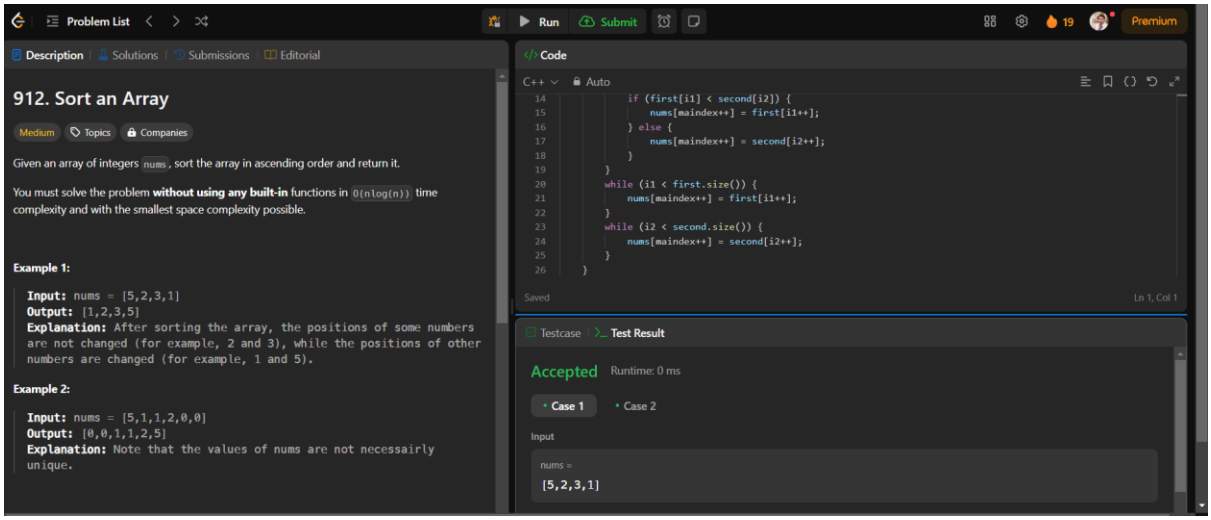
912. Sort an Array



Code:

```
class Solution {
public:
    void merge(vector<int>& nums, int s, int e) {
        int m = (s + e) / 2;
        vector<int> first(m - s + 1), second(e - m);
        for (int i = 0; i < first.size(); i++) {
            first[i] = nums[s + i];
        }
        for (int i = 0; i < second.size(); i++) {
            second[i] = nums[m + 1 + i];
        }
        int i1 = 0, i2 = 0, maindex = s;
        while (i1 < first.size() && i2 < second.size()) {
            if (first[i1] < second[i2]) {
                nums[maindex++] = first[i1++];
            } else {
                nums[maindex++] = second[i2++];
            }
        }
        while (i1 < first.size()) {
```

```
            nums[maindex++] = first[i1++];
        }
        while (i2 < second.size()) {
            nums[maindex++] = second[i2++];
        }
    }
    void mergesort(vector<int>& nums, int s, int e) {
        if (s >= e) {
            return;
        }
        int m = (s + e) / 2;
        mergesort(nums, s, m);
        mergesort(nums, m + 1, e);
        merge(nums, s, e);
    }
    vector<int> sortArray(vector<int>& nums) {
        mergesort(nums, 0, nums.size() - 1);
        return nums;
    }
};
```