

Problem 3429: Paint House IV

Problem Information

Difficulty: Medium

Acceptance Rate: 44.62%

Paid Only: No

Tags: Array, Dynamic Programming

Problem Description

You are given an **even** integer n representing the number of houses arranged in a straight line, and a 2D array `cost` of size $n \times 3$, where `cost[i][j]` represents the cost of painting house i with color $j + 1$.

The houses will look **beautiful** if they satisfy the following conditions:

- * No **two** adjacent houses are painted the same color.
- * Houses **equidistant** from the ends of the row are **not** painted the same color. For example, if $n = 6$, houses at positions $(0, 5)$, $(1, 4)$, and $(2, 3)$ are considered equidistant.

Return the **minimum** cost to paint the houses such that they look **beautiful**.

Example 1:

Input: $n = 4$, `cost = [[3,5,7],[6,2,9],[4,8,1],[7,3,5]]`

Output: 9

Explanation:

The optimal painting sequence is $[1, 2, 3, 2]$ with corresponding costs $[3, 2, 1, 3]$. This satisfies the following conditions:

- * No adjacent houses have the same color.
- * Houses at positions 0 and 3 (equidistant from the ends) are not painted the same color $(1 \neq 2)$.
- * Houses at positions 1 and 2 (equidistant from the ends) are not painted the same color $(2 \neq 3)$.

The minimum cost to paint the houses so that they look beautiful is $3 + 2 + 1 + 3 = 9$.

Example 2:

Input: $n = 6$, $\text{cost} = [[2,4,6],[5,3,8],[7,1,9],[4,6,2],[3,5,7],[8,2,4]]$

Output: 18

Explanation:

The optimal painting sequence is $[1, 3, 2, 3, 1, 2]$ with corresponding costs $[2, 8, 1, 2, 3, 2]$. This satisfies the following conditions:

- * No adjacent houses have the same color.
- * Houses at positions 0 and 5 (equidistant from the ends) are not painted the same color $(1 \neq 2)$.
- * Houses at positions 1 and 4 (equidistant from the ends) are not painted the same color $(3 \neq 1)$.
- * Houses at positions 2 and 3 (equidistant from the ends) are not painted the same color $(2 \neq 3)$.

The minimum cost to paint the houses so that they look beautiful is $2 + 8 + 1 + 2 + 3 + 2 = 18$.

Constraints:

- * $2 \leq n \leq 105$
- * n is even.
- * $\text{cost.length} == n$
- * $\text{cost}[i].\text{length} == 3$
- * $0 \leq \text{cost}[i][j] \leq 105$

Code Snippets

C++:

```
class Solution {
public:
    long long minCost(int n, vector<vector<int>>& cost) {

    }
};
```

Java:

```
class Solution {  
    public long minCost(int n, int[][] cost) {  
  
    }  
}
```

Python3:

```
class Solution:  
    def minCost(self, n: int, cost: List[List[int]]) -> int:
```