

H-Index

Given an array of citations (each citation is a non-negative integer) of a researcher, write a function to compute the researcher's h-index.

According to the [definition of h-index on Wikipedia](#): "A scientist has index h if h of his/her N papers have **at least** h citations each, and the other $N - h$ papers have **no more than** h citations each."

For example, given `citations = [3, 0, 6, 1, 5]`, which means the researcher has 5 papers in total and each of them had received 3, 0, 6, 1, 5 citations respectively. Since the researcher has 3 papers with **at least** 3 citations each and the remaining two with **no more than** 3 citations each, his h-index is 3.

Note: If there are several possible values for h , the maximum one is taken as the h-index.

1. An easy approach is to sort the array first.
2. What are the possible values of h-index?
3. A faster approach is to use extra space.

Credits:

Special thanks to [@jianchao.li.fighter](#) for adding this problem and creating all test cases.

Solution 1

```
public class Solution {
    // 9.3 70 years diaoZhaTian China jiaYou
    public int hIndex(int[] citations) {
        int length = citations.length;
        if (length == 0) {
            return 0;
        }

        int[] array2 = new int[length + 1];
        for (int i = 0; i < length; i++) {
            if (citations[i] > length) {
                array2[length] += 1;
            } else {
                array2[citations[i]] += 1;
            }
        }
        int t = 0;
        int result = 0;

        for (int i = length; i >= 0; i--) {
            t = t + array2[i];
            if (t >= i) {
                return i;
            }
        }
        return 0;
    }
}
```

written by [pythonicjava](#) original link [here](#)

Solution 2

```
public int hIndex(int[] citations) {  
    int len = citations.length;  
    int[] count = new int[len + 1];  
  
    for (int c: citations)  
        if (c > len)  
            count[len]++;  
        else  
            count[c]++;  
  
    int total = 0;  
    for (int i = len; i >= 0; i--) {  
        total += count[i];  
        if (total >= i)  
            return i;  
    }  
  
    return 0;  
}
```

written by [jinrf](#) original link [here](#)

Solution 3

```
public int hIndex(int[] citations) {  
    Arrays.sort(citations);  
    int len=citations.length;  
    for(int i=0;i<len;i++){  
        if(citations[i]>=len-i) return len-i;  
    }  
    return 0;  
}
```

written by [dezhihe](#) original link [here](#)

From [LeetCoder](#).