Experiment-3.3

# Student Name: Praduman Kumar UID: 20BCS9446

**Branch: BE-CSE Section/Group: 714/A**

# Semester: 6th Subject Code: 20CSP-351

**Subject Name: Competitive coding - II**

# Aim/Overview of the practical:

**To implement the concept of dynamic programming.**

**House Robber - ii.**

https://leetcode.com/problems/house-robber-ii/

# Code:

class Solution {

public int rob(int[] nums) { if(nums.length == 1)

return nums[0]; if(nums.length == 2)

return Math.max(nums[0], nums[1]);

int resultWithFirst = solve(nums, 0, nums.length - 2); int resultWithLast = solve(nums, 1, nums.length - 1);

return Math.max(resultWithFirst, resultWithLast);

}

public int solve(int[] nums, int start, int end) { if(start == end) return nums[start];

int money[] = new int[nums.length]; money[start] = nums[start];

money[start + 1] = Math.max(nums[start + 1], nums[start]);

for (int i = start + 2; i <= end; ++i)

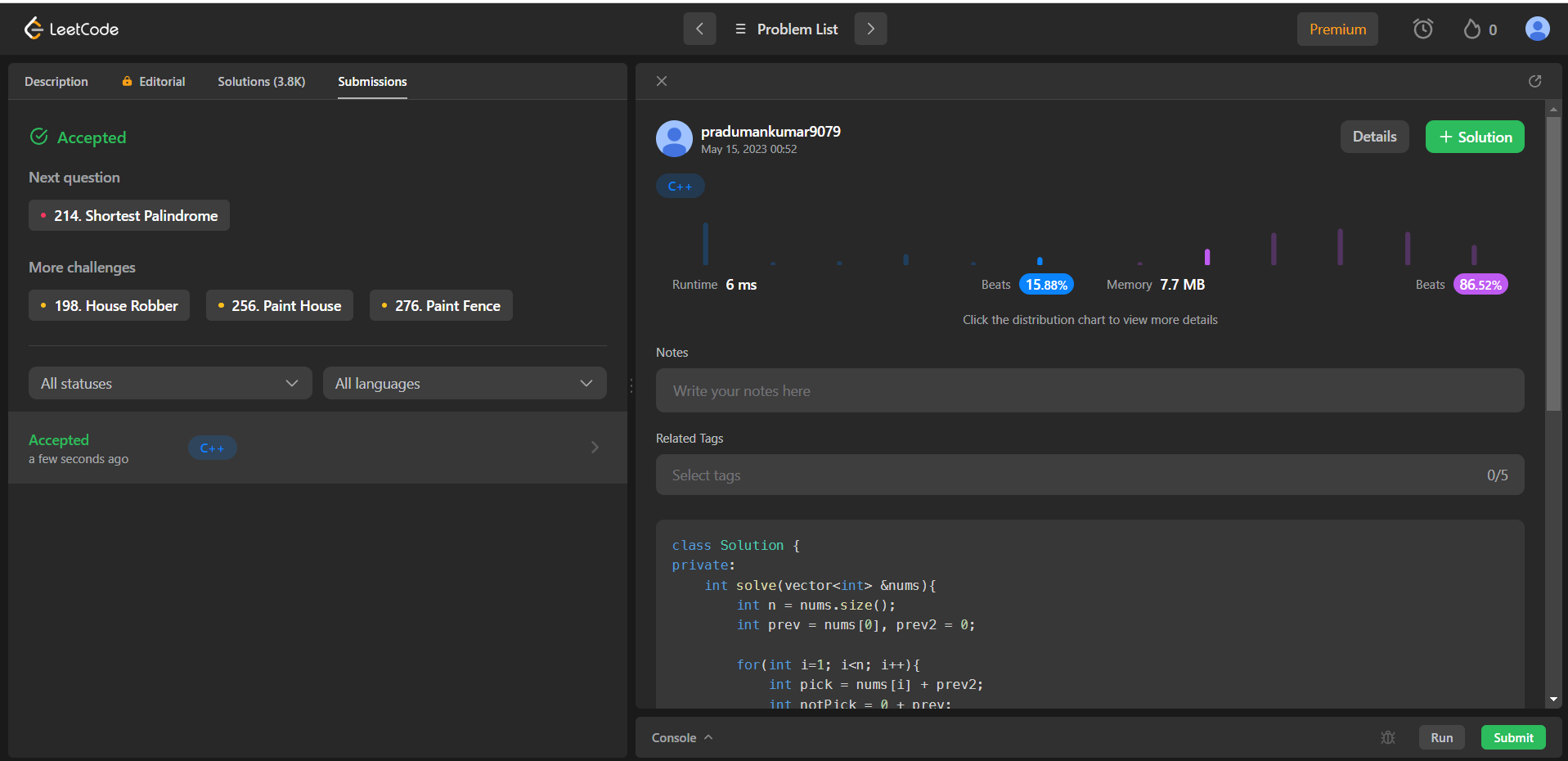
money[i] = Math.max(money[i - 1], money[i - 2] + nums[i]);

return money[end];

}

}

# Result/Output/Writing Summary:



1. **Maximum - Subarray.**

https://leetcode.com/problems/maximum-subarray/

# Code:

class Solution {

public int maxSubArray(int[] nums) { int max=Integer.MIN\_VALUE; int sum=0; int n=nums.length;

for(int i=0;i<n;i++){ sum+=nums[i];

max = Math.max(sum,max);

if(sum<0)sum = 0;

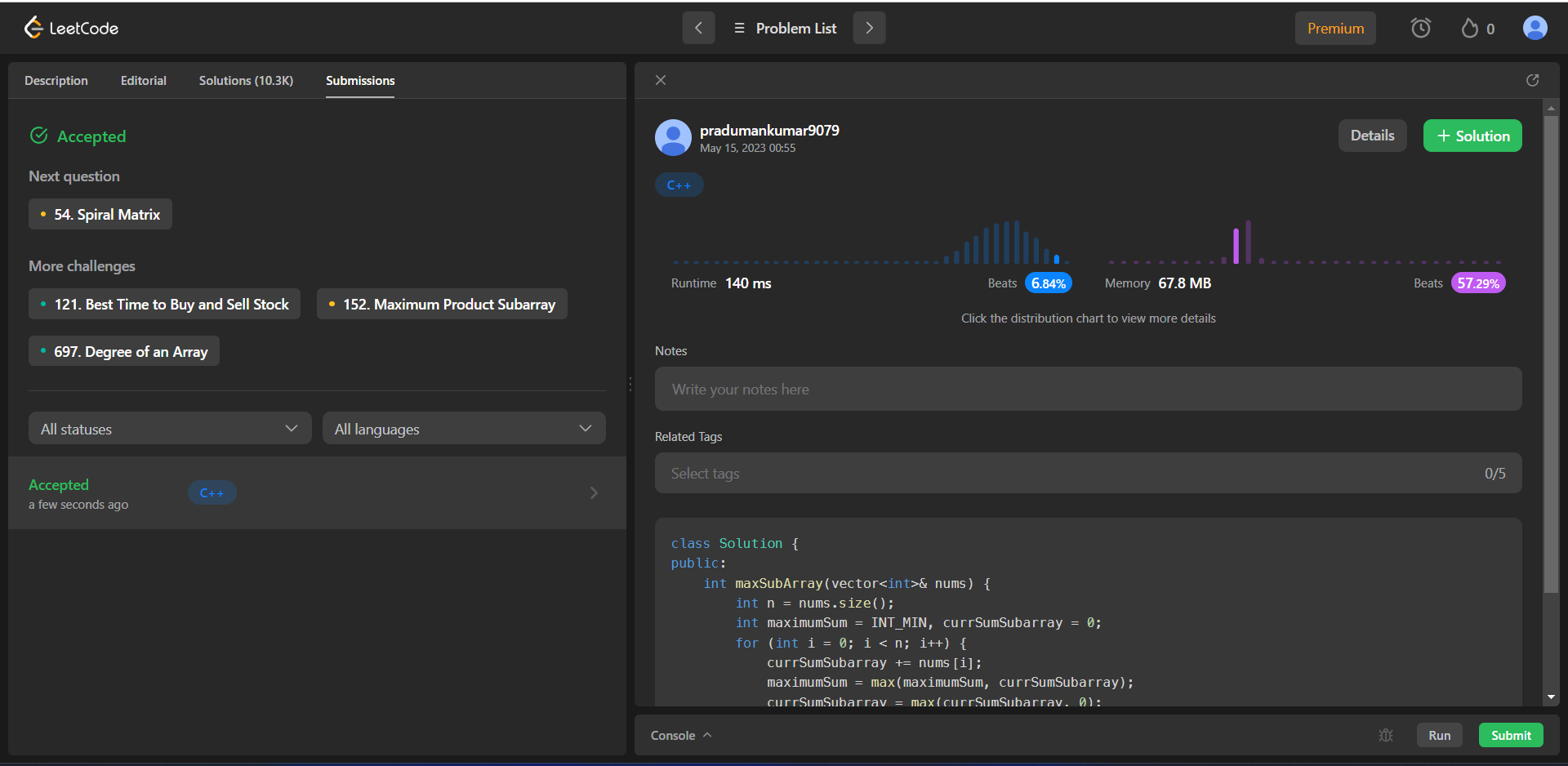
}

return max;

}

}

# Result/Output/Writing Summary:



**Learning outcomes (What I have learnt):**

* Learned the concept of Dynamic Programming in Fibonacci Sequence and so on.
* Learnt about House Robber-ii to Target & Maximum Subarray.