

千锋HTML5学院

第二阶段javascript课程课件



1

数组的概念

2

一维数组的使用

3

数组常用的冒泡排序的算法

4

数组的常用函数

为什么使用数组

当我们需要表示一组数据，或者叫做一次性定义很多相似的变量时，就需要使用数组，如：表示一个班级学生的成绩，一年十二个月的销售数据等等。

数组的概念

□数组的概念

数组就是一组数（一般来说，数据类型一样），相当于一次性定义了很多相同数据类型的变量，如果说单个变量是一个小容器，数组就是有多个格档的大容器（每个格档是个小容器）。

数组是一种数据类型。

一维数组（变量）的定义

□一维数组的定义

`var arr= new Array(10);` //创建一个包含 10 个元素的数组，没有赋值

`var arr2= new Array("芙蓉姐姐",30);` //使用构造数组创建

`var arr3 = [1,2,3,4,5];` //创建一个包含5个元素的数组，并给每个元素分别赋值为1,2,3,4,5

□使用数组元素；

`arr[0]`:表示数组的第一个元素，0是下标，也叫索引

`arr[1]`:表示数组的第二个元素，1是下标

.....

数组的几个名词

□ 数组的长度（**length**属性）：

数组元素的个数

□ 数组的下标：

下标就是索引，即元素的序号，从**0**开始，下标最大取值是：长度-**1**；

下标可以是变量或表达式。

数组的赋值

□ 数组的赋值

给数组赋值，就是给数组的元素赋值，需要通过给数组的每个元素一一赋值，

如：**arr[0] = 20;**//让数组的第一个元素的值为**20**;

arr[1] = 12;//让数组的第二个元素的值为**12**;

//以下为通过循环给数组的每个元素赋值，赋成下标的平方。

```
for(var i=0;i<10;i++){
```

```
    arr[i] = i*i
```

```
}
```

//以下为通过循环给数组的每个元素赋值，赋成下标的平方。

```
for(var i=0;i<10;i++){
```

```
    arr[i] = Math.random();
```

```
}
```

数组的使用

□不能一次使用整个数组，使用数组就是在使用数组的每个元素，因为数组相当于若干个相同类型的变量。

□遍历数组：

■普通for循环

```
for(var i=0; i<5; i++){  
    document.write(arr[i]);  
}
```

■for each循环

```
for(var i in arr){  
    document.write(arr[i]);  
}
```


示例

- 求数组中所有元素的和，平均值；
- 求一组数的最大数和最小数
- 判断一个数组里的数是不是回文数组（回文数：第一个数和最后一个数相等，第二个数和倒数第二个数相等.....）
- 请将数组`a[0.....9]`中的元素循环右移1位。

