

Problem 312: Burst Balloons

Problem Information

Difficulty: Hard

Acceptance Rate: 62.48%

Paid Only: No

Tags: Array, Dynamic Programming

Problem Description

You are given `n` balloons, indexed from `0` to `n - 1`. Each balloon is painted with a number on it represented by an array `nums`. You are asked to burst all the balloons.

If you burst the `ith` balloon, you will get `nums[i - 1] * nums[i] * nums[i + 1]` coins. If `i - 1` or `i + 1` goes out of bounds of the array, then treat it as if there is a balloon with a `1` painted on it.

Return _the maximum coins you can collect by bursting the balloons wisely_.

Example 1:

Input: nums = [3,1,5,8] **Output:** 167 **Explanation:** nums = [3,1,5,8] --> [3,5,8] --> [3,8] --> [8] --> [] coins = $3 \cdot 1 \cdot 5 + 3 \cdot 5 \cdot 8 + 1 \cdot 3 \cdot 8 + 1 \cdot 8 \cdot 1 = 167$

Example 2:

Input: nums = [1,5] **Output:** 10

Constraints:

* `n == nums.length` * `1 <= n <= 300` * `0 <= nums[i] <= 100`

Code Snippets

C++:

```
class Solution {  
public:  
int maxCoins(vector<int>& nums) {  
  
}  
};
```

Java:

```
class Solution {  
public int maxCoins(int[] nums) {  
  
}  
}
```

Python3:

```
class Solution:  
def maxCoins(self, nums: List[int]) -> int:
```