maximum sum such that no 2 elements are adjacent

Stickler the thief wants to loot money from a society having **n** houses in a single line. He is a weird person and follows a certain **rule** when looting the houses. According to the rule, he will **never loot two consecutive houses**. At the same time, he wants to **maximize** the amount he **loots**. The thief knows which house has what amount of money but is unable to come up with an optimal looting strategy. He asks for your help to **find the maximum money he can get** if he strictly **follows** the **rule**. Each house has **a[i]**** amount of money** present in it.

Example 1:

Input: n = 6
a[] = {5,5,10,100,10,5}
Output: 110 **Explanation:** 5+100+5=110

Example 2:

Input: n = 3
a[] = {1,2,3}
Output: 4 **Explanation:** 1+3=4

Your Task:
Complete the function **FindMaxSum() **which takes an array **arr[]** and **n** as input which returns the maximum money he can get following the rules

Expected Time Complexity: O(N).
Expected Space Complexity: O(N).

Constraints:
1 <= n <= 104
1 <= a[i] <= 104

```
def FindMaxSum(self,arr, n):

        # code here
        if n<3:
            return max(arr)
        dp = [0]*(n+1)
        dp[0] = 0
        dp[1] = arr[0]
```

```
        dp[2] = max(arr[0],arr[1])
        for i in range(3,n+1):
            dp[i] = max(arr[i-1]+max(dp[i-2],dp[i-3]),dp[i-1])
        return dp[n]
```