Problem List

Description | Editorial | Solutions | Submissions

300. Longest Increasing Subsequence

Medium | Topics | Companies

Given an integer array `nums`, return *the length of the longest* ***strictly increasing*** *subsequence*.

Example 1:

> **Input:** nums = [10,9,2,5,3,7,101,18]
> **Output:** 4
> **Explanation:** The longest increasing subsequence is [2,3,7,101], therefore the length is 4.

Example 2:

> **Input:** nums = [0,1,0,3,2,3]
> **Output:** 4

Example 3:

> **Input:** nums = [7,7,7,7,7,7,7]
> **Output:** 1

Constraints:

- `1 <= nums.length <= 2500`
- `-10^4 <= nums[i] <= 10^4`

Follow up: Can you come up with an algorithm that runs in `O(n log(n))` time complexity?

Seen this question in a real interview before? 1/5

21.6K | 217 | 314 Online

Premium

Code

Python3 | Auto

```python
            idx = bisect_left(sub, num)  # Find the first element greater or equal to num
            if idx == len(sub):
                sub.append(num)  # Append if num is larger than all elements
            else:
                sub[idx] = num  # Replace an element to maintain a minimal sequence
        return len(sub)  # The length of sub is the answer

# Example usage:
solution = Solution()
print(solution.lengthOfLIS([10,9,2,5,3,7,101,18]))  # Output: 4
print(solution.lengthOfLIS([0,1,0,3,2,3]))  # Output: 4
print(solution.lengthOfLIS([7,7,7,7,7,7,7]))  # Output: 1

```

Ln 19, Col 1 | Saved

Run | Submit

Testcase | Test Result

Accepted Runtime: 0 ms

Case 1 | Case 2 | Case 3

Input

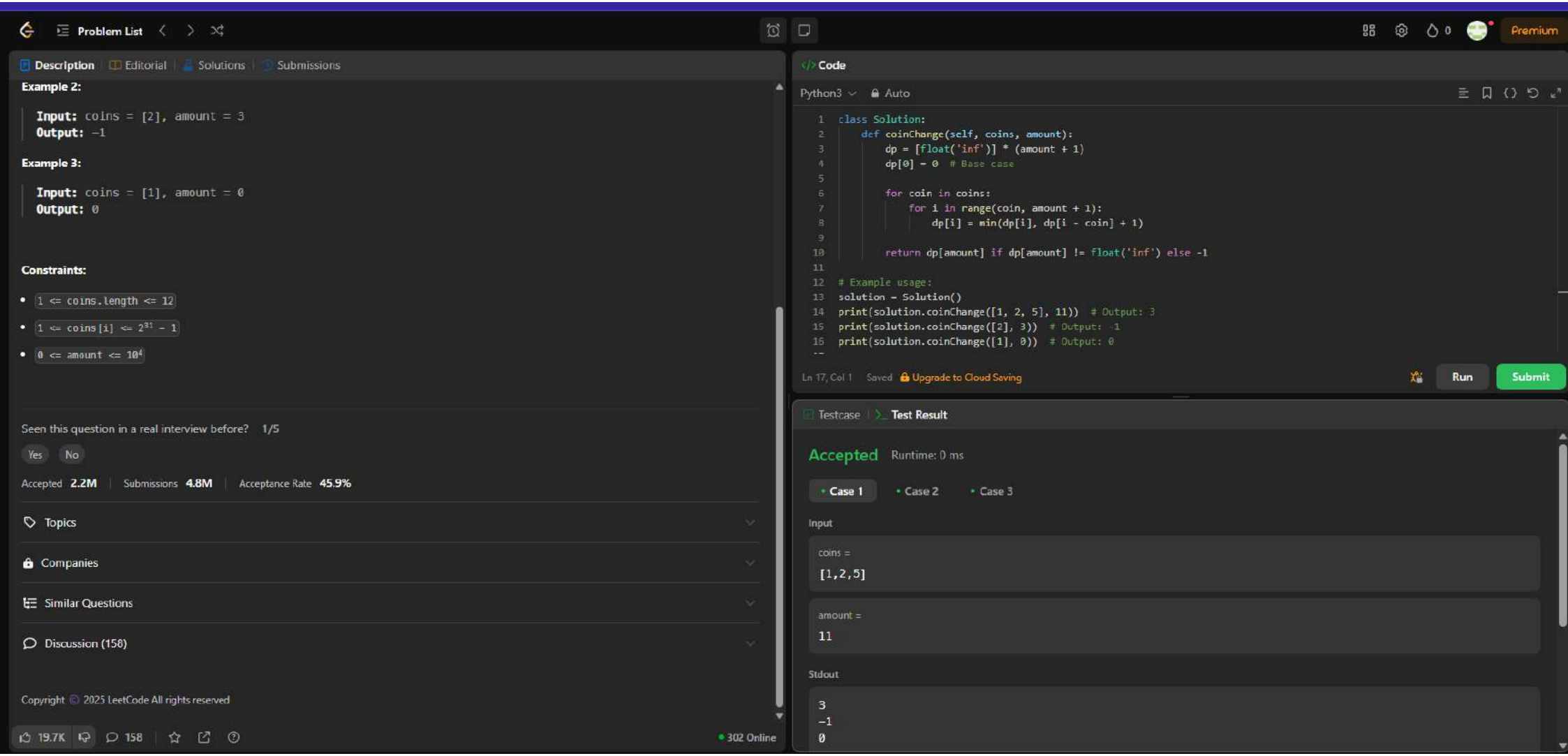
nums =
[10,9,2,5,3,7,101,18]

