

桶数组思想入门

人员

初锦阳、田心一、李知朔、吴念远、李瑞涵、郭雨宸、孙泽轩、纪博涵、周纪先、赵牧之、吴言恩、亓骏泽、周沁言、郭韩、董岱诚 到课

作业

<https://www.luogu.com.cn/contest/208147> (E F 必做, G H I 选做)

课堂表现

课上有些同学说话比较多，被老师批评了，希望以后可以改正。

课堂内容

P2141 [NOIP2014 普及组] 珠心算测验

需要注意的 2 个问题:

1. 一个数最多被统计 1 次。比如对于一个 7 来说, 他可以被 3+4 构成, 也可以被 2+5 构成, 但是这只算 1 次
2. 不能用一个数 +2 次得到另外一个数

```
int cnt = 0;
for (int i = 1; i <= n; i++) {
    先认为 a[i] 不符合这个性质, 然后去检测一下是否有其它 2 个数相加能得到 a[i]
    如果能, 认为 a[i] 是符合性质

    bool flag = false;
    for (int j = 1; j <= n; j++) {
        for (int k = 1; k <= n; k++) {
            if (j!=k && a[j]+a[k]==a[i]) {
                flag = true;
            }
        }
    }

    if (flag==true) {
        cnt++;
    }
}
cout << cnt << endl;
```

```
#include <iostream>

using namespace std;

int a[105];

int main()
{
    int n;
    cin >> n;
    for (int i = 1; i <= n; i++) {
        cin >> a[i];
    }

    int cnt = 0;
    for (int i = 1; i <= n; i++) {
        bool flag = false;
        for (int j = 1; j <= n; j++) {
            for (int k = 1; k <= n; k++) {
                if (j!=k && a[j]+a[k]==a[i]) {
                    flag = true;
                }
            }
        }

        if (flag == true) {
            cnt++;
        }
    }
    cout << cnt << endl;
    return 0;
}
```

U489769 数字出现次数

桶数组做法

设一个 **cnt数组**, 其中 cnt[i] 代表 i 这个数出现的次数

此时, 每次输入一个 x 之后, 说明 x 出现的次数 +1, 那么让 cnt[x]++

最后, cnt[0] 代表 0 出现的次数, cnt[1] 代表 1 出现的次数, ..., cnt[19] 代表 19 出现的次数

因此, 在 cnt[0] ~ cnt[19] 中找一个最大值即可

```
int cnt[20]; // cnt[i]: i 出现的次数    ->    桶数组

int x;
for (int i = 1; i <= 50; i++) {
    cin >> x; // 输入一个 x, x 出现的次数就多了 1 次, 所以应该让 cnt[x]++
```

```
    cnt[x]++;
}

cout << (cnt[0], cnt[1], cnt[2], cnt[3], ..., cnt[19]) 中的最大值

int maxx = 0;
for (int i = 0; i <= 19; i++) {
    if (cnt[i] > maxx) {
        maxx = cnt[i];
    }
}

cout << maxx << endl;
```

```
#include <iostream>

using namespace std;

int a[20];

int main()
{
    int x;
    for (int i = 1; i <= 50; i++) {
        cin >> x;
        a[x]++;
    }

    int maxx = 0;
    for (int i = 0; i <= 19; i++) {
        if (a[i] > maxx) {
            maxx = a[i];
        }
    }

    cout << maxx << endl;
    return 0;
}
```

U489746 金额改变

```
#include <iostream>

using namespace std;

int a[1005];

int main()
{
```

```
int n;
cin >> n;
for (int i = 1; i <= n; i++) {
    cin >> a[i];
}

int maxx = 0;
for (int i = 1; i <= n; i++) {
    if (a[i] > maxx) {
        maxx = a[i];
    }
}

for (int i = 1; i <= n; i++) {
    if (a[i] == maxx) {
        if (a[i]%2 == 0) {
            a[i] *= 2;
        } else {
            a[i]++;
        }
    }
}

for (int i = 1; i <= n; i++) {
    cout << a[i] << " ";
}
return 0;
}
```

