



高等学校Java课程系列教材



Java 面向对象程序设计

(第3版) 实验指导与习题解答

◎ 耿祥义 张跃平 主编

清华大学出版社



Java 面向对象程序设计

(第3版) 实验指导与习题解答

高等学校Java课程系列教材

- Java 2实用教程 (第5版)
- Java 2实用教程 (第5版) 实验指导与习题解答
- Java课程设计 (第3版)
- Java程序设计精编教程 (第3版)
- Java面向对象程序设计 (第3版) 微课视频版
- Java面向对象程序设计 (第3版) 实验指导与习题解答
- 面向对象与设计模式
- JSP程序设计 (第3版)
- JSP程序设计 (第3版) 上机实验与综合实训

课件下载·样书申请



书圈

清华社官方微信号



扫我有惊喜

ISBN 978-7-302-52923-1



9 787302 529231 >

定价: 29.80元



高等学校Java课程系列教材



Java 面向对象程序设计

(第3版) 实验指导与习题解答

◎ 耿祥义 张跃平 主编

清华大学出版社
北京

内 容 简 介

本书是《Java 面向对象程序设计(第3版)-微课视频版》的配套实验指导和习题解答。本书的第一部分为16次上机实践的内容,每次上机实践由若干个实验组成。每个实验由相关知识点、实验目的、实验要求、程序模板、实验指导和实验报告等组成。在进行实验之前,首先通过实验目的了解实验要完成的关键主题,通过实验要求知道本实验应达到怎样的标准;然后,完成实验模板,填写实验报告。本书的第二部分为主教材的习题参考解答。

本书提供书中所有实验的实验模板的源程序,扫描封底的课件二维码可以下载。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Java 面向对象程序设计(第3版)实验指导与习题解答/耿祥义,张跃平主编.—3版.—北京:清华大学出版社,2020.1

高等学校 Java 课程系列教材

ISBN 978-7-302-52923-1

I. ①J… II. ①耿… ②张… III. ①JAVA 语言—程序设计—高等学校—教学参考资料 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2019)第 085768 号

策划编辑:魏江江

责任编辑:王冰飞

封面设计:刘 键

责任校对:胡伟民

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-83470236

印 装 者:北京密云胶印厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:11

字 数:280千字

版 次:2011年12月第1版 2020年1月第3版

印 次:2020年1月第1次印刷

印 数:1~1500

定 价:29.80元

产品编号:083959-01

前 言

本书是《Java 面向对象程序设计(第 3 版)-微课视频版》的配套实验指导和习题解答,目的是通过一系列实验练习使学生巩固所学的知识。要求 JDK 版本不低于 JDK 6,部分实验需要 JDK 8 之后的版本,如第 6 章的实验 3 要求 JDK 10 之后版本,第 9 章的实验 3、第 10 章的实验 2 和第 13 章的实验 3 要求 JDK 8 之后版本。

本书的第一部分为 16 次上机实践的内容,每次上机实践由若干个实验组成,每个实验由 8 个主要部分构成:

1. 相关知识点

这一部分给出和该实验相关的重点知识和难点知识。

2. 实验目的

这一部分让学生了解本实验需要掌握哪些知识,实验将以这些知识为中心。

3. 实验要求

这一部分给出该实验需要达到的基本标准。

4. 程序效果示例

运行效果图,就是程序运行的效果参考图。

5. 程序模板

程序模板是一个 Java 源程序,其中删除了需要学生重点掌握的代码,这部分代码要求学生来完成。模板起到引导作用,学生通过完成模板可以深入了解解决问题的方式。

6. 实验指导

这一部分针对实验的难点给出必要的提示。要求学生向指导老师演示模板程序的运行效果。

7. 实验后的练习

在实验的基础上,给出需要进一步完成的任务。

8. 填写实验报告

实验者根据自己的完成情况,填写实验报告单。

本书的第二部分为主教材的习题解答,仅供参考。

注:本书提供书中所有实验的实验模板的源程序,扫描封底的课件二维码可以下载。

作者

2019 年 9 月

目 录

第 1 章 Java 入门	1
实验 1 一个简单的应用程序	1
实验 2 教室、教师和学生	3
实验答案	5
第 2 章 基本数据类型	6
实验 1 输出特殊偏旁的汉字	6
实验 2 输入、输出学生的基本信息	7
实验 3 超大整数的加法	9
实验答案	10
第 3 章 运算符、表达式与语句	11
实验 1 托运行李	11
实验 2 自动售货机	12
实验 3 猜数字游戏	14
实验答案	16
第 4 章 类与对象	17
实验 1 Tank 类	17
实验 2 计算机与 CD	19
实验 3 家族的姓氏	22
实验答案	24
第 5 章 继承与接口	26
实验 1 中国人与美国人	26
实验 2 银行与利息	29
实验 3 面积之和	31
实验 4 歌手大赛	34
实验 5 天气预报	36
实验答案	38

第 6 章 内部类与异常类	40
实验 1 校内报纸	40
实验 2 检查危险品	41
实验 3 Lambda 语法糖	44
实验答案	45
第 7 章 面向对象的几个基本原则	46
实验 1 楼房的窗户	46
实验 2 搭建流水线	49
实验答案	52
第 8 章 几个重要的设计模式	53
实验 1 分组策略	53
实验 2 房屋中介	56
实验 3 编写文件的步骤	59
实验答案	61
第 9 章 常用实用类	62
实验 1 检索图书	62
实验 2 购物小票	64
实验 3 比较日期	65
实验 4 处理大整数	67
实验 5 替换 IP	68
实验答案	70
第 10 章 Java Swing 图形用户界面	72
实验 1 算术测试	72
实验 2 布局与日历	76
实验 3 华容道	79
实验答案	83
第 11 章 对话框	84
实验 1 字体对话框	84
实验 2 计算平方根	87
实验答案	89
第 12 章 输入输出流	90
实验 1 举重成绩单	90

实验 2 统计英文单词	92
实验 3 密码流	95
实验答案	97
第 13 章 泛型与集合框架	98
实验 1 按身高排序	98
实验 2 电话簿	100
实验 3 演出节目单	102
实验答案	105
第 14 章 JDBC 数据库操作	106
实验 1 抽取样本	106
实验 2 用户转账	108
实验答案	110
第 15 章 多线程	112
实验 1 汉字输入练习	112
实验 2 多线程猜数字	114
实验 3 月亮围绕地球	117
实验答案	120
第 16 章 Java 中的网络编程	121
实验 1 读取服务器端文件	121
实验 2 会结账的服务器	123
实验 3 读取服务器端的窗口	127
实验 4 与服务器玩猜数游戏	130
实验 5 传输图像	134
实验答案	138
附录 主教材习题参考解答	140

实验 1 一个简单的应用程序

1. 相关知识点

Java 语言的出现是源于对独立于平台语言的需要,即这种语言编写的程序不会因为芯片的变化而无法运行或出现运行错误。目前,随着网络的迅速发展,Java 语言的优势越发明显,Java 已经成为网络时代最重要的语言之一。

Sun 公司要实现“编写一次,到处运行”的目标,就必须提供相应的 Java 运行平台,目前 Java 运行平台主要分为下列 3 个版本。

(1) Java SE: 称为 Java 标准版或 Java 标准平台。Java SE 提供了标准的 JDK 开发平台。利用该平台可以开发 Java 桌面应用程序和低端的服务器应用程序,也可以开发 Java Applet 程序。当前的 JDK 版本为 JDK 12。

(2) Java EE: 称为 Java 企业版或 Java 企业平台。使用 Java EE 可以构建企业级的服务应用,Java EE 平台包含了 Java SE 平台,并增加了附加类库,以便支持目录管理、交易管理和企业级消息处理等功能。

(3) Java ME: 称为 Java 微型版或 Java 小型平台。Java ME 是一种很小的 Java 运行环境,用于嵌入式的消费产品中,如移动电话、掌上电脑或其他无线设备等。

上述 Java 运行平台都包括了相应的 Java 虚拟机(Java Virtual Machine),虚拟机负责将字节码文件(包括程序使用的类库中的字节码)加载到内存,然后采用解释方式来执行字节码文件,即根据相应硬件的机器指令翻译一句执行一句。Java SE 平台是学习 Java 语言的最佳平台,而掌握 Java SE 又是进一步学习 Java EE 和 Java ME 所必需的。