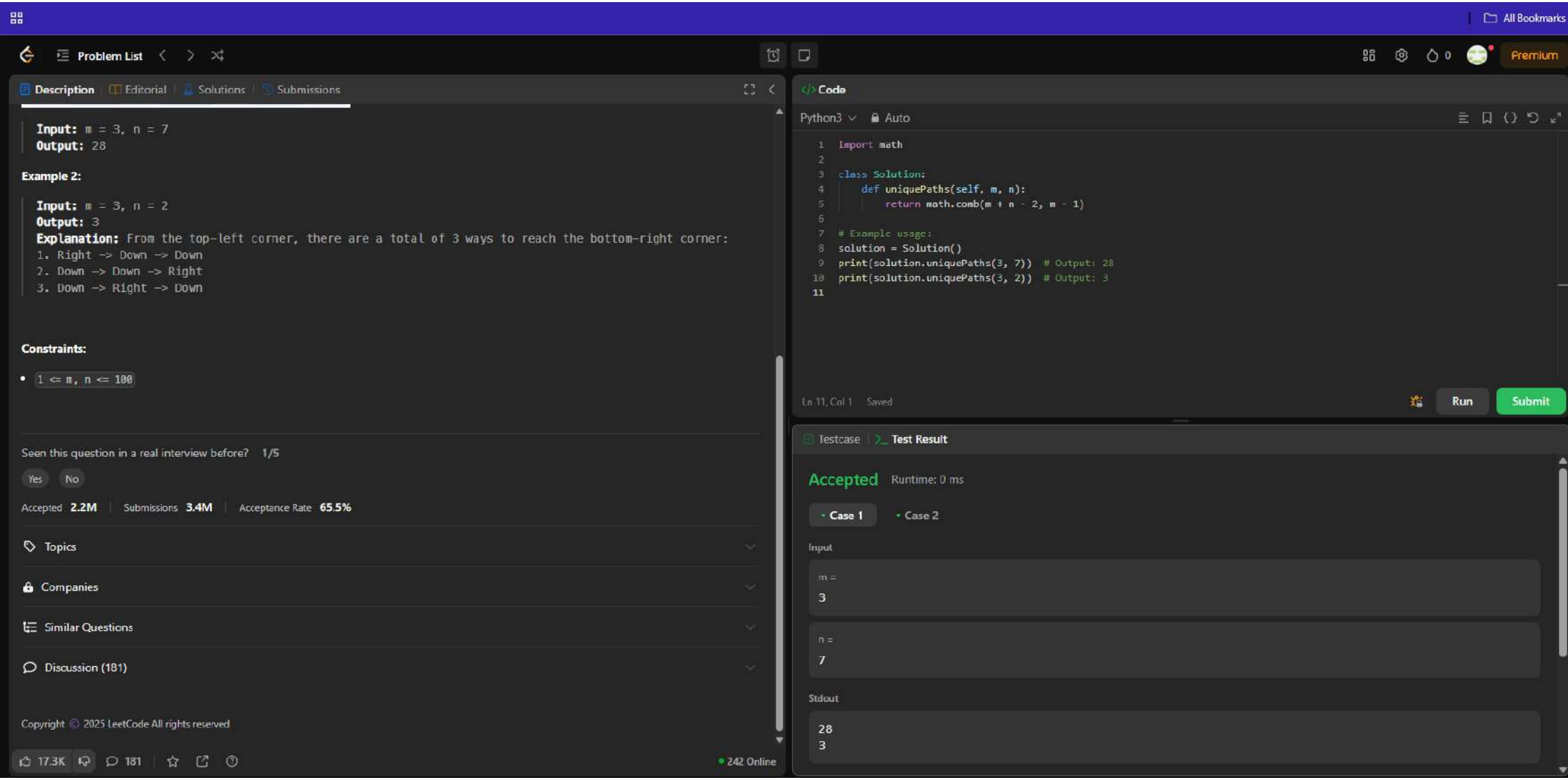
Stdout

```
4
4
1
```

Output

4


Problem List
Premium
Description
Editorial
Solutions
Submissions
Example 2:
Input: coins = [2], amount = 3
Output: -1
Example 3:
Input: coins = [1], amount = 0
Output: 0
Constraints:
1 <= coins.length <= 12
1 <= coins[i] <= 2^31 - 1
0 <= amount <= 10^4
Seen this question in a real interview before? 1/5
Yes
No
Accepted 2.2M | Submissions 4.8M | Acceptance Rate 45.9%
Topics
Companies
Similar Questions
Discussion (158)
Copyright © 2025 LeetCode All rights reserved
19.7K
158
302 Online
Code
Python3
Auto
class Solution:
def coinChange(self, coins, amount):
dp = [float('inf')] * (amount + 1)
dp[0] = 0 # Base case
for coin in coins:
for i in range(coin, amount + 1):
dp[i] = min(dp[i], dp[i - coin] + 1)
return dp[amount] if dp[amount] != float('inf') else -1
# Example usage:
solution = Solution()
print(solution.coinChange([1, 2, 5], 11)) # Output: 3
print(solution.coinChange([2], 3)) # Output: -1
print(solution.coinChange([1], 0)) # Output: 0
Ln 17, Col 1
Saved
Upgrade to Cloud Saving
Run
Submit
Testcase
Test Result
Accepted
Runtime: 0 ms
Case 1
Case 2
Case 3
Input
coins =
[1,2,5]
amount =
11
Stdout
3
-1
0


All Bookmarks
Problem List
Premium
Description
Editorial
Solutions
Submissions
Input: m = 3, n = 7
Output: 28
Example 2:
Input: m = 3, n = 2
Output: 3
Explanation: From the top-left corner, there are a total of 3 ways to reach the bottom-right corner:
1. Right -> Down -> Down
2. Down -> Down -> Right
3. Down -> Right -> Down
Constraints:
1 <= m, n <= 100
Seen this question in a real interview before? 1/5
Yes
No
Accepted 2.2M
Submissions 3.4M
Acceptance Rate 65.5%
Topics
Companies
Similar Questions