U489748 金额改变2

```
#include <iostream>

using namespace std;

int a[1005];

int main()
{
    int n;
    cin >> n;
    for (int i = 1; i <= n; i++) {
        cin >> a[i];
    }

    for (int i = 1; i <= n/2; i++) {
        a[i] *= 2;
    }
    for (int i = n/2+1; i <= n; i++) {
        a[i]++;
    }
}
```

```
    for (int i = 1; i <= n; i++) {  
        cout << a[i] << " ";  
    }  
    return 0;  
}
```

U489751 金额改变3

```
#include <iostream>  
  
using namespace std;  
  
int a[1005];  
  
int main()  
{  
    int n;  
    cin >> n;  
    for (int i = 1; i <= n; i++) {  
        cin >> a[i];  
    }  
  
    int maxx = 0, minn = 1000000;  
    for (int i = 1; i <= n; i++) {  
        if (a[i] > maxx) {  
            maxx = a[i];  
        }  
        if (a[i] < minn) {  
            minn = a[i];  
        }  
    }  
  
    for (int i = 1; i <= n; i++) {  
        if (a[i] == maxx) {  
            a[i] *= 2;  
        }  
        else if (a[i] == minn) {  
            a[i]++;  
        }  
    }  
  
    for (int i = 1; i <= n; i++) {  
        cout << a[i] << " ";  
    }  
    return 0;  
}
```

U492925 判断数组是否对称

```
#include<cstdio>
#include<iostream>
using namespace std;

int main() {
    int n;
    cin>>n;
    int a[n];
    bool b=true;
    for (int i=1;i<=n;i++) {
        cin>>a[i];
    }
    for (int i=1;i<=n;i++) {
        if (a[i]!=a[n+1-i]) {
            b=false;
        }
    }
    if (b==true) {
        cout<<"YES";
    }
    else {
        cout<<"NO";
    }
}
```

U492931 选扑克牌

枚举 左边的数 和 右边的数, 把 2 个数的和记录在 桶 里, 最后看有几个数对应的 桶 的值 大于0 即可

```
#include <iostream>

using namespace std;

int a[55];
int f[30];

int main()
{
    int n;
    cin >> n;
    for (int i = 1; i <= n; i++) {
        cin >> a[i];
    }

    for (int i = 1; i <= n; i++) {
        for (int j = i+1; j <= n; j++) {
            f[a[i]+a[j]]++;
        }
    }
}
```

```
int cnt = 0;
for (int i = 2; i <= 26; i++) {
    if (f[i] > 0) {
        cnt++;
    }
}

cout << cnt << endl;
for (int i = 2; i <= 26; i++) {
    if (f[i] > 0) {
        cout << i << " ";
    }
}
return 0;
}
```

U489764 修建地铁

开一个 bool 类型的 a 数组

- 当 `a[i] == false` 的时候, 说明第 `i` 个位置没砍
- 当 `a[i] == true` 的时候, 说明第 `i` 个位置被砍了

最后从 `0 ~ L` 统计有几个位置的树没被砍即可

```
#include <iostream>

using namespace std;

bool a[10005];

int main()
{
    int L, n;
    cin >> L >> n;
    for (int i = 1; i <= n; i++) {
        int l, r;
        cin >> l >> r;
        for (int j = l; j <= r; j++) {
            a[j] = true;
        }
    }

    int cnt = 0;
    for (int i = 0; i <= L; i++) {
        if (a[i] == false) {
            cnt++;
        }
    }
}
```

```
    }  
    cout << cnt << endl;  
    return 0;  
}
```

U489773 兴趣班

开一个 a 数组, $a[i]$ 代表第 i 天来了几个人

如果一个人每隔 x 天来一次, 那么让 $a[1]++$, $a[x+1]++$, $a[2*x+1]++$, ...

最后, 在 $a[1] \sim a[100000]$ 中遍历一下, 看在哪个位置 $a[i]$ 最先等于 n 即可

```
#include <iostream>  
  
using namespace std;  
  
int a[100005];  
  
int main()  
{  
    int n;  
    cin >> n;  
    for (int i = 1; i <= n; i++) {  
        int x;  
        cin >> x;  
        for (int j = 1; j <= 100000; j += x) {  
            a[j]++;  
        }  
    }  
  
    for (int i = 2; i <= 100000; i++) {  
        if (a[i] == n) {  
            cout << i << endl;  
            break;  
        }  
    }  
    return 0;  
}
```