

题目描述：

1.众数是指一组数据中出现次数多的那个数，众数可以是多个

2.中位数是指把一组数据从小到大排列，最中间的那个数，如果这组数据的个数是奇数，那最中间那个就是中位数，如果这组数据的个数为偶数，那就把中间的两个数之和除以 2，所得的结果就是中位数

3.查找整型数组中元素的众数并组成一个新的数组，求新数组的中位数

输入描述：

输入一个一维整型数组，数组大小取值范围 $0 < N < 1000$ ，数组中每个元素取值范围 $0 < E < 1000$

输出描述：

输出众数组成的新数组的中位数

补充说明：

示例 1

输入：

10 11 21 19 21 17 21 16 21 18 15

输出：

21

说明：

示例 2

输入：

2 1 5 4 3 3 9 2 7 4 6 2 15 4 2 4

输出:

3

说明:

示例 3

输入:

5 1 5 3 5 2 5 5 7 6 7 3 7 11 7 55 7 9 98 9 17 9 15 9 9 1 39

输出:

7

说明:

```
const rl = require("readline").createInterface({ input: process.stdin });
var iter = rl[Symbol.asyncIterator]();
const readline = async () => (await iter.next()).value;
```

```
void (async function () {
    // Write your code here
    while ((line = await readline())) {
        const nums=line.split(" ").map(Number);
        const map = new Map();
        let maxTime = 0;

        for (let num of nums) {
            var time = (map.get(num) || 0) + 1;
            maxTime = Math.max(time, maxTime);
            map.set(num, time);
        }

        var values = [...map.keys()]
            .filter((key) => map.get(key) === maxTime)
            .sort((a, b) => a - b);

        if (values.length % 2) {
            console.log(values[(values.length - 1) / 2]);
        } else {
            var leftIndex = values.length / 2 - 1;
            console.log((values[leftIndex] + values[leftIndex + 1]) / 2);
        }
    }
})
```

});