Discussion (181)
Copyright © 2025 LeetCode All rights reserved
17.3K
181
242 Online
Code
Python3
Auto
1 Import math
2
3 class Solution:
4 def uniquePaths(self, m, n):
5 return math.comb(m + n - 2, m - 1)
6
7 # Example usage:
8 solution = Solution()
9 print(solution.uniquePaths(3, 7)) # Output: 28
10 print(solution.uniquePaths(3, 2)) # Output: 3
11
Ln 11, Col 1
Saved
Run
Submit
Testcase
Test Result
Accepted
Runtime: 0 ms
Case 1
Case 2
Input
m =
3
n =
7
Stdout
28
3

Explanation: The subarray [1] has the largest sum 1.

Example 3:

```
Input: nums = [5,4,-1,7,8]
Output: 23
Explanation: The subarray [5,4,-1,7,8] has the largest sum 23.
```

Constraints:

- `1 <= nums.length <= 10^5`
- `-10^4 <= nums[i] <= 10^4`

Follow up: If you have figured out the `O(n)` solution, try coding another solution using the **divide and conquer** approach, which is more subtle.

Seen this question in a real interview before? 1/5

Yes No

Accepted 4.8M | Submissions 9.2M | Acceptance Rate 51.8%

Topics

Companies

Similar Questions

Discussion (342)

Code

Python3 Auto

```python
class Solution:
    def maxSubArray(self, nums):
        max_sum = cur_sum = nums[0]
        for num in nums[1:]:
            cur_sum = max(num, cur_sum + num)
            max_sum = max(max_sum, cur_sum)
        return max_sum

# Example usage:
solution = Solution()
print(solution.maxSubArray([-2,1,-3,4,-1,2,1,-5,4]))  # Output: 6
print(solution.maxSubArray([1]))  # Output: 1
print(solution.maxSubArray([5,4,-1,7,8]))  # Output: 23

```

Ln 14, Col 1 | Saved

Run Submit

Testcase | Test Result

Accepted Runtime: 0 ms

Case 1 Case 2 Case 3

Input

nums =
[-2,1,-3,4,-1,2,1,-5,4]

Stdout

```
6
1
23
```

Output

6