

普通高等院校计算机课程设计指导丛书

Java

课程设计

耿祥义 编著



清华大学出版社

普通高等院校计算机课程设计指导丛书

Java 课程设计

耿祥义 编著

清华大学出版社
北京

内 容 简 介

许多读者在掌握 Java 的基本知识后,都希望通过系列的课程设计来巩固和提高 Java 编程技术,本书即是针对这一目的编写的。本书不仅可以作为理工科各专业 Java 课程设计的教材,也适合作为撰写相关课程论文的参考书。

本书以 15 个课程设计题目为框架,从各个方面介绍了 Java 在应用系统开发和网络开发中的实用技巧。各个课程设计题目相互独立,可以从任何一个课程设计题目开始阅读本书。每个课程设计题目都给出了详细的设计步骤,包括设计内容、总体设计、具体设计、软件发布、课程设计作业等。本书全部程序代码可由前言中指定网站下载。

版权所有,翻印必究

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

Java 课程设计/耿祥义编著. —北京:清华大学出版社, 2004.1

(普通高等院校计算机课程设计指导)

ISBN 7-302-07915-3

I. J… II. 耿… III. JAVA 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2004)第 000294 号

出 版 者: 清华大学出版社

<http://www.tup.com.cn>

社 总 机: 010-62770175

地 址: 北京清华大学学研大厦

邮 编: 100084

客户服务: 010-62776969

责任编辑: 徐培忠

封面设计: 付弘宇

封面设计: 付剑飞

印 刷 者: 北京密云胶印厂

装 订 者: 三河市化甲屯小学装订二厂

发 行 者: 新华书店总店北京发行所

开 本: 185×260 印张: 17.75 字数: 440 千字

版 次: 2004 年 1 月第 1 版 2004 年 1 月第 1 次印刷

书 号: ISBN 7-302-07915-3/TP·5748

印 数: 1~3000

定 价: 26.00 元

前 言

Java 不依赖平台的特点使它受到广泛的关注, Java 已成为网络时代最重要的语言之一。目前, Java 语言不仅是一门被广泛使用的编程语言, 而且已成为软件设计开发者应当掌握的一门基础语言。因为很多新的技术领域都涉及到 Java 语言, 国内外许多大学已将 Java 语言列入了本科教学计划, 而 IT 行业对 Java 人才的需求也在不断的增长, 因此, 掌握 Java 已经成为共识。

许多读者在掌握了 Java 基本知识后, 都希望通过系列的课程设计来提高 Java 编程技术, 从而能在将来的工作中利用 Java 语言开发具体项目。本书就是针对这一目的编写的。

本书给出了 15 个具有一定代表性的课程设计题目, 各个课程设计题目相互独立, 可以从任何一个课程设计题目开始阅读本书。每个课程设计题目都给出了详细的设计步骤, 包括设计内容、总体设计、具体设计、软件发布、课程设计作业等; 代码都附有详细的注释, 以便读者阅读。读者可以按着作者布置的课程设计作业来开发一个软件, 也可以参考这些课程设计题目, 设计类似的软件。每个课程设计题目既有理论分析, 又有具体步骤, 以便于读者掌握开发 Java 程序的基本方法和原理, 为今后从事项目开发或撰写毕业论文积累经验。

虽然本书是《Java 2 实用教程》的配套教材, 但也可独立使用, 本书仅有个别题目所用知识超出《Java 2 实用教程》, 但都给出了详细的说明, 无须额外查阅资料, 就能顺利阅读。