2. 实验目的

本实验的目的是让学生掌握开发 Java 应用程序的 3 个步骤:编写源文件、编译源文件和运行应用程序。

3. 实验要求

编写一个简单的 Java 应用程序,该程序在命令行窗口输出两行文字:“你好,很高兴学习 Java”和“We are students”。

4. 程序效果示例

程序运行效果如图 1.1 所示。

5. 程序模板

按模板要求,将【代码】部分替换为 Java 程序代码。

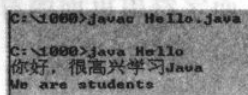


图 1.1 简单的应用程序

```

//Hello.java
public class Hello {
public static void main (String args[ ]) {
    【代码 1】 //命令行窗口输出"你好,很高兴学习 Java"
        A a = new A();
        a.fA();
    }
}
class A {
    void fA() {
        【代码 2】 //命令行窗口输出"We are students"
    }
}

```

6. 实验指导

打开一个文本编辑器。如果是 Windows 操作系统,打开“记事本”编辑器,可以通过“程序”|“附件”|“记事本”来打开文本编辑器;如果是其他操作系统,请在指导老师的帮助下打开一个纯文本编辑器。

按照“程序模板”的要求编辑输入源程序。

(1) 保存源文件。

将源文件命名为 Hello.java。可以将源文件保存到 C 盘的某个文件夹中,例如 C:\1000。

(2) 编译源文件。

打开命令行窗口,对于 Windows 操作系统,打开 MS-DOS 窗口。对于 Windows 2000/XP 操作系统,可以通过单击“开始”按钮,选择“程序”|“附件”|MS-DOS 打开命令行窗口,也可以选择“开始”|“运行”命令,在打开的“运行”对话框中输入“cmd”,打开命令行窗口。如果当前 MS-DOS 窗口显示的逻辑符是“D:\”,输入“C:”,按 Enter 键确认,使得当前 MS-DOS 窗口的状态是“C:\”。如果当前 MS-DOS 窗口的状态是 C 盘符的某个子目录,请输入“cd\”,使得当前 MS-DOS 窗口的状态是“C:\”。当 MS-DOS 窗口的状态是“C:\”时,输入进入文件夹目录的命令,例如,“CD 1000”,然后执行下列编译命令: C:\1000 > javac Hello.java。

在编译源文件时如果遇到错误提示“Command not Found”,请检查是否正确设置了系统变量 Path。如果 JDK 的安装目录是 C:\jdk12,可以在命令行临时设置系统变量 Path: path C:\jdk12\bin。

在编译源文件时如果遇到错误提示“File not Found”,请检查源文件是否保存在当前目录中。

在编译源文件时可能遇到一些语法错误的提示,例如“非法字符: \65307”,其原因是在汉语输入状态下输入了程序中需要的语句分号。Java 源程序中语句所涉及的小括号及标点符号都是英文状态下输入的,如“你好,很高兴学习 Java”中的引号必须是英文状态下的引号,而字符串里面的符号不受汉语或英文的限制。

(3) 运行程序。

```
C:\1000 > java Hello
```

运行程序如果出现错误提示“Exception in thread main java.lang.NoClassFondError”,

请检查是否正确设置了系统变量 ClassPath,或检查是否运行的是主类的名字。

7. 实验后的练习

- (1) 编译器怎样提示丢失大括号的错误。
- (2) 编译器怎样提示语句丢失分号的错误。
- (3) 编译器怎样提示将 System 写成 system 这一错误。
- (4) 编译器怎样提示将 String 写成 string 这一错误。

8. 填写实验报告

实验编号: 101

学生姓名:

实验时间:

教师签字:

实验效果评价	A	B	C	D	E
模板完成情况					
实验后练习效果评价	A	B	C	D	E
练习完成情况					
总评					

实验 2 教室、老师和学生

1. 相关知识点

一个 Java 应用程序(也称为一个工程)是由若干个类所构成,这些类可以在一个源文件中,也可以分布在若干个源文件中。Java 应用程序有一个主类,即含有 main 方法的类,Java 应用程序从主类的 main 方法开始执行。在编写一个 Java 应用程序时,可以编写若干个 Java 源文件,每个源文件编译后产生若干个类的字节码文件。经常需要进行如下的操作。

