992. Subarrays with K Different Integers

Given an array `A` of positive integers, call a (contiguous, not necessarily distinct) subarray of `A` *good* if the number of different integers in that subarray is exactly `K`.

(For example, `[1,2,3,1,2]` has `3` different integers: `1`, `2`, and `3`.)

Return the number of good subarrays of `A`.

```
class Solution {
    public int subarraysWithKDistinct(int[] A, int K) {

        // sliding window
        return countsOfSubarraysWithKDistinct(A, K) -
            countsOfSubarraysWithKDistinct(A, K - 1);
    }

    // 计算所有Distinct Number个数小于等于 K 的 SubArray 个数之和
    private int countsOfSubarraysWithKDistinct(int[] A, int K) {
        Map<Integer, Integer> map = new HashMap<>();
        int count = 0, ans = 0;

        for (int left = 0, right = 0; right < A.length; right++) {
            map.put(A[right], map.getOrDefault(A[right], 0) + 1);
            if (map.get(A[right]) == 1) {
                count++;
            }

            while (count > K) {
                map.put(A[left], map.get(A[left]) - 1);
                if (map.get(A[left]) == 0) {
                    count--;
                }
                left++;
            }
            ans += right - left + 1; // ??
        }
        return ans;
    }

}
```