本书的全部代码由作者亲自编写, 在 JDK1.3 运行环境下调试通过。本书代码仅供学习 Java 使用, 欢迎读者对不当之处提出批评建议。

本书的全部代码可由 <http://www.tupwq.net> 网站的“资源下载”栏下载。

编 者

2003 年 12 月

目 录

第 1 章 记忆测试系统	1
1.1 设计内容	1
1.2 设计要求	1
1.3 总体设计	1
1.4 具体设计	2
1.4.1 运行效果与程序发布	2
1.4.2 主类 Memory	4
1.4.3 方块 Block	9
1.4.4 记忆测试板 MemoryTestPane	10
1.4.5 显示成绩 ShowRecord	13
1.4.6 记录成绩 Record	16
1.4.7 随机排列图标 RandomSetIcon	18
1.4.8 测试者 People	19
1.5 课程设计作业	20
第 2 章 计算器	21
2.1 设计内容	21
2.2 设计要求	21
2.3 总体设计	21
2.4 具体设计	21
2.4.1 运行效果与程序发布	21
2.4.2 主类 ComputerPad	23
2.4.3 数字按钮 NumberButton	31
2.4.4 运算符号按钮 OperationButton	32
2.5 课程设计作业	33
第 3 章 Hanoi 塔	34
3.1 设计内容	34
3.2 设计要求	34
3.3 总体设计	34
3.4 具体设计	35
3.4.1 运行效果与程序发布	35
3.4.2 主类 Tower	36
3.4.3 Hanoi 塔 HanoiTower	39
3.4.4 塔点 TowerPoint	49

3.4.5 盘子 Disk	50
3.5 课程设计作业	51
第 4 章 JPEG 图像生成器	52
4.1 设计内容	52
4.2 设计要求	52
4.3 总体设计	52
4.4 具体设计	52
4.4.1 运行效果与程序发布	52
4.4.2 主类 MakeJPEG	54
4.5 课程设计作业	59
第 5 章 标准化考试系统 (单机版)	60
5.1 设计内容	60
5.2 设计要求	60
5.3 总体设计	60
5.4 具体设计	61
5.4.1 运行效果与程序发布	61
5.4.2 主类 EnglishTest	62
5.4.3 考试区域 TestArea	63
5.4.4 读取试题 ReadTestquestion	69
5.5 课程设计作业	72
第 6 章 标准化考试系统 (C/S 网络版)	73
6.1 设计内容	73
6.2 设计要求	73
6.3 总体设计	73
6.4 具体设计	75
6.4.1 运行效果与程序发布	75
6.4.2 客户端主类 Client	77
6.4.3 客户端选择试题界面 ChoiceFile	80
6.4.4 客户端考试界面 ClientTestArea	83
6.4.5 服务器端主类 Server	88
6.4.6 服务器端读取试题 ReadTestquestion	91
6.5 课程设计作业	94
第 7 章 标准化考试系统 (B/S 网络版)	95
7.1 设计内容	95
7.2 设计要求	95
7.3 总体设计	95

7.4	具体设计	97
7.4.1	运行效果与程序发布	97
7.4.2	客户端主类 ClientBS	97
7.4.3	客户端选择试题界面 ChoiceFile	100
7.4.4	客户端考试界面 ClientTestArea	103
7.4.5	服务器端主类 Server	108
7.4.6	服务器端读取试题 ReadTestquestion	112
7.5	课程设计作业	115
第 8 章	日历记事本	116
8.1	设计内容	116
8.2	设计要求	116
8.3	总体设计	116
8.4	具体设计	117
8.4.1	运行效果与程序发布	117
8.4.2	主类 CalendarPad	118
8.4.3	记事本 NotePad	123
8.4.4	年 Year	128
8.4.5	月 Month	130
8.5	课程设计作业 1	132
8.6	课程设计作业 2	132
第 9 章	学籍管理系统	133
9.1	设计内容	133
9.2	设计要求	133
9.3	总体设计	133
9.4	具体设计	134
9.4.1	运行效果与程序发布	134
9.4.2	主类 StudentManager	136
9.4.3	录入界面 StudentSituation	138
9.4.4	查询界面 Inquest	142
9.4.5	修改界面 ModifySituation	146
9.4.6	删除界面 Delete	151
9.4.7	学生对象 Student	155
9.5	课程设计作业	157
第 10 章	图书查询系统 (B/S 网络版)	158
10.1	设计内容	158
10.2	设计要求	158
10.3	总体设计	158

