

算法设计与分析第一次上机：编写寻找多数元素程序

学号：21009200158 姓名：游霄童

时间：2023/10/10

代码语言为Python

源代码如下：

```
# 找出数组中超过50%的元素
# Created by 游霄童 2023/10/10

n = int(input("请输入数组的个数n:"))
# 数组的输入
A = []
print("请依次输入数组元素：")
for i in range(n):
    num = input()
    A.append(num)

print("您输入的数组为：", A)
# 思路：使用python的字典实现哈希映射，仅需比较一遍数组的值便可得出结论，算法复杂度为O(n)
hashh = {}

# 初始化
for i in range(n):
    hashh[A[i]] = 0
for i in range(n):
    hashh[A[i]] += 1

flag = 0
for key in hashh:
    if hashh[key] > n / 2:
        flag = 1
        print("存在出现次数大于n/2的元素为：", key, "出现次数为：", hashh[key])
if flag == 0: print("不存在出现次数大于n/2的元素")
```

测试用例结果如下图所示：

测试用例1

- $n=5$
- 1、8、9、9、9

```
D:\CONDA\python.exe D:\游霄童\Python\FindOver50%.py
请输入数组的个数n:5
请依次输入数组元素:
1
8
9
9
9
您输入的数组为: ['1', '8', '9', '9', '9']
存在出现次数大于n/2的元素为: 9 出现次数为: 3

进程已结束，退出代码为 0
```

测试用例2

- $n=10$
- ndsa、dqs、python、python、python、python、python、c++、c++、c++

```
D:\CONDA\python.exe D:\游霄童\Python\FindOver50%.py
请输入数组的个数n:10
请依次输入数组元素:
ndsa
dqs
python
python
python
python
python
python
c++
c++
c++
您输入的数组为: ['ndsa', 'dqs', 'python', 'python', 'python', 'python', 'python', 'python', '++', 'c++', 'c++']
不存在出现次数大于n/2的元素
```

测试用例3

- $n=9$
- English, English, Chinese, Chinese, Chinese, Chinese, Chinese, Spanish, Spanish

```
D:\CONDA\python.exe D:\游霄童\Python\FindOver50%.py
请输入数组的个数n:9
请依次输入数组元素:
English
English
Chinese
Chinese
Chinese
Chinese
Chinese
Chinese
Spanish
Spanish
您输入的数组为: ['English', 'English', 'Chinese', 'Chinese', 'Chinese', 'Chinese', 'Chinese', 'Spanish', 'Spanish']
存在出现次数大于n/2的元素为: Chinese 出现次数为: 5
```

测试用例4（特殊情况）

- $n=0$

```
D:\CONDA\python.exe D:\游霄童\Python\FindOver50%.py
请输入数组的个数n:0
请依次输入数组元素:
您输入的数组为: []
不存在出现次数大于n/2的元素

进程已结束，退出代码为 0
```

算法时间复杂度分析：

代码中共运用单层for循环3次，比较1次，即

$$3n + m$$

m 为输入元素的种类数，那么

$$m \leq n$$

故该算法的O时间复杂度为：

$$O(n)$$