

# 第一部分

主讲人：计试2101杨思成

## 1.输出索引

键盘输入一个数组，找到两个总和为特定值的索引。

例如给定数组 [1, 2, 3, -2, 5, 7]，给定总和 7，则返回索引 [1, 4]。

若有多组符合情况则输出索引对中小索引最小的任意一组。

样例

输入：第一个数是数量，第二行是数组元素输入，第三行是给定总和  
6  
1 2 3 -2 5 7  
7

输出：（逗号前后没有空格）  
[1,4]

注意：数组要求使用动态内存分配

考察：标记的使用，循环体

```
#include<bits/stdc++.h>
using namespace std;
int main(){
    int n;
    cin>>n;
    int* a = new int[n];

    for(int i = 0;i<n;i++)
        cin>>a[i];

    int s;
    cin>>s;
```

```

int flag = 0;

for(int i = 0;i<n;i++){
    for(int j = i+1;j<n;j++){

        if((a[i]+a[j] == s)){
            cout << '[' << i << ',' << j << ']'<<endl;
            flag = 1;
            break;
        }
    }

    if(flag)break;
}

if(!flag) cout<<"Wrong input!"<<endl;
delete []a;
return 0;
}

```

## 2.开灯问题

有 $n$ 盏灯，编号为 $1 \sim n$ ，第1个人把所有灯打开，第2个人按下所有编号为2的倍数的开关（这些灯将被关掉），第3个人按下所有编号为3的倍数的开关（其中关掉的灯将被打开，开着的灯将被关闭），依此类推。一共有 $k$ 个人，问最后有哪些灯开着？输入： $n$ 和 $k$ ，输出开着的灯编号。 $k \leq n \leq 1000$

输入

输入一组数据: $n$ 和 $k$

输出

输出开着的灯编号

样例输入

7 3

样例输出

1 5 6 7

考察：01标记与异或运算

```
#include<iostream>
using namespace std;
int main()
{
    int n, k, i;
    cin>>n>>k;

    int array[n];
    for(i = 0; i < n; i++)
        array[i] = 1;

    for(i = 2; i <= k; i++)
    {
        for(int p = 1; p < n; p++)
        {
            if((p+1) % i == 0)
                array[p] = !array[p];
        }
    }

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

```
#include<bits/stdc++.h>

int a[1007];

int main(){
    int n, k;
    std::cin >> n >> k;
    for(int i = 1; i <= k; ++i)
        for(int j = i; j <= n; j+=i)
            a[j]^=1;
    for(int i = 1; i <= n; ++i)
        if(a[i])std::cout<< i << ' ';
    return 0;
}
```

### 3.字符串统计

键盘输入一段文字，输出其中出现次数最多的词及其出现次数。

如果有多个词出现次数相同，则输出最小的那个词及其次数。

注意：此题大小写敏感;有可能输入多行，输入结束时没有回车

样例

输入：  
He is a teacher and his teacher is his father  
输出：  
(中间一个空格)  
his 2

考察：while(cin),简单桶排序

```
#include<bits/stdc++.h>
using namespace std;
#define max_len 100

string save_str[max_len];

int cnt_str[max_len];

int pos;
int main(){
    string s;
    while(cin>>s){
```

```
int flag = 0;
for(int i = 0; i < pos; i++){
    if(s == save_str[i]){
        cnt_str[i]++;
        flag = 1;
        break;
    }
}

if(!flag){
    save_str[pos] = s;
    cnt_str[pos]++;
    pos++;
}

}
```

```
int p = 0;
for(int i = 1; i < pos; i++){
    bool f1 = (cnt_str[i] > cnt_str[p]);
    bool f2 = (cnt_str[i] == cnt_str[p]) && (save_str[i] < save_str[p]);

    if(f1 || f2) p = i;
}
```

```
cout << save_str[p] << ' ' << cnt_str[p] << '\n';
return 0;
}
```