10.4	具体设计	158
10.4.1	运行效果与程序发布	158
10.4.2	客户端主类 DatabaseClient	159
10.4.3	服务器端主类 DatabaseServer	163
10.5	课程设计作业	166
第 11 章	中国象棋打谱系统	167
11.1	设计内容	167
11.2	设计要求	167
11.3	总体设计	167
11.4	具体设计	169
11.4.1	运行效果与程序发布	169
11.4.2	主类 Chess	170
11.4.3	对弈棋盘 ChessBoard	174
11.4.4	棋子 ChessPiece	182
11.4.5	棋点 ChessPoint	184
11.4.6	走棋规则 Rule	186
11.4.7	步骤 MoveStep	193
11.4.8	记录棋谱 MakeChessManual	194
11.4.9	棋谱演示 Demon	197
11.5	课程设计作业 1	202
11.6	课程设计作业 2	202
第 12 章	魔板游戏	203
12.1	设计内容	203
12.2	设计要求	203
12.3	总体设计	203
12.4	具体设计	204
12.4.1	运行效果与程序发布	204
12.4.2	主类 PuzzleFrame	205
12.4.3	魔板 PuzzlePad	208
12.4.4	魔板中的点 SquarePoint	213
12.5	课程设计作业	214
第 13 章	挖雷游戏	215
13.1	设计内容	215
13.2	设计要求	215
13.3	总体设计	215
13.4	具体设计	217
13.4.1	运行效果与程序发布	217

13.4.2	主类 Game	219
13.4.3	方块 Block	222
13.4.4	雷区 MineSquare	226
13.4.5	雷标数目 FindAroundMineMarkNumber	231
13.4.6	雷标判断 DetermineMineMarkIsRightOrWrong	232
13.4.7	成片挖开区域 DetermineDigArea	233
13.4.8	无雷连通区 FindSafeArea	235
13.4.9	随机布雷 RandomSetMine	236
13.4.10	周围地雷个数 FindAroundMineNumber	237
13.4.11	显示剩余雷数 CountMine	238
13.4.12	计时器 TimeCount	239
13.4.10	英雄榜录入对话框 Record	240
13.4.14	显示英雄榜对话框 ShowRecord	243
13.4.15	挖雷成功 DecideWinner	246
13.5	课程设计作业	247
第 14 章	网络聊天室 (B/S 模式)	248
14.1	设计内容	248
14.2	设计要求	248
14.3	总体设计	248
14.4	具体设计	249
14.4.1	运行效果与程序发布	249
14.4.2	客户端主类 ClientChat	250
14.4.3	客户端输入昵称界面 InputNameTextField	253
14.4.4	客户端聊天界面 ChatArea	257
14.4.5	服务器端主类 ChatServer	261
14.5	课程设计作业	265
第 15 章	局域网广播系统	266
15.1	设计内容	266
15.2	设计要求	266
15.3	总体设计	266
15.4	具体设计	267
15.4.1	运行效果与程序发布	267
15.4.2	客户端主类 Receive	268
15.4.3	服务器端主类 BroadCastWord	271
15.5	课程设计作业	274

第 1 章 记忆测试系统

1.1 设计内容

记忆测试系统通过回忆法测试记忆，分为初级、中级、高级三个级别，并通过记忆榜存储每个级别的成绩。

1.2 设计要求

1. 单击“记忆测试”菜单可以选择初级/中级/高级，也可以查看初级记忆榜、中级记忆榜或高级记忆榜。

2. 选择级别后，将显示相应级别的测试区域。测试区域由若干个方块组成，每个方块上都有一个图标，但该图标默认情况下不可见。用鼠标单击测试区中的任何一个方块，则计时器开始启动，同时该方块上的图标将显示出来。对于初级级别，用户必须用鼠标连续单击出 6 个图标相同的方块，也就是说，如果用户已经用鼠标连续单击出 n ($n < 6$) 个图标相同的方块，单击另一个方块时，如果该方块的图标和上一个相同，那么该方块的图标和前 n 次单击出的方块的图标仍然可见，否则，前 n 次单击出的方块的图标都变为不可见。

3. 连续单击出该级别所要求的若干个图标相同的方块后，将显示保存成绩对话框，成绩按所用时间排序。用户可以通过该对话框选择是否将自己的成绩保存到成绩表中。

1.3 总体设计

记忆测试系统共包含 7 个 java 源文件和 7 个图像文件。

1. java 文件

类之间的主要关系如图 1.1 所示。

Memory.java

该 java 文件定义的类负责创建记忆测试系统的“主窗口”。该文件含有 main 方法，程序从该类开始执行。用户可以通过主窗口菜单选择级别，然后窗口内显示相应的 MemoryTestPane 对象。主窗口默认的 MemoryTestPane 对象是初级。