- 将应用程序涉及的 Java 源文件保存在相同的目录中,分别编译通过,得到 Java 应用程序所需要的字节码文件。

- 运行主类。

当使用解释器运行一个 Java 应用程序时,Java 虚拟机将 Java 应用程序需要的字节码文件加载到内存,然后再由 Java 虚拟机解释执行。因此,可以事先单独编译一个 Java 应用程序所需要的其他源文件,并将得到的字节码文件和主类的字节码文件存放在同一目录中。如果应用程序的主类的源文件和其他的源文件在同一目录中,也可以只编译主类的源文件,Java 系统会自动地先编译主类需要的其他源文件。

2. 实验目的

熟悉 Java 应用程序的基本结构,并能联合编译应用程序所需要的类。

3. 实验要求

编写 3 个源文件: Classroom.java、Teacher.java 和 Student.java,每个源文件只有一个类。Classroom.java 含有应用程序的主类(含有 main 方法),并使用了 Teacher 和 Student 类。将 3 个源文件保存到同一目录中,例如 C:\1000,然后编译 Classroom.java。

4. 程序效果示例

程序运行效果如图 1.2 所示。

5. 程序模板

请按模板要求,将【代码】部分替换为 Java 程序代码。

教学活动从教室开始
我是张老师
我是学生,名字是:奖励

图 1.2 只编译主类

```
//ClassRoom.java
public class ClassRoom {
    public static void main (String args[ ]) {
        【代码 1】 //命令行窗口输出"教学活动从教室开始"
        Teacher zhang = new Teacher();
        Student jiang = new Student();
        zhang.introduceSelf();
        jiang.introduceSelf();
    }
}

//Teacher.java
public class Teacher {
    void introduceSelf() {
        【代码 2】 //命令行窗口输出"我是张老师"
    }
}

//Student.java
public class Student {
    void introduceSelf() {
        【代码 3】 //命令行窗口输出"我是学生,名字是:奖励"
    }
}
```

6. 实验指导

编译 ClassRoom.java 的过程中,Java 系统会自动地先编译 Teacher.java、Student.java,编译通过后,C:\1000 中将会有 ClassRoom.class、Teacher.class 和 Student.class 这 3 个字节码文件。

当运行上述 Java 应用程序时,虚拟机将 ClassRoom.class、Teacher.class 和 Student.class 加载到内存。当虚拟机将 ClassRoom.class 加载到内存时,就为主类中的 main 方法分配了入口地址,以便 Java 解释器调用 main 方法开始运行程序。如果编写程序时错误地将主类中的 main 方法写成 public void main(String args[]),那么,程序可以编译通过,但却无法运行。

7. 实验后的练习

(1) 将 ClassRoom.java 编译通过以后,不断地修改 Teacher.java 源文件中的【代码 2】,例如,在命令行窗口输出“我是数学老师”或“我是物理老师”。要求每次修改 Teacher.java 源文件后,单独编译 Teacher.java,然后直接运行应用程序(不要再编译 ClassRoom.java)。

(2) 如果需要编译某个目录(例如 C:\1000 目录)下的全部 Java 源文件,可以使用如下命令:

```
C:\1000> javac *.java
```

请练习上述命令。

8. 填写实验报告

实验编号: 102 学生姓名: 实验时间: 教师签字:

实验效果评价	A	B	C	D	E
模板完成情况					
实验后练习效果评价	A	B	C	D	E
练习完成情况					
总评					

实验答案

实验 1

【代码 1】 `System.out.println("你好,很高兴学习 Java");`

【代码 2】 `System.out.println("We are students");`

实验 2

【代码 1】 `System.out.println("教学活动从教室开始");`

【代码 2】 `System.out.println("我是张老师");`

【代码 3】 `System.out.println("我是学生,名字是:奖励");`

实验 2 输入、输出学生的基本信息

1. 相关知识点

`Scanner` 是 JDK 1.5 新增的一个类,可以使用该类创建 `Scanner` 对象。

创建 `Scanner` 对象并调用 `nextLine()` 方法,可以读取用户输入的一行文本,并返回该文本。

调用 `nextLine()` 方法时,程序等待用户在控制台输入一行文本,并按下回车键。

调用 `nextLine()` 方法时,程序等待用户在控制台输入一行文本,并按下回车键。

调用 `nextLine()` 方法时,程序等待用户在控制台输入一行文本,并按下回车键。

实验1 输出特殊偏旁的汉字

1. 相关知识点

Java 的简单数据类型(也称基本数据类型)包括 byte、short、int、long、float、double 和 char。简单数据类型按精度级别由低到高的顺序是:

```
byte short char int long float double
```

简单类型的数据转换规则是:当把级别低的变量的值赋给级别高的变量时,系统自动完成数据类型的转换;当把级别高的变量的值赋给级别低的变量时,需用类型转换运算。

要观察一个字符在 Unicode 表中的顺序位置,需使用 int 类型转换,如(int)'a'。要得到一个 0~65 535 的数所代表的 Unicode 表中相应位置上的字符,需使用 char 型转换。char 型数据和 byte、short、int 运算的结果是 int 型数据。

2. 实验目的

掌握 char 型数据和 int 型数据之间的互相转换,同时了解 Unicode 字符表。

3. 实验要求

编写应用程序,在命令行窗口输出 5 个“石”字旁的汉字和 5 个“女”字旁的汉字。

4. 程序效果示例

程序运行效果如图 2.1 所示。

```
输出5个石字旁的汉字
研(30740)研(30741)研(30742)研(30743)研(30744)研(30745)
输出5个女字旁的汉字
妈(22920)姨(22921)姪(22922)姨(22923)姪(22924)研(22925)
```

图 2.1 输出特殊偏旁的汉字

5. 程序模板

按模板要求,将【代码】部分替换为 Java 程序代码。

```
//InputChinese.java
public class InputChinese {
    public static void main (String args[ ]){
        char ch = '研',zifu = 0;
        int p = 22920,count = 5,position = 0;
        System.out.printf("输出 %d 个石字旁的汉字:\n",count);
        for(char c = ch;c <= ch + count;c++) {
            【代码 1】 //c 进行 int 型转换数据运算,并将结果赋值给 position
            System.out.printf(" %c( %d)",c,position);
        }
        System.out.printf("\n输出 %d 个女字旁的汉字:\n",count);
        for(int n = p;n <= p + count;n++) {
```

```

【代码 2】          // n 做 char 型转换运算, 并将结果赋值给 zifu
System.out.printf(" %c( %d)", zifu, n);
}
}
}

```

6. 实验指导

Unicode 表将偏旁相同的汉字按顺序排列。

7. 实验后的练习

(1) 将一个 double 型数据直接赋值给 float 型变量, 程序编译时系统会提示怎样的错误?

(2) 在应用程序的 main 方法中增加语句:

```
float x = 0.618;
```

程序能编译通过吗?

(3) 在应用程序的 main 方法中增加语句:

```
byte y = 128;
```

程序能编译通过吗? 在应用程序的 main 方法中增加语句:

```
int z = (byte)128;
```

程序输出变量 z 的值是多少?

8. 填写实验报告

实验编号: 201

学生姓名:

实验时间:

教师签字:

实验效果评价	A	B	C	D	E
模板完成情况					
实验后练习效果评价	A	B	C	D	E
练习完成情况					
总评					

实验 2 输入、输出学生的基本信息

1. 相关知识

Scanner 是 JDK 1.5 新增的一个类, 可以使用该类创建对象:

```
Scanner reader = new Scanner(System.in);
```

然后 reader 对象调用下列方法, 读取用户在命令行(如 MS-DOS 窗口)输入的各种简单类型数据:

```
nextBoolean(), nextByte(), nextShort(), nextInt(), nextLong(), nextFloat(), nextDouble()
```

上述方法执行时都会堵塞, 程序等待用户在命令行输入数据后按 Enter 键确认。

2. 实验目的

掌握从键盘为简单型变量输入数据。

3. 实验要求

编写 Java 应用程序,使用 Scanner 对象输入学生的基本信息,并输出基本信息。

4. 程序效果示例

程序运行效果如图 2.2 所示。

5. 程序模板

按模板要求,将【代码】部分替换为 Java 程序代码。

```
//InputMess.java
import java.util.Scanner;
public class InputMess {
    public static void main(String args[]) {
        Scanner reader = new Scanner(System.in);
        System.out.println("输入姓名(按 Enter 键确认):");
        String name = 【代码 1】 //从键盘为 name 赋值
        System.out.println("输入年龄(按 Enter 键确认):");
        byte age = 【代码 2】 //从键盘为 age 赋值
        System.out.println("输入身高(按 Enter 键确认):");
        float height = 【代码 3】 //从键盘为 height 赋值
        System.out.printf("% 28s\n", "-- 基本信息 --");
        System.out.printf("% 10s % - 10s", "姓名:", name);
        System.out.printf("% 4s % - 4d", "年龄:", age);
        System.out.printf("% 4s % - 4.2f", "身高:", height);
    }
}
```

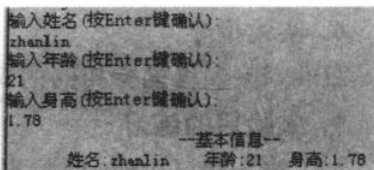


图 2.2 输入、输出学生基本信息

6. 实验指导

JDK 1.5 后续版本新增了和 C 语言中 printf 函数类似的数据输出方法,其格式为:

```
System.out.printf("格式控制部分",表达式 1,表达式 2, ...,表达式 n)
```

输出数据时也可以控制数据在命令行的位置,例如: %md——输出的 int 型数据占 m 列, %m.nf——输出的浮点型数据占 m 列,小数点保留 n 位。

7. 实验后的练习

编写一个 Java 应用程序,在主类的 main 方法中声明用于存放矩形的宽和高的两个 double 型变量 width、height 以及存放矩形面积的 double 型变量 area。

使用 Scanner 对象调用 nextDouble() 方法,让用户从键盘为 width、height 变量输入值,然后程序计算出矩形的面积,并输出矩形的宽和高以及面积。

8. 填写实验报告

实验编号: 202

学生姓名:

实验时间:

教师签字:

实验效果评价	A	B	C	D	E
模板完成情况					
实验后练习效果评价	A	B	C	D	E
练习完成情况					
总评					

实验3 超大整数的加法

1. 相关知识

对于特别大的整数,无法使用 long 型变量来处理大整数的加法。一种简单的处理办法是使用数组。可以将一个大整数的各个位上的数字存放到一个数组中,然后只需将存放大整数各个位上的数字的两个数组的各个元素按照一定的算法进行加法运算,将结果存放到另一个数组中即可。

2. 实验目的

本实验的目的是让学生掌握使用数组处理大整数的加法。

3. 实验要求

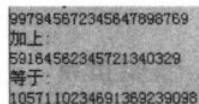
声明 3 个 int 型数组: a、b、c,要求三者的长度相同。将其中的 a、b 初始化为大整数的形式表示,但大整数的数字的长度必须小于数组的长度,以便保证数组 a 和 b 的首元素的值是 0。对 a 和 b 的各个元素进行加法运算(需要进位时,需改变 a 元素的值),结果依次存放到数组 c 中,输出数组 c。

4. 程序效果示例

程序运行效果如图 2.3 所示。

5. 程序模板

仔细阅读模板代码,完成实验后的练习。



```
997945672345647696769
加上
59184562345721340329
等于
1057110234691369239098
```

图 2.3 大整数的加法

```
//HandleLargeNumber.java
```

```
public class HandleLargeNumber {
    public static void main(String args[]) {
        int a[] = {0,9,9,7,9,4,5,6,7,2,3,4,5,6,4,7,8,9,8,7,6,9};
        int b[] = {0,0,5,9,1,6,4,5,6,2,3,4,5,7,2,1,3,4,0,3,2,9};
        int c[] = new int[a.length];
        int i = 0,result = 0,k = 0;
        for(i = 0;i < a.length;i++) {
            if(a[i] != 0) {
                k = i;
                break;
            }
        }
        for(i = k;i < a.length;i++) {
            System.out.printf("%d",a[i]);
        }
        System.out.printf("\n加上:\n");
        for(i = 0;i < b.length;i++) {
            if(b[i] != 0) {
                k = i;
                break;
            }
        }
        for(i = k;i < b.length;i++) {
            System.out.printf("%d",b[i]);
        }
    }
}
```