

初试数组

- 如何写一个程序计算用户输入的数字的平均数？

```
Scanner in = new Scanner(System.in);
int x;
double sum = 0;
int cnt = 0;
x = in.nextInt();
while ( x != -1 )
{
    sum += x;
    cnt ++;
    x = in.nextInt();
}
if ( cnt > 0 )
{
    System.out.println(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字的平均数，并输出所有大于平均数的数？

- 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.....?`

```
int x;
double sum = 0;
int cnt = 0;
int[] numbers = new int[100];
x = in.nextInt();
while ( x != -1 )
{
    numbers[cnt] = x;
    sum += x;
    cnt ++;
    x = in.nextInt();
}
if ( cnt > 0 )
{
    double average = sum/cnt;
    for ( int i=0; i<cnt; i++ )
    {
        if ( numbers[i] > average )
        {
            System.out.println(numbers[i]);
        }
    }
}
```



```
int x;  
double sum = 0;  
int cnt = 0;  
int[] numbers = new int[100];  
x = in.nextInt();  
while ( x != -1 )  
{  
    numbers[cnt] = x;  
    sum += x;  
    cnt ++;  
    x = in.nextInt();  
}  
if ( cnt > 0 )  
{  
    double average = sum/cnt;  
    for ( int i=0; i<cnt; i++ )  
    {  
        if ( numbers[i] > average )  
        {  
            System.out.println(numbers[i]);  
        }  
    }  
}  
}
```

定义数组

```
int x;  
double sum = 0;  
int cnt = 0;  
int[] numbers = new int[100];  
x = in.nextInt();  
while ( x != -1 )  
{  
    numbers[cnt] = x;  
    sum += x;  
    cnt ++;  
    x = in.nextInt();  
}  
if ( cnt > 0 )  
{  
    double average = sum/cnt;  
    for ( int i=0; i<cnt; i++ )  
    {  
        if ( numbers[i] > average )  
        {  
            System.out.println(numbers[i]);  
        }  
    }  
}  
}
```

定义数组

对数
组中的元

```
int x;  
double sum = 0;  
int cnt = 0;  
int[] numbers = new int[100];
```

定义数组

```
x = in.nextInt();  
while ( x != -1 )
```

对数
组中的元

```
{  
    numbers[cnt] = x;  
    sum += x;  
    cnt ++;  
    x = in.nextInt();  
}
```

```
if ( cnt > 0 )
```

```
{  
    double average = sum/cnt;  
    for ( int i=0; i<cnt; i++ )  
    {  
        if ( numbers[i] > average )  
        {  
            System.out.println(numbers[i]);  
        }  
    }  
}
```

使用
数组中的

```
int x;  
double sum = 0;  
int cnt = 0;  
int[] numbers = new int[100];
```

定义数组

```
x = in.nextInt();  
while ( x != -1 )
```

对数
组中的元

```
{  
    numbers[cnt] = x;
```

```
    sum += x;
```

```
    cnt ++;
```

```
    x = in.nextInt();
```

```
}
```

```
if ( cnt > 0 )
```

```
{
```

```
    double average = sum/cnt;
```

```
    for ( int i=0; i<cnt; i++ )
```

```
    {
```

```
        if ( numbers[i] > average )
```

```
        {
```

```
            System.out.println(numbers[i]);
```

```
        }
```

```
    }
```

```
}
```

遍历数组

使用
数组中的


```
int x;  
double sum = 0;  
int cnt = 0;  
int[] numbers = new int[100];
```

定义数组

```
x = in.nextInt();  
while ( x != -1 )
```

对数
组中的元

```
{  
    numbers[cnt] = x;
```

```
    sum += x;
```

```
    cnt ++;
```

```
    x = in.nextInt();
```

这个程序存在安全隐患，是什么？

```
}  
if ( cnt > 0 )
```

遍历数组

```
{  
    double average = sum/cnt;
```

```
    for ( int i=0; i<cnt; i++ )
```

```
    {
```

```
        if ( numbers[i] > average )
```

```
        {
```

```
            System.out.println(numbers[i]);
```

```
        }
```

```
    }
```

使用
数组中的

```
}
```

创建数组

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
- 数组中的每个数据叫做元素

数组

