

## 8.1-1 初试数组

- 如何写一个程序计算用户输入的数字的平均数？

```
int x;  
double sum = 0;  
int cnt = 0;  
scanf("%d", &x);  
while ( x != -1 ) {  
    sum += x;  
    cnt ++;  
    scanf("%d", &x);  
}  
if ( cnt > 0 ) {  
    printf("%f\n", sum/cnt);  
}
```

- 不需要记录输入的每一个数

- 如何写一个程序计算用户输入的数的平均数，并输出所有大于平均数的数？
- 必须先记录每一个输入的数，计算平均数之后，再检查记录下来的每一个数，与平均数比较，决定是否输出

# 如何记录很多数？

- `int num1,num2,num3.....?`

# 数组

- `int number[100];`
- `scanf("%d", &number[i]);`



# 数组

```
int x;  
double sum = 0;  
int cnt = 0;  
int number[100];  
scanf("%d", &x);  
while ( x != -1 ) {  
    number[cnt] = x;  
    sum += x;  
    cnt ++;  
    scanf("%d", &x);  
}  
if ( cnt > 0 ) {  
    int i;  
    double average = sum / cnt;  
    for ( i=0; i<cnt; i++ ) {  
        if ( number[i] > average ) {  
            printf("%d ", number[i]);  
        }  
    }  
}
```

定义数组

对数组中的元素  
赋值

这个程序存在安全隐患，是什么？

使用数组中的元素

遍历数组

## 8.1-2 数组的定义和使用



# 定义数组

- `<类型> 变量名称[元素数量];`
  - `int grades[100];`
  - `double weight[20];`
- 元素数量必须是整数
- C99之前：元素数量必须是编译时刻确定的字面量

# 数组

- 是一种容器（放东西的东西），特点是：
  - 其中所有的元素具有相同的数据类型；
  - 一旦创建，不能改变大小
  - \*（数组中的元素在内存中是连续依次排列的）

# int a[10]

- 一个int的数组
- 10个单元：a[0],a[1],...,a[9]



- 每个单元就是一个int类型的变量
- 可以出现在赋值的左边或右边：
  - $a[2] = a[1] + 6;$
- \* 在赋值左边的叫做左值

# 数组的单元

- 数组的每个单元就是数组类型的一个变量
- 使用数组时放在[]中的数字叫做下标或索引，下标从0开始计数：
  - `grades[0]`
  - `grades[99]`
  - `average[5]`

# 有效的下标范围

- 编译器和运行环境都不会检查数组下标是否越界，无论是对数组单元做读还是写
- 一旦程序运行，越界的数组访问可能造成问题，导致程序崩溃
- **segmentation fault**
- 但是也可能运气好，没造成严重的后果
- 所以这是程序员的责任来保证程序只使用有效的下标值：[0, 数组的大小 - 1]

```
int x;
double sum = 0;
int cnt = 0;
int number[100];
scanf("%d", &x);
while ( x != -1 ) {
    number[cnt] = x;
    sum += x;
    cnt++;
    scanf("%d", &x);
}
if ( cnt > 0 ) {
    int i;
    double average = sum / cnt;
    for ( i=0; i<cnt; i++ ) {
        if ( number[i] > average ) {
            printf("%d ", number[i]);
        }
    }
}
```

- 这个程序是危险的，因为输入的数据可能超过100个



# 计算平均数

- 如果先让用户输入有多少数字要计算，可以用C99的新功能来实现

```
int x;  
double sum = 0;  
int cnt;  
printf("请输入数字的数量: ");  
scanf("%d", &cnt);  
if ( cnt>0 ) {  
    int number[cnt];  
    scanf("%d", &x);  
    while ( x!= -1 ) {  
        number[cnt] = x;  
        sum += x;  
        cnt ++;  
        scanf("%d", &x);  
    }  
}
```

C99 ONLY!

# 长度为0的数组?

- `int a[0];`
- 可以存在，但是无用

## 8.1-3 数组的例子：投票统计

- 写一个程序，输入数量不确定的[0,9]范围内的整数，统计每一种数字出现的次数，输入-1表示结束



```
const int number = 10;
```

数组的大小

C99

```
int x;
```

```
int count[number];
```

定义数组

```
int i;
```

```
for ( i=0; i<number; i++ ) {  
    count[i] = 0;  
}
```

初始化数组

```
scanf("%d", &x);
```

```
while ( x!= -1 ) {
```

```
    if ( x>=0 && x<=9 ) {
```

```
        count[x]++;
```

数组参与运算

```
    }
```

```
    scanf("%d", &x);
```

```
}
```

```
for ( i=0; i<number; i++ ) {
```

```
    printf("%d:%d\n", i, count[i]);
```

遍历数组输出

```
}
```