

# 《数据结构》课程实验指导书

(2012年4月修订)

课程编号: 1300627  
实用专业: 电子商务  
学时数: 72;  
学分: 2  
编写者: 李湘东

## 实验一: 基本的数据结构 (9 个学时)

### 【实验目的】

复习和巩固Java语言的有关知识,特别是Java语言所提供的数组等基本的数据结构。

### 【实验内容】

数组元素的倒置(显示倒置/交换的过程)。

计算数组全部元素的值之和。

复制数组的全部元素。

十进制数变换成2至36进制数(从数组的最初元素开始按照高位到低位顺序存储变换后进制的各位数字)。

十进制数变换成2至36进制数(显示变换的详细过程)。

计算某年内已经过去的天数(要求不使用变量i和days的dayOfYear方法;要求使用while语句)。

计算y年m月d日在该年内剩下的天数(如果是12月31日,剩下的日子为0;如果是12月30日,剩下的日子为1)。

由姓名、身高和视力数据构成类,从该类的数组中求出平均身高和视力的分布(用图表示)。

### 【实验结果的检验】

当场演示

## 实验二：查找（9 个学时）

### 【实验目的】

实现查找的有关算法。

### 【实验内容】

使用for语句实现哨兵法查找。

线性查找（显示详细过程）。

查找数组中与key值相等的全部元素的下标。

二分法查找（显示详细过程）。

二分法查找（查找最前面/最靠近起始元素方向的相等元素）。

实现链接哈希法的例子（以姓名作为键值）。

### 【实验结果的检验】

当场演示

## 实验三：栈和队列（9 个学时）

### 【实验目的】

实现栈和队列的数据结构。

### 【实验内容】

使用IntStack类的所有成员方法实现int型栈的利用程序例。

一个数组的两边各一个int型栈。

实现泛型栈Gstack<E>类并使用该泛型栈类的所有成员方法。

实现int型队列及其测试程序。

### 【实验结果的检验】

当场演示

## 实验四：递归算法（5 个学时）

### 【实验目的】

理解并实现递归算法的基本思想。

### 【实验内容】

编写不使用递归调用的factorial()方法，实现阶乘计算。

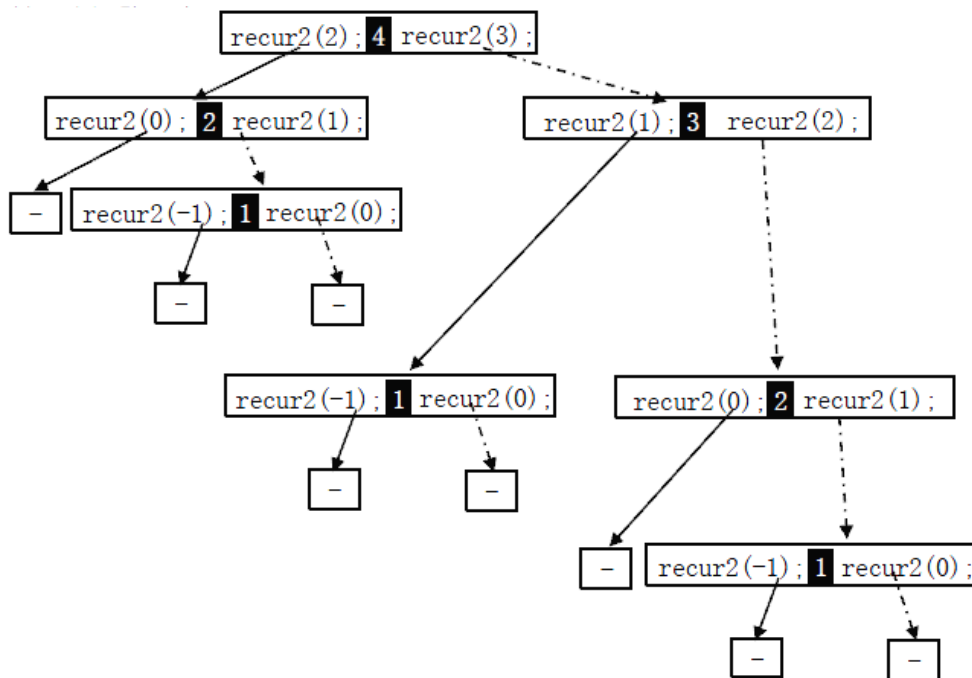
编写不使用递归调用的gcd()方法，实现欧几里德辗转相除法求最大公约数。

编写(使用递归调用的)gcdArray()方法，求解数组a的全部元素的最大公约数。

请使用自上而下以及自下而上等两种解析手法分析下记的recur2()递归方法：

```
static void recur2(int n) {  
    if (n > 0) {  
        recur2(n - 2);  
        System.out.println(n);  
        recur2(n - 1);  
    }  
}
```