冒泡算法

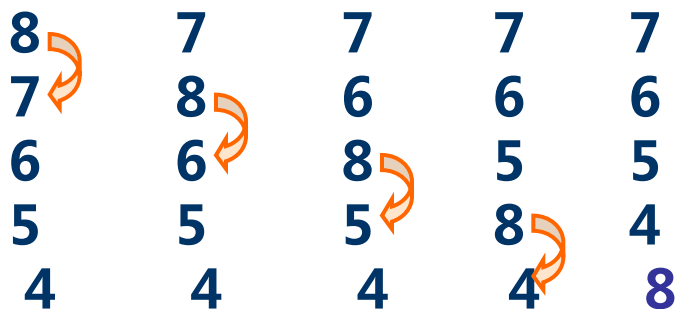
□ 用冒泡排序，对输入的6个数进行排序

■ 思路：输入6个无序的数字，从头到尾依次比较相邻两个数字大小，若大数在前、小数在后，则交换两数位置，依次比较，使全部数据按从小到大排列



冒泡算法

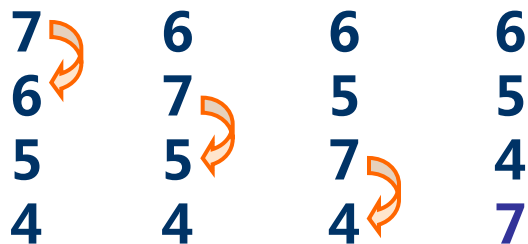
第二趟



第1次 第2次 第3次 第4次 结果

7
6
5
4
8
9

第三趟



第1次 第2次 第3次 结果

6
5
4
7
8
9

冒泡算法

第四趟

6
5
4



5

6
4



5
4
6

第1次 第2次 结果

5
4
6
7
8
9

第五趟

5
4



4
5

第1次 结果

4
5
6
7
8
9

示例

□ 请将数组 $a[0.....9]$ 中的元素循环右移 k 位。

□ 插数: 有一个已经排好序的数组。现输入一个数, 要求按原来的规律将它插入数组中。

常见算法（数组应用扩展知识）

□用选择法对若干个数组排序（由小到大）。

选择法：通过比较首先选出最小的数放在第一个位置上，然后在其余的数中选出次小数放在第二个位置上，依此类推，直到所有的数成为有序序列。

□用折半查找法在一组排好序(递增有序或递减有序)的值中查找某个数据。

■折半查找的基本思想：

■首先将待查数据 k 与排好序（递增有序）的一组数据的中间位置上的数据进行比较，若相等，则查找成功；

■若 $k > a[mid]$ ，则待查数据 k 只可能出现在右半部 $a[mid+1...n]$ 中，则应在这个右半部中再进行折半查找；

■若 $k < a[mid]$ ，则待查数据 k 只可能出现在左半部 $a[1...mid-1]$ 中，则应在这个左半部中再进行折半查找；

■这样通过逐步缩小查找范围，直到找到或找不到该数据 k 为止

常见算法（数组应用扩展知识）

□ 选择法：通过比较首先选出最小的数放在第一个位置上，然后在其余的数中选出次小数放在第二个位置上,依此类推,直到所有的数成为有序序列。

如：已知原始数据： [47 33 61 82 72 11 25 47]

第一趟排序后： 11 [33 61 82 72 47 25 47]

第二趟排序后： 11 25 [61 82 72 47 33 47]

第三趟排序后： 11 25 33 [82 72 47 61 47]

第四趟排序后： 11 25 33 47 [72 82 61 47]

第五趟排序后： 11 25 33 47 47 [82 61 72]

第六趟排序后： 11 25 33 47 47 61 [82 72]

第七趟排序后： 11 25 33 47 47 61 72 [82]

最后排序结果： 11 25 33 47 47 61 72 82

□ 折半查找

查找k=21的过程:

3. 05, 13, 19, 21, 37, 56, 64, 75, 80, 88, 92

 ↑ ↑

 low hig

 mid

 3 4

 查找成功

查找成功

数组的常用操作（函数）

□ 常用函数：

■ 追加元素：

`arr.push('BJ');` // 数组末尾添加一个元素，也可以添加多个

■ 插入元素：

`arr.unshift('TJ');` // 数组开头添加一个元素，也可以添加多个

■ 指定位置插入元素：

`arr.splice(2, 0, 'three');` // 在数组下标为2的位置插入three

■ 删除元素

`var str = arr.pop();` // 移除末尾一个元素，并返回该元素

`var str = arr.shift();` // 移除开头一个元素，并返回该元素

```
var arr = new Array("芙蓉姐姐");  
arr.push("凤姐追随芙蓉","红太郎追随凤姐");  
arr.unshift("春哥在头","猪猪侠跟随春哥","灰太狼跟随猪猪侠");  
for(var i in arr){  
    document.write(arr[i]+"<br>");  
}
```

示例

□插数:有一个已经排好序的数组。现输入一个数，要求按原来的规律将它插入数组中。

数组的常用操作（函数）

□ 常用函数：

- 合并两个数组 - `concat()`
- 用数组的元素组成字符串 - `join()`
- - `sort()`
- - `sort()`

学会看API文档
Application interface

```
var arr = new Array(3)
arr[0] = "George "
arr[1] = "John "
arr[2] = "Thomas"
document.write(arr.join("."))
```

示例

必做：

1. 找出所有1-100之间7的倍数和包含7的数字，放入到数组中。
2. 翻转reverse方法（第一个和最后一个交换，第二个和倒数第二交换）
3. 从小到大排序sort方法（数字的排序）
4. 冒泡排序算法
5. 随机提问（点名）程序（课前分享的点名）

可选：

1. 选择排序算法（扩展）

扩展

□ 数据结构： 栈结构、队列结构

作业

必做：

1. 冒泡排序(13K)
2. 随机生成一个五位以内的数，然后输出该数共有多少位，每位分别是什么（10k）
3. 有一个从小到大排好序的数组。现输入一个数，要求按原来的规律将它插入数组中。（11K）
4. 开发一个标题为“FlipFlop”的游戏应用程序。它从1计数到100，遇到3的倍数就替换为单词“Flip”，5的倍数就替换为单词“Flop”，既为3的倍数又为5的倍数则替换为单词“FlipFlop”。(8K)
5. 彩票程序（31选5）。（5个数重复不重复无所谓）
6. 随机点名程序
7. 把以上题封装成函数

可选：

1. 给定一组数，求哪个元素更接近平均数（数组的平均数）。
2. 把今天的课堂案例和作业封装成函数

作业

可选：

1. 生成13位条形码，封装成函数(6K)

Ean-13码规则：第十三位数字是前十二位数字经过计算得到的校验码。

例如：690123456789

第十三位计算其校验码的过程为：

@前十二位的奇数位和 $6+0+2+4+6+8=26$

@前十二位的偶数位和 $9+1+3+5+7+9=34$

@将奇数和与偶数和的三倍相加 $26+34*3=128$

@取结果的个位数：128的个位数为8

@用10减去这个个位数 $10-8=2$

所以校验码为2（注：如果取结果的个位数为0，那么校验码不是（ $10-0=10$ ），而是0）实现方法ean13（）计算验证码，输入12位条码，返回带验证码的条码。

例如：输入：692223361219输出：6922233612192