```
int[] numbers = new int[4];
```

```
numbers[0] = 101;
```

```
numbers[1] = 232;
```

```
numbers[2] = 231;
```

```
numbers[3] = 0;
```


定义数组变量

- `<类型>[] <名字>= new <类型>[元素个数];`
- `int[] grades= new int[100];`
- `double[] averages = new double[20];`
- 元素个数必须是整数
- 元素个数必须给出
- 元素个数可以是变量

int[] a= new int[10]

- 创建了一个int型的数组
- 10个元素: a[0],a[1],...,a[9]



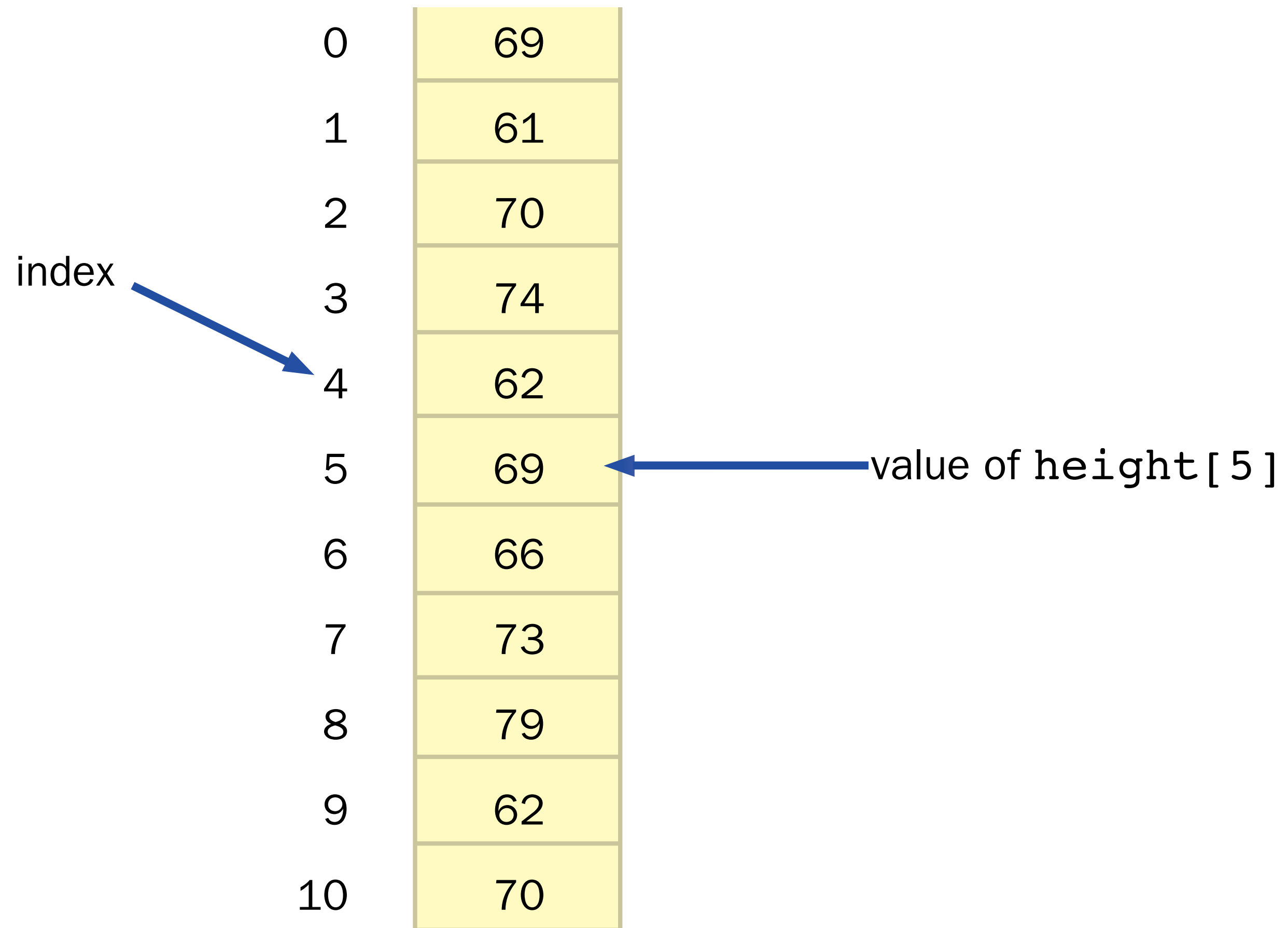
- 每个元素是一个int的变量
- 可以读和写:
 - a[2] = a[1]+6;

数组元素

数组的元素

- 每个元素都是那种类型的变量
- 索引或下标是从0开始的
 - `grades[0]`
 - `grades[99]`
 - `average[5]`

height[8]



0	69
1	61
2	70
3	74
4	62
5	69
6	66
7	73
8	79
9	62
10	70

index

value of height[5]

有效的下标

- 最小的下标是0，最大的下标是数组的元素个数-1
- 可是编译器不会检查你是否用了有效的下标
- 但是如果运行的时候出现了无效的下标，可能会导致程序终止

```
int x;
double sum = 0;
int cnt = 0;
int[] numbers = new int[100];
x = in.nextInt();
while ( x != -1 )
{
    numbers[cnt] = x;
    sum += x;
    cnt ++;
    x = in.nextInt();
}
if ( cnt > 0 )
{
    double average = sum/cnt;
    for ( int i=0; i<cnt; i++ )
    {
        if ( numbers[i] > average )
        {
            System.out.println(numbers[i]);
        }
    }
}
```

- 这个程序是危险的，因为输入的数据可能超过100个

定义数组变量

- `<类型>[] <名字>= new <类型>[元素个数];`
- `int[] grades= new int[100];`
- `double[] averages = new double[20];`
- 元素个数必须是整数
- 元素个数必须给出
- 元素个数可以是变量

计算平均数

- 如果先让用户输入有多少数字要计算，就可以根据这个数字来制造数组

```
double sum = 0;
int cnt = in.nextInt();
if ( cnt>0 )
{
    int[] numbers = new int[cnt];
    for ( int i=0; i<cnt; i++ )
    {
        numbers[i] = in.nextInt();
        sum += numbers[i];
    }
    double average = sum/cnt;
    for ( int i=0; i<cnt; i++ )
    {
        if ( numbers[i] > average )
        {
            System.out.println(numbers[i]);
        }
    }
}
```

length

- 每个数组有一个内部成员length，会告诉你它的元素的数量

```
for ( i=0; i<100; ++i )
```

```
    sum += grade[i];
```

最好是：

```
for ( i=0; i<grade.length; ++i )
```

```
    sum += grade[i];
```

数组的例子：投票统计

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

```
int[] numbers = new int[10];
int x;
x = in.nextInt();
while ( x != -1 )
{
    if ( x>=0 && x<=9 )
    {
        numbers[x]++;
    }
    x = in.nextInt();
}
for ( int i=0; i<numbers.length; i++ )
{
    System.out.println(i+": "+numbers[i]);
}
```

```
int[] numbers = new int[10];
```

创建

```
int x;
```

```
x = in.nextInt();
```

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

```
{
```

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

```
    {
```

```
        numbers[x]++;
```

```
    }
```

```
    x = in.nextInt();
```

```
}
```

```
for ( int i=0; i<numbers.length; i++ )
```

```
{
```

```
    System.out.println(i+": "+numbers[i]);
```

```
}
```

```
int[] numbers = new int[10];
```

创建

```
int x;
```

```
x = in.nextInt();
```

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

```
{
```

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

```
    {
```

```
        numbers[x]++;
```

数组

```
    }
```

```
    x = in.nextInt();
```

```
}
```

```
for ( int i=0; i<numbers.length; i++ )
```

```
{
```

```
    System.out.println(i+": "+numbers[i]);
```

```
}
```



```
int[] numbers = new int[10];
```

创建

```
int x;
```

```
x = in.nextInt();
```

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

```
{
```

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

```
    {
```

```
        numbers[x]++;
```

```
    }
```

```
    x = in.nextInt();
```

```
}
```

```
for ( int i=0; i<numbers.length; i++ )
```

```
{
```

```
    System.out.println(i+": "+numbers[i]);
```

```
}
```

数组

遍历