340. Longest Substring with At Most K Distinct Characters

Given a string, find the length of the longest substring T that contains at most *k* distinct characters.

```
class Solution {
    public int lengthOfLongestSubstringKDistinct(String s, int k) {
        if (s.length() < 1)
            return 0;

        HashMap<Character, Integer> map = new HashMap<Character, Integer>();
        int lo = 0, hi = 0;
        int maxLength = 0;

        while(hi < s.length()) {
            if(map.size() <= k) {
                char c = s.charAt(hi);
                map.put(c, hi);
                hi++;
            }

            if (map.size() > k) {
                // only initialize with a large value
                int leftMost = s.length();

                for (int i: map.values())
                    leftMost = Math.min(leftMost, i);

                char c = s.charAt(leftMost);
                map.remove(c);
                lo = leftMost + 1;
            }

            maxLength = Math.max(maxLength, hi - lo);
        }

        return maxLength;
    }
}
```