Block.java

该文件生成的类负责创建测试区中的 Block 对象。使用鼠标左键单击 Block 对象，可以显示该对象上的图标。

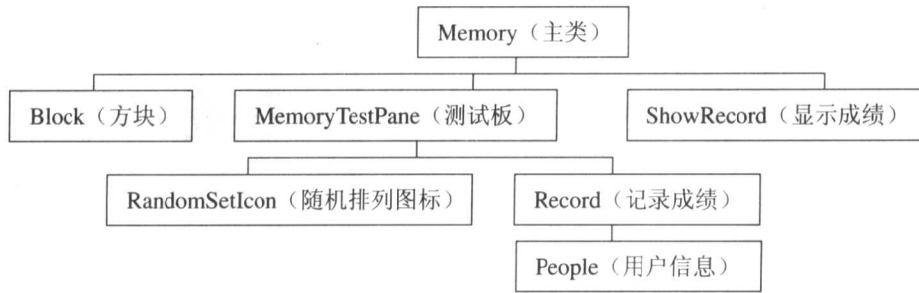


图 1.1 类之间的关系

MemoryTestPane.java

该文件生成的类负责为主类 Memory 创建 MemoryTestPane 对象，主类将级别、Block 对象、图标对象等传递给 MemoryTestPane。

RandomSetIcon.java

该文件生成的类负责为 MemoryTestPane 创建“排列图标”对象，“排列图标”可以随机设置 MemoryTestPane 对象中“方块”上的图标。

ShowRecord.java

该文件生成的类负责为“主窗口”创建“显示成绩”对象，用户通过该对象可以查看各个级别的成绩记录。

Record.java

该文件生成的类负责记录各个级别的成绩。

People.java

该文件生成的类负责存放用户的信息。

2. 图像文件

- a0.gif ：方块上的图标。
- a1.gif ：方块上的图标。
- a2.gif ：方块上的图标。
- a3.gif ：方块上的图标。
- a4.gif ：方块上的图标。
- a5.gif ：方块上的图标。
- a6.gif ：方块上的图标。
- 关闭.gif ：方块关闭时显示的图标。

1.4 具体设计**1.4.1 运行效果与程序发布**

1. 将 7 个 java 文件保存到一个文件夹中，例如，C:\jyiceshi。将 7 个 Java 文件分别编

译生成相应的字节码 (.class) 文件。然后, 用 java 解释器运行主类:

```
C:\jijiceshi\java Memory
```

效果如图 1.2、1.3 所示。

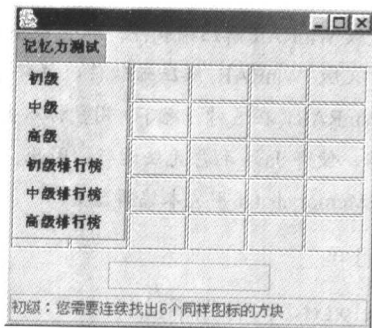


图 1.2 选择级别, 然后进行测试



图 1.3 连续找到 6 个相同的图标

2. 可以使用 jar.exe 把 java 应用程序中涉及到的类压缩成一个 JAR 文件, 来发布我们的应用程序, 例如 Memory.jar。

首先, 用文本编辑器 (例如 Windows 下的记事本) 编写一个清单文件 (Manifestfiles):

```
Manifest-Version: 1.0
Main-Class: Memory
Created-By: 1.2.2 (Sun Microsystems Inc.)
```

本例中, 将清单文件命名为 moon.mf, 保存到 C:\jijiceshi。

注意: 编写清单文件时, 在 “Manifest-Version:” 和 “1.0” 之间、“Main-Class:” 和主类 “Memory” 之间, 以及 “Created-By:” 和 “1.2.2” 之间必须有且只有一个空格。

然后, 如下生成 JAR 文件:

```
c:\jijiceshi\jar cfm Memory.jar moon.mf *.class
```

其中, 参数 c 表示要生成一个新的 JAR 文件; f 表示要生成的 JAR 文件的名称; m 表

示文件清单文件的名字。

现在可以将 Memory.jar 以及 7 个图标文件拷贝到任何一个安装了 Java 运行环境（版本号需等于或高于 1.2.2）的计算机上，只需用鼠标双击该文件就可以运行该 Java 应用程序。

