Experiment-7

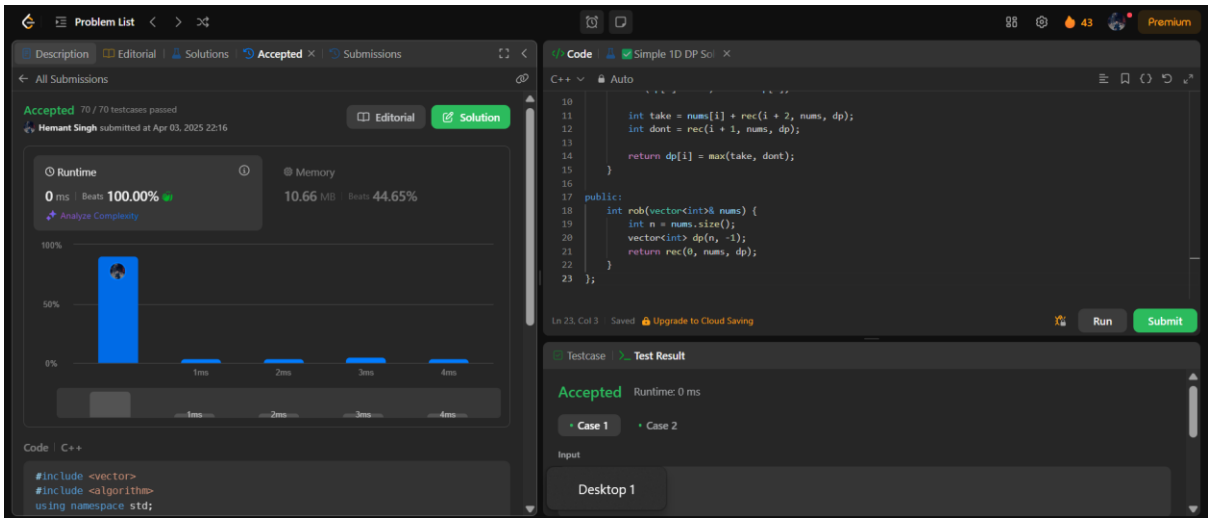
Name: Hemant Singh **UID-22BCS12820**

Q1. https://leetcode.com/problems/house-robber/submissions/1595767881/

Code:

```
#include <vector>
#include <algorithm>
using namespace std;
class Solution {
private:
    int rec(int i, vector<int>& nums, vector<int>& dp) {
        if (i >= nums.size()) return 0;
        if (dp[i] != -1) return dp[i];
        int take = nums[i] + rec(i + 2, nums, dp);
        int dont = rec(i + 1, nums, dp);
        return dp[i] = max(take, dont);
    }
public:
    int rob(vector<int>& nums) {
        int n = nums.size();
        vector<int> dp(n, -1);
        return rec(0, nums, dp);
    }
};
```

Output:

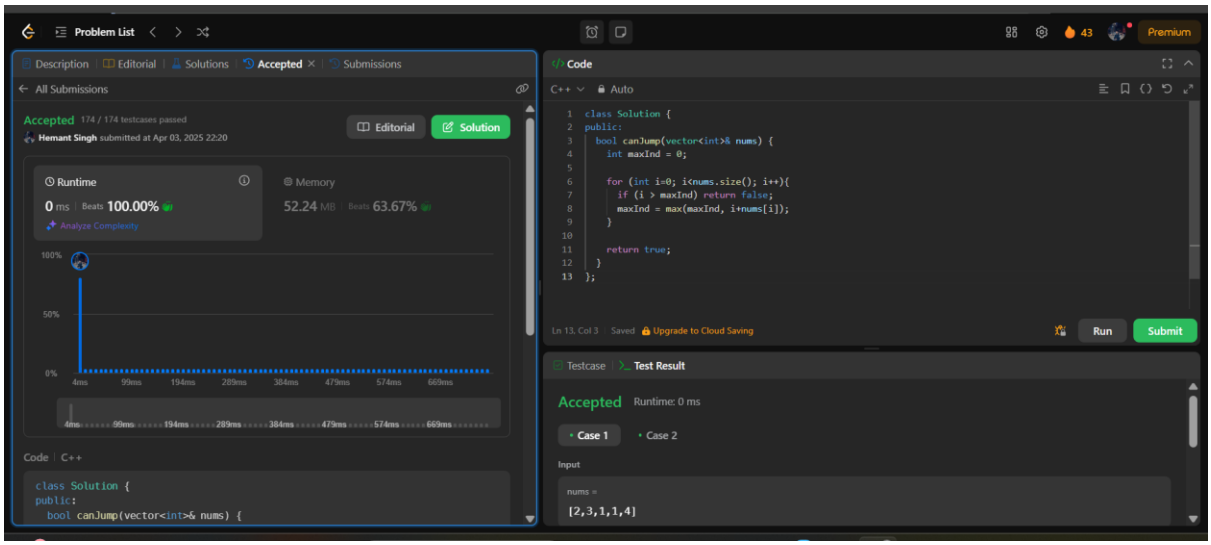


Q2. https://leetcode.com/problems/jump-game/solutions/5130181/video-move-goal-position/

Code:

```
class Solution {
public:
  bool canJump(vector<int>& nums) {
    int maxInd = 0;
    for (int i=0; i<nums.size(); i++){
      if (i > maxInd) return false;
      maxInd = max(maxInd, i+nums[i]);
    }
    return true;
  }
};
```

Output:



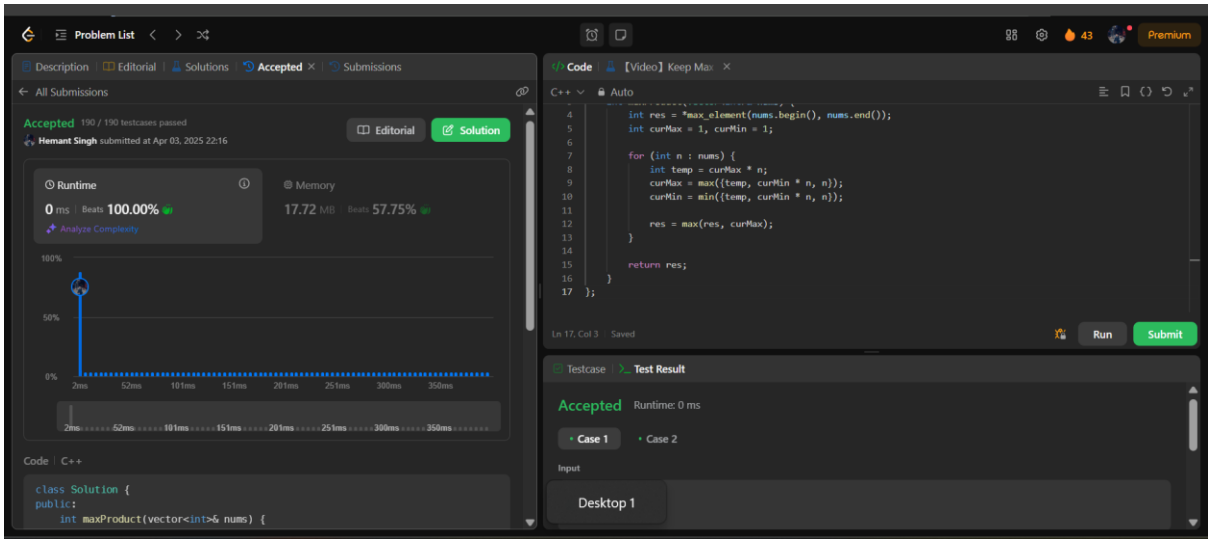
Q3. https://leetcode.com/problems/maximum-product-subarray/submissions/1595768633/

Code: class Solution {

public:

```
    int maxProduct(vector<int>& nums) {
        int res = *max_element(nums.begin(), nums.end());
        int curMax = 1, curMin = 1;
        for (int n : nums) {
            int temp = curMax * n;
            curMax = max({temp, curMin * n, n});
            curMin = min({temp, curMin * n, n});
            res = max(res, curMax);
        }
        return res;
    }};
```

Output:



Q4. https://leetcode.com/problems/perfect-squares/submissions/1595769259/

Code:

```
class Solution {
public:
    int numSquares(int n) {
        vector<int> dp(n + 1, INT_MAX);
        dp[0] = 0;
        for (int i = 1; i <= n; ++i) {
            for (int j = 1; j * j <= i; ++j){
            dp[i] = min(dp[i], dp[i - j * j] + 1);
            }

        }
```

```
        return dp[n];
    }
};
```

Output: