

石头剪刀布精讲 + 初学桶标记法

人员

温郝冬、李沛都、罗启宸、吴青汉、王奕皓、周苇杰、陶汇笙、张昱霖、康佳、锁喜栋、郭栩睿到课

作业

东方博宜 1334 - 扑克牌组合

要求在纸上写出分析过程，第一步做什么、第二步做什么、第三步做什么

课堂表现

课上一部分同学听课不认真，老师讲过的内容完全没听到，需要老师重复讲好几次

课堂内容

东方博宜 1180 - 数字出现次数

对于这个题目来说，可以定义一个 $a[20]$ 的数组

$a[0]$ 代表0的次数

$a[1]$ 代表1的次数

...

$a[19]$ 代表19的次数

最开始的时候， $a[0]=a[1]=a[2]=\dots=a[19]=0$ ，每个数都没有出现过，出现次数都是0

之后，循环50次，每次输入一个数 x ，则说明 x 多出现了一次，可以让 $a[x]++$

最后，50次循环结束之后， $a[0]$ 就是0出现的次数， $a[1]$ 就是1出现的次数，...， $a[19]$ 就是19出现的次数

想要找到出现最多的次数为多少次，就是在 $a[0]$ ， $a[1]$ ， $a[2]$ ，...， $a[19]$ 这个数组中，找到这个数组的最大值

```

#include <iostream>

using namespace std;

int a[20];

int main() {
    int x;
    for (int i = 1; i <= 50; i++) {
        cin >> x;
        a[x]++;
    }
    int t = 0;
    for (int i = 0; i <= 19; i++) {
        if (a[i] > t) {
            t = a[i];
        }
    }
    cout << t << endl;
    return 0;
}

```

东方博宜 1333 兴趣班的排班

利用桶标记法的思想，开一个f数组，其中 $f[i]$ 代表第 i 天有多少名学生一起上课

假设有 3 名学生，3 名学生分别每 3/2/4 天上一次课

则对于第一名学生来说：

第1天上课、第4天上课、第7天上课、第10天上课、第13天上课、第16天上课、...

可以让 $f[1]++$, $f[4]++$, $f[7]++$, $f[10]++$, $f[13]++$, $f[16]++$, ...

对于第二名学生：则是 $f[1]++$, $f[3]++$, $f[5]++$, $f[7]++$, $f[9]++$, $f[11]++$, $f[13]++$, $f[15]++$, $f[17]++$, ...

对于第三名学生：是 $f[1]++$, $f[5]++$, $f[9]++$, $f[13]++$, $f[17]++$, ...

最终 $f[1]=3$, $f[2]=0$, $f[3]=1$, ..., $f[13]=3$

说明第1天有3个人，第2天有0个人，第3天有1个人，..., 第13天有3个人

所以，在第1天之后，在第13天可以有 3 个人同时上课

```

#include <bits/stdc++.h>

using namespace std;

int w[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) {
            w[j]++;
        }
    }
    for (int i = 2; i <= 100000; i++) {
        if (w[i] == n) {
            cout << i << endl;
            return 0;
        }
    }
    return 0;
}

```

noi 1.5 42:画矩形

n, m, c, k

n行m列的字符c

如果是实心:

```
for (int i = 1; i <= n; i++) {           // 行
    for (int j = 1; j <= m; j++) {       // 列
        cout << c;
    }
    cout << endl;
}
```

如果是空心:

循环:

```
if (位置在 最左边 或者 最右边 或者 最上边 或者 最下边) {
    cout << c;
} else {
    cout << " ";
}
```

最左边 -> j==1

最右边 -> j==m

最上边 -> i==1

最下边 -> i==n

```
for (int i = 1; i <= n; i++) {           // 行
    for (int j = 1; j <= m; j++) {       // 列
        if (在边缘) {
            cout << c;
        } else {
            cout << " ";
        }
    }
    cout << endl;
}
```

```

#include <iostream>

using namespace std;

int main() {
    int n, m, k;
    char c;
    cin >> n >> m >> c >> k;
    if (k == 1) {
        for (int i = 1; i <= n; ++i) {
            for (int j = 1; j <= m; ++j) {
                cout << c;
            }
            cout << endl;
        }
    } else {
        for (int i = 1; i <= n; ++i) {
            for (int j = 1; j <= m; ++j) {
                if (i==1 || i==n || j==1 || j==m) {
                    cout << c;
                } else {
                    cout << " ";
                }
            }
            cout << endl;
        }
    }
    return 0;
}

```

东方博宜 1334 扑克牌组合

定义数组 `f[k]`: `k`这个数出现的次数

两两匹配, 统计哪些数出现过, 让`f[k]++`

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

计数器统计一共有多少个数出现过

```
int sum = 0;  
for (int i = 1; i <= 100; i++) {  
    if (f[i] >= 1) {  
        sum++;  
    }  
}  
cout << sum << endl;
```

输出所有出现过的数

```
for (int i = 1; i <= 100; i++) {  
    if (f[i] >= 1) {  
        cout << i;  
    }  
}
```