Problem 1918: Kth Smallest Subarray Sum

Problem Information

Difficulty: Medium
Acceptance Rate: 52.78%
Paid Only: Yes
Tags: Array, Binary Search, Sliding Window

Problem Description

Given an integer array `nums` of length `n` and an integer `k`, return _the_`kth` _**smallest subarray sum**._

A **subarray** is defined as a **non-empty** contiguous sequence of elements in an array. A **subarray sum** is the sum of all elements in the subarray.

Example 1:

Input: nums = [2,1,3], k = 4 **Output:** 3 **Explanation:** The subarrays of [2,1,3] are: - [2] with sum 2 - [1] with sum 1 - [3] with sum 3 - [2,1] with sum 3 - [1,3] with sum 4 - [2,1,3] with sum 6 Ordering the sums from smallest to largest gives 1, 2, 3, _3_ , 4, 6. The 4th smallest is 3.

Example 2:

Input: nums = [3,3,5,5], k = 7 **Output:** 10 **Explanation:** The subarrays of [3,3,5,5] are: - [3] with sum 3 - [3] with sum 3 - [5] with sum 5 - [5] with sum 5 - [3,3] with sum 6 - [3,5] with sum 8 - [5,5] with sum 10 - [3,3,5], with sum 11 - [3,5,5] with sum 13 - [3,3,5,5] with sum 16 Ordering the sums from smallest to largest gives 3, 3, 5, 5, 6, 8, _10_ , 11, 13, 16. The 7th smallest is 10.

Constraints:

* `n == nums.length` * `1 <= n <= 2 * 104` * `1 <= nums[i] <= 5 * 104` * `1 <= k <= n * (n + 1) / 2`

Code Snippets

C++:

```
class Solution {
public:
int kthSmallestSubarraySum(vector<int>& nums, int k) {

}
};
```

Java:

```
class Solution {
public int kthSmallestSubarraySum(int[] nums, int k) {

}
}
```

Python3:

```
class Solution:
def kthSmallestSubarraySum(self, nums: List[int], k: int) -> int:
```