# Count distinct elements in every window

Given an array of integers and a number K. Find the count of distinct elements in every window of size K in the array.

**Example 1:**

```
Input: N = 7, K = 4
A[] = {1,2,1,3,4,2,3}
Output: 3 4 4 3 Explanation: Window 1 of size k = 4 is
1 2 1 3. Number of distinct elements in
this window are 3.
Window 2 of size k = 4 is 2 1 3 4. Number
of distinct elements in this window are 4.
Window 3 of size k = 4 is 1 3 4 2. Number
of distinct elements in this window are 4.
Window 4 of size k = 4 is 3 4 2 3. Number
of distinct elements in this window are 3.
```

**Example 2:**

```
Input: N = 3, K = 2
A[] = {4,1,1}
Output: 2 1
```

**Your Task:**
Your task is to complete the function **countDistinct()** which takes the array A[], the size of the array(N) and the window size(K) as inputs and returns an array containing the count of distinct elements in every contiguous window of size K in the array A[].

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

**Constraints:**

$1 <= K <= N <= 10^5$

$1 <= A[i] <= 10^5$ , for each valid i

```
def countDistinct(self, arr, N, K):
        # Code here
        freq = {}
        for i in range(k):
```

```
            freq[arr[i]] = freq.get(arr[i], 0) + 1
        ans = []
        j = 0
        ans.append(len(freq))
        for i in range(k, len(arr)):
            freq[arr[i]] = freq.get(arr[i], 0) + 1
            temp = arr[j]
            if freq[temp] == 1:
                del freq[temp]
            else:
                freq[temp] = freq[temp] - 1
            ans.append(len(freq))
            j = j + 1
        return ans
```