2551. Put Marbles in Bags

Description

You have `k` bags. You are given a **0-indexed** integer array `weights` where `weights[i]` is the weight of the `i th` marble. You are also given the integer `k.`

Divide the marbles into the `k` bags according to the following rules:

- No bag is empty.
- If the `i th` marble and `j th` marble are in a bag, then all marbles with an index between the `i th` and `j th` indices should also be in that same bag.
- If a bag consists of all the marbles with an index from `i` to `j` inclusively, then the cost of the bag is `weights[i] + weights[j]`.

The **score** after distributing the marbles is the sum of the costs of all the `k` bags.

Return *the **difference** between the **maximum** and **minimum** scores among marble distributions*.

Example 1:

```
Input: weights = [1,3,5,1], k = 2
Output: 4
Explanation:
The distribution [1],[3,5,1] results in the minimal score of (1+1) + (3+1) = 6.
The distribution [1,3],[5,1], results in the maximal score of (1+3) + (5+1) = 10.
Thus, we return their difference 10 - 6 = 4.
```

Example 2:

```
Input: weights = [1, 3], k = 2
Output: 0
Explanation: The only distribution possible is [1],[3].
Since both the maximal and minimal score are the same, we return 0.
```

Constraints:

- $1 <= k <= weights.length <= 10^5$
- $1 <= weights[i] <= 10^9$