Problem 746: Min Cost Climbing Stairs

Problem Information

Difficulty: Easy
Acceptance Rate: 67.74%
Paid Only: No
Tags: Array, Dynamic Programming

Problem Description

You are given an integer array `cost` where `cost[i]` is the cost of `ith` step on a staircase. Once you pay the cost, you can either climb one or two steps.

You can either start from the step with index `0`, or the step with index `1`.

Return _the minimum cost to reach the top of the floor_.

Example 1:

Input: cost = [10,_15_ ,20] **Output:** 15 **Explanation:** You will start at index 1. - Pay 15 and climb two steps to reach the top. The total cost is 15.

Example 2:

Input: cost = [_1_ ,100,_1_ ,1,_1_ ,100,_1_ ,_1_ ,100,_1_] **Output:** 6 **Explanation:** You will start at index 0. - Pay 1 and climb two steps to reach index 2. - Pay 1 and climb two steps to reach index 4. - Pay 1 and climb two steps to reach index 6. - Pay 1 and climb one step to reach index 7. - Pay 1 and climb two steps to reach index 9. - Pay 1 and climb one step to reach the top. The total cost is 6.

Constraints:

* `2 <= cost.length <= 1000` * `0 <= cost[i] <= 999`

Code Snippets

C++:

```
class Solution {
public:
int minCostClimbingStairs(vector<int>& cost) {

}
};
```

Java:

```
class Solution {
public int minCostClimbingStairs(int[] cost) {

}
}
```

Python3:

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