注意：如果机器上没有安装过中文版 WinRAR 解压缩软件，那么 Memory.jar 的文件类型是 Executable Jar File。如果机器上安装过中文版 WinRAR 解压缩软件，并将 jar 文件与该软件做了关联，那么 Memory.jar 的文件类型是 WinRAR，在这种情况下，用鼠标双击该文件时，WinRAR 解压缩软件会自动运行并进行解压缩操作，使得 Java 程序无法运行。因此，发布本程序时，还应该创建一个包含如下内容的 bat 文件——Memory.bat（用文本编辑器）。

```
javaw -jar Memory.jar
```

另外创建一个帮助文件 help.txt，内容如下：

您可以用鼠标双击 Memory.jar 或 Memory.bat 来运行本软件，但是，如果您的计算机上安装了中文版 WinRAR 解压缩软件，并将 jar 文件与该软件做了关联，那么，当用鼠标双击 Memory.jar 文件时，WinRAR 解压缩软件会自动运行进行解压缩操作，这时您可以双击 Memory.bat 文件来运行我们的软件。

最后将 .bat 文件、.jar 文件、帮助文件以及 7 个图标文件一同发布。

1.4.2 主类 Memory

1. 成员变量（见表 1.1）

表 1.1 主要成员变量（属性）

成员变量描述	变量类型	名称
菜单条	JMenuBar	Bar
菜单	JMenuItem	FileMenu
菜单项	JMenuBar	初级，中级，高级，初级排行榜，中级排行榜，高级排行榜
面板容器	MemoryTestPane	记忆测试板
方块数组	Block[]	block
显示成绩	ShowRecord	显示成绩表对话框
图标数组	ImageIcon[]	icon

2. 方法（见表 1.2）

表 1.2 主要方法

名称	功能	备注
Memory	创建窗口	构造方法
给出级别测试	选择级别	
actionPerformed	处理 ActionEvent 事件	接口方法
Main	主程序开始运行	

3. 代码

Memory.java

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.io.*;
import java.util.LinkedList;
public class Memory extends JFrame implements ActionListener
{
    JMenuBar bar;
    JMenu fileMenu;
    JMenuItem 初级, 中级, 高级, 初级排行榜, 中级排行榜, 高级排行榜;
    Block block[];
    ImageIcon icon[];
    MemoryTestPane 记忆测试板=null;
    File file 初级=new File(" 初级记忆排行榜.txt" ),
        file 中级=new File(" 中级记忆排行榜.txt" ),
        file 高级=new File(" 高级记忆排行榜.txt" );

    LinkedList 成绩表=null;
    ShowRecord 显示成绩表对话框=null;
    int m=5, n=6; //测试区域的行数和列数。
    int 图标个数=0;
    Container con=null;
    JTextField 提示条=null;
    File gradeFile=null;
    public Memory()
    {
        block=new Block[m*n];
        图标个数=m;
        icon=new ImageIcon[图标个数];
        for(int i=0; i<icon.length; i++)
        {
            icon[i]=new ImageIcon(" a" +i+ ".gif" );
        }
        for(int i=0; i<block.length; i++)
        {
            block[i]=new Block();
            block[i].设置关闭时的图标(new ImageIcon(" 关闭.gif" ));
        }

        bar=new JMenuBar();
        fileMenu=new JMenu(" 记忆力测试" );
        初级=new JMenuItem(" 初级" );
        中级=new JMenuItem(" 中级" );
        高级=new JMenuItem(" 高级" );
        fileMenu.setMnemonic('G');
        初级.setMnemonic('B');
```

