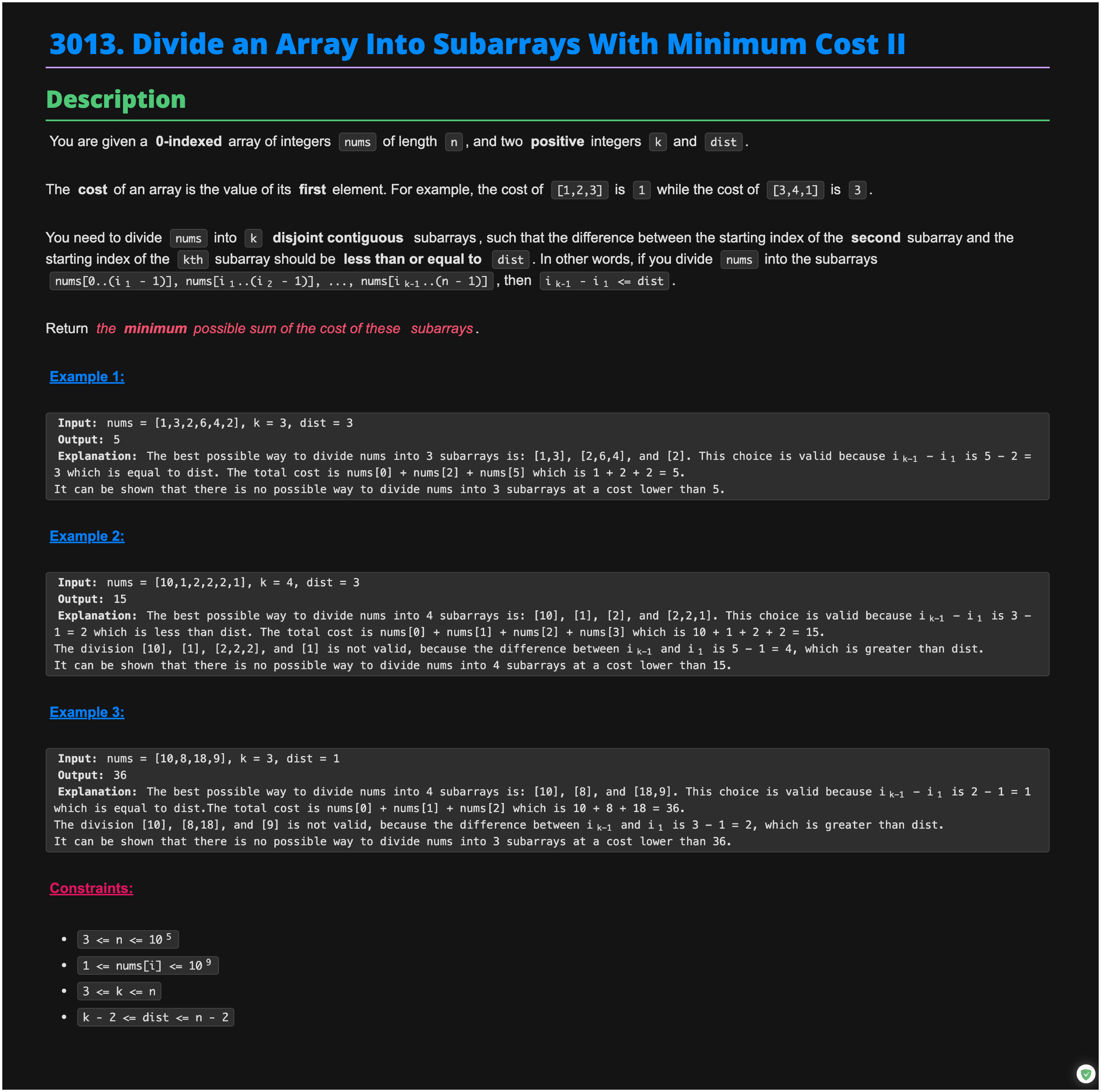

3013. Divide an Array Into Subarrays With Minimum Cost II

Description

You are given a **0-indexed** array of integers `nums` of length `n`, and two **positive** integers `k` and `dist`.

The **cost** of an array is the value of its **first** element. For example, the cost of `[1,2,3]` is `1` while the cost of `[3,4,1]` is `3`.

You need to divide `nums` into `k` **disjoint contiguous** subarrays, such that the difference between the starting index of the **second** subarray and the starting index of the `kth` subarray should be **less than or equal to** `dist`. In other words, if you divide `nums` into the subarrays nums[0..(i_1 - 1)], nums[i_1..(i_2 - 1)], ..., nums[i_{k-1}..(n - 1)], then $i_{k-1} - i_1 <= dist$.

Return *the **minimum** possible sum of the cost of these subarrays*.

Example 1:

```
Input: nums = [1,3,2,6,4,2], k = 3, dist = 3
Output: 5
Explanation: The best possible way to divide nums into 3 subarrays is: [1,3], [2,6,4], and [2]. This choice is valid because i_{k-1} − i_1 is 5 − 2 = 3 which is equal to dist. The total cost is nums[0] + nums[2] + nums[5] which is 1 + 2 + 2 = 5.
It can be shown that there is no possible way to divide nums into 3 subarrays at a cost lower than 5.
```

Example 2:

```
Input: nums = [10,1,2,2,2,1], k = 4, dist = 3
Output: 15
Explanation: The best possible way to divide nums into 4 subarrays is: [10], [1], [2], and [2,2,1]. This choice is valid because i_{k-1} − i_1 is 3 − 1 = 2 which is less than dist. The total cost is nums[0] + nums[1] + nums[2] + nums[3] which is 10 + 1 + 2 + 2 = 15.
The division [10], [1], [2,2,2], and [1] is not valid, because the difference between i_{k-1} and i_1 is 5 − 1 = 4, which is greater than dist.
It can be shown that there is no possible way to divide nums into 4 subarrays at a cost lower than 15.
```

Example 3:

```
Input: nums = [10,8,18,9], k = 3, dist = 1
Output: 36
Explanation: The best possible way to divide nums into 4 subarrays is: [10], [8], and [18,9]. This choice is valid because i_{k-1} − i_1 is 2 − 1 = 1 which is equal to dist.The total cost is nums[0] + nums[1] + nums[2] which is 10 + 8 + 18 = 36.
The division [10], [8,18], and [9] is not valid, because the difference between i_{k-1} and i_1 is 3 − 1 = 2, which is greater than dist.
It can be shown that there is no possible way to divide nums into 3 subarrays at a cost lower than 36.
```

Constraints:

- $3 <= n <= 10^5$
- $1 <= nums[i] <= 10^9$
- `3 <= k <= n`
- `k - 2 <= dist <= n - 2`