6271. Maximum Tastiness of Candy Basket

My Submissions (/contest/weekly-contest-325/problems/maximum-tastiness-of-candy-basket/submissions/) Back to Contest (/contest/weekly-contest-325/)

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Medium

You are given an array of positive integers `price` where `price[i]` denotes the price of the i^{th} candy and a positive integer `k`.

The store sells baskets of `k` **distinct** candies. The **tastiness** of a candy basket is the smallest absolute difference of the **prices** of any two candies in the basket.

Return *the **maximum** tastiness of a candy basket.*

Example 1:

```
Input: price = [13,5,1,8,21,2], k = 3
Output: 8
Explanation: Choose the candies with the prices [13,5,21].
The tastiness of the candy basket is: min(|13 - 5|, |13 - 21|, |5 - 21|) = min(8, 8, 16) = 8.
It can be proven that 8 is the maximum tastiness that can be achieved.
```

Example 2:

```
Input: price = [1,3,1], k = 2
Output: 2
Explanation: Choose the candies with the prices [1,3].
The tastiness of the candy basket is: min(|1 - 3|) = min(2) = 2.
It can be proven that 2 is the maximum tastiness that can be achieved.
```

Example 3:

```
Input: price = [7,7,7,7], k = 2
Output: 0
Explanation: Choosing any two distinct candies from the candies we have will result in a tastiness of 0.
```

Constraints:

- `1 <= price.length <= 10^5`
- `1 <= price[i] <= 10^9`
- `2 <= k <= price.length`

JavaScript

```
let a, k, n;
const maximumTastiness = (price, K) => {
    a = price, k = K, n = a.length;
    a.sort((x, y) => x - y);
    let max = Math.max(a[n - 1] - a[0]);
    return BinarySearch(0, max)
};

const BinarySearch = (low, high) => {
    while (low <= high) {
        let mid = low + parseInt((high - low) / 2);
        if (possible(mid)) {
            low = mid + 1;
        } else {
            high = mid - 1;
        }
    }
    return high;
};

const possible = (diff) => {
    let pick = 0, cur = Number.MIN_SAFE_INTEGER;
    for (const x of a) {
        if (x - diff >= cur) {
```

```
            cur = x;
            pick++;
        }
    }
    return pick >= k;
};
```

Custom Testcase | Use Example Testcases

Run | Submit

Submission Result: Accepted (/submissions/detail/865081932/) | More Details > (/submissions/detail/865081932/)

Share your acceptance!