```
    中级.setMnemonic('I');
    高级.setMnemonic('E');
    高级排行榜=new JMenuItem("高级排行榜");
    中级排行榜=new JMenuItem("中级排行榜");
    初级排行榜=new JMenuItem("初级排行榜");
    初级.setMnemonic('T');
    fileMenu.add(初级);
    fileMenu.add(中级);
    fileMenu.add(高级);
    fileMenu.add(初级排行榜);
    fileMenu.add(中级排行榜);
    fileMenu.add(高级排行榜);
    bar.add(fileMenu);
    setJMenuBar(bar);
    初级.addActionListener(this);
    中级.addActionListener(this);
    高级.addActionListener(this);
    初级排行榜.addActionListener(this);
    中级排行榜.addActionListener(this);
    高级排行榜.addActionListener(this);
    成绩表=new LinkedList();
    if(!file 初级.exists())
    {
        try{
            FileOutputStream out=new FileOutputStream("初级记忆排行榜.txt");
            ObjectOutputStream object_out=new ObjectOutputStream(out);
            object_out.writeObject(成绩表);
            object_out.close();
            out.close();
        }
        catch(IOException e)
        {
        }
    }
    if(!file 中级.exists())
    {
        try{
            FileOutputStream out=new FileOutputStream("中级记忆排行榜.txt");
            ObjectOutputStream object_out=new ObjectOutputStream(out);
            object_out.writeObject(成绩表);
            object_out.close();
            out.close();
        }
        catch(IOException e)
        {
        }
    }
    if(!file 高级.exists())
    {
        try{
            FileOutputStream out=new FileOutputStream("高级记忆排行榜.txt");
```

```
        ObjectOutputStream object_out=new ObjectOutputStream(out);
        object_out.writeObject(成绩表);
        object_out.close();
        out.close();
    }
    catch(IOException e)
    {
    }
}
gradeFile=file 初级;
setBounds(100,100,300,260);
setVisible(true);
addWindowListener(new WindowAdapter()
{
    public void windowClosing(WindowEvent e)
    {
        System.exit(0);
    }
});
con=getContentPane();
记忆测试板=new MemoryTestPane(block,icon,m,n,gradeFile);
提示条=new JTextField(" 初级: 您需要连续找出 "+6+" 个相同图标的方块");
提示条.setEditable(false);
提示条.setForeground(Color.red);
con.add(记忆测试板, BorderLayout.CENTER);
con.add(提示条, BorderLayout.SOUTH);
con.validate();
this.validate();
}
public void 给出级别测试(int 宽,int 高,File f)
{
    m=宽;
    n=高;
    图标个数=m;
    gradeFile=f;
    block=new Block[m*n];
    icon=new ImageIcon[图标个数];
    for(int i=0;i<icon.length;i++)
    {
        icon[i]=new ImageIcon(" a" +i+" .gif");
    }
    for(int i=0;i<block.length;i++)
    {
        block[i]=new Block();
        block[i].设置关闭时的图标(new ImageIcon(" 关闭.gif"));
    }
    记忆测试板=new MemoryTestPane(block,icon,m,n,gradeFile);
    con.removeAll();
    con.add(记忆测试板, BorderLayout.CENTER);
}
```



```
        con.add(提示条, BorderLayout.SOUTH);
        con.validate();
        this.validate();
    }
    public void actionPerformed(ActionEvent event)
    {
        if(event.getSource()==初级)
        {
            给出级别测试(5,6,file 初级);
            setBounds(100,100,300,260);
            this.validate();
            提示条.setText(" 初级: 您需要连续找出" +6+" 个相同图标的方块");
        }
        if(event.getSource()==中级)
        {
            给出级别测试(6,7,file 中级);
            setBounds(100,100,340,280);
            this.validate();
            提示条.setText(" 中级: 您需要连续找出" +7+" 个相同图标的方块");
        }
        if(event.getSource()==高级)
        {
            给出级别测试(7,8,file 高级);
            setBounds(100,100,360,300);
            this.validate();
            提示条.setText(" 高级: 您需要连续找出" +8+" 个相同图标的方块");
        }
        if(event.getSource()==高级排行榜)
        {
            显示成绩表对话框=new ShowRecord(this,file 高级);
            显示成绩表对话框.setVisible(true);
        }
        if(event.getSource()==中级排行榜)
        {
            显示成绩表对话框=new ShowRecord(this,file 中级);
            显示成绩表对话框.setVisible(true);
        }
        if(event.getSource()==初级排行榜)
        {
            显示成绩表对话框=new ShowRecord(this,file 初级);
            显示成绩表对话框.setVisible(true);
        }
    }

    public static void main(String args[])
    {
        new Memory();
    }
}
```