

# 一维数组2

## 人员

韩承睿、徐浩然、牟茗、辛帅辰、高健桓、崔宸赫、崔吉诺、刘祺、方俊喆、刘峰烁 到课，刘嘉航、李翰如 线上

## 作业

东方博宜 1886 - 连胜统计  
东方博宜 1540 - 小X数字母

## 课堂表现

有同学们课上存在看题解的问题，以后要求所有同学注意，养成独立思考的习惯，不许看题解

## 课堂内容

### noi 1.6 08:石头剪刀布

难点分析：

1. 输入0 2 5这个数组并不难，如何把这个数组变成0 2 5 0 2 5 0 2 5 0
2. 两个数字的比较，什么时候A赢，什么时候B赢

难点1解决方案：

输入a[1] ~ a[na],之后让 a[na+1]=a[1], a[na+2] = a[2], ..., a[n] = a[n-na]

数组b可以采用相同的处理方案

这样处理之后，a[1]~a[n]/b[1]~b[n] 即为a、b两个人这n轮每轮的结果

然后比较a[1],b[1]    a[2],b[2]    a[3],b[3]    ...    a[n],b[n] 即可

难点2解决方案：

对于第i次比较时，a出的是a[i]，b出的是b[i]

```
if (a[i] == b[i]) { // 平手
```

```
} else if (a赢) { // a赢
```

```
} else { // b赢
```

```
}
```

`continue`: 结束本次循环，进入下一次循环

## 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 1886 - 连胜统计

12

```
    9 9 1 2 2 2 9 9 3 3 3 4
cnt: 1 2 1 1 2 3 1 2 1 2 3 1
    x √ x x x √ x √ x x x √
```

难点:

1. 怎么控制cnt的变化

-> 根据 `a[i]` 与 `a[i-1]` 是否相同

-> 相同: `cnt++`

-> 不同: `cnt=1`

2. 在哪些地方输出

-> 当遇见某个位置的数 与 前面位置的数 不相同时, 并且`cnt>=2`时, 输出前面的数, 以及cnt

此题不提供具体代码, 要求同学们课下思考实现