自上而下递归过程：



### 【实验结果的检验】

当场演示，实验报告

## 实验五：排序（12个学时）

### 【实验目的】

实现排序的有关算法。

### 【实验内容】

实现冒泡排序（要求：按照单纯或直接交换排序，每一轮的比较及交换从待排序数据的起始元素开始，而不是从末尾元素开始；且每相邻两个数据的比较时将较大的值移动到右边的元素之中）。

冒泡排序（单纯或直接交换排序；显示详细的交换过程）。

显示选择排序的比较与交换的详细过程。

显示插入排序的比较与交换的详细过程。

希尔排序 version 1（添加对移动/赋值操作的计数）。

希尔排序 version 2（添加对移动/赋值操作的计数  $h = \dots, 40, 13, 4, 1$ ）。

快速排序（要求：以数组和数据个数等两个参数，定义快速排序的重载方法 `quickSort(int[] a, int n)`）。

在快速排序程序中增加显示分割（分组）过程的语句。

### 【实验结果的检验】

当场演示

## 实验六：集合（5 个学时）

### 【实验目的】

理解并实现集合的基本思想。

### 【实验内容】

`IntSet` 类中增加以下的方法：`boolean isEmpty()`、`boolean isFull()`、`void clear()`等。

将集合 `s` 的全部元素添加进集合（即，取集合与集合 `S` 的逻辑和）。

判断集合是否是集合 `s` 的子集。

复制集合 `s1` 和集合 `s2` 的交集。

实现验证上述各种方法的程序例。

### 【实验结果的检验】

当场演示

## 实验七：字符串查找（5 个学时）

### 【实验目的】

实现字符串查找的有关算法。

### 【实验内容】

强力法字符串查找（显示详细的匹配过程）。

强力法字符串查找（匹配文本字符串中最后面的模式字符串）。

KMP 法字符串查找（显示匹配的详细过程）。

**【实验结果的检验】**

当场演示

## **实验八：线性链表（9 个学时）**

**【实验目的】**

理解并实现线形链表数据结构。

**【实验内容】**

针对链表的第  $n$  个结点，实现返回指向该结点的引用或指针的方法：

`E retrieve(int n)`

例如， $n$  为 0 时，返回指向头结点的指针， $n$  为 1 时，返回指向第二个结点的指针等。

请参照循环链表 `CircLinkedList` 类 `List9-3` 中删除尾结点的 `public void removeLast()` 方法，完成删除头结点的 `public void removeFirst()` 方法。

针对循环双向链表的第  $n$  个结点，实现返回指向该结点的引用或指针的方法：

`E retrieve(int n)`

例如， $n$  为 0 时，返回指向头结点的指针， $n$  为 1 时，返回指向第二个结点的指针等。

**【实验结果的检验】**

当场演示

## **实验九：树（9 个学时）**

**【实验目的】**

理解并实现树数据结构。

**【实验内容】**

编写按照键值的降序显示二分查找树的全部结点的方法：

`void printReverse()`

编写下列方法，分别返回二分查找树中的最小键值、最小键值的结点中的数据、二分查找树中的最大键值、最大键值的结点中的数据。如果树为空 `null`，则返回 `null`：

K getMinkey() //二分查找树中的最小键值

V getDattaWithMinKey() //二分查找树中最小键值的结点中的数据

K getMaxkey() //二分查找树中的最大键值

V getDattaWithMaxKey() //二分查找树中最大键值的结点中的数据

编写使用二叉查找树 BinTree<K,V>类的程序例（要求：比较键值大小时使用比较函数 comparator 接口）。

**【实验结果的检验】**

当场演示.