



Java

程序设计语言课程设计丛书

Java课程设计

黄明 梁旭 周绍斌 编著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

程序设计语言课程设计丛书

Java 课程设计

黄 明 梁 旭 周绍斌 编著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书是专门为课程设计而编写的。内容包括语言介绍、课程设计指导和例子。每个例子的内容都相对独立,且其程序全部调试通过。本书的例子有贪食蛇游戏、俄罗斯方块游戏、个人信息助理工具、家庭理财工具、图书管理系统、人事档案管理系统、学生成绩管理系统共7个课程设计的例子。

本书适合高等院校相关专业的学生,也适用于高职、高专及成人教育等不同层次的人员。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

Java 课程设计 / 黄明, 梁旭, 周绍斌编著. —北京: 电子工业出版社, 2006.6

(程序设计语言课程设计丛书)

ISBN 7-121-02742-9

I. J… II. 黄①… ②梁… ③周… III. Java 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 060844 号

责任编辑: 赵 平

印 刷: 北京市顺义兴华印刷厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

经 销: 各地新华书店

开 本: 787×1 092 1/16 印张: 14.25 字数: 364.8 千字

印 次: 2006 年 6 月第 1 次印刷

印 数: 5 000 册 定价: 19.80 元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。
联系电话: (010) 68279077。质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

前 言

课程设计工作是高等院校人才培养计划的重要组成部分，是对学生专业知识、综合素质和实际能力训练的阶段，是人才培养质量的重要体现。

目前，为培养实用型人才，计算机专业与非计算机专业的课程设计量非常大，但课程设计方面的辅导材料却极少，使学生在做课程设计时遇到了很大困难，为此，我们编写了这套“程序设计语言课程设计丛书”。

参与编写本套丛书的老师都是多年从事课程设计指导的高校教师，他们既有丰富的实际开发经验，又真切地了解大多数学生在学习中的缺欠，针对性很强。

本丛书的特点如下：

1. 基本覆盖了当前比较流行的软件开发语言。
2. 每本书包括语言介绍、课程设计指导以及 6~8 个课程设计例子，每一个例子内容都相对独立。每个例子包括需求分析、总体设计、详细设计等。
3. 对每个例子都进行详细分析，突出指导性与可操作性相结合的特点。
4. 书中的程序全部调试通过，放在电子工业出版社网站上。

由于编者水平有限，编写时间仓促，书中错误和不妥之处在所难免，敬请读者和专家批评指正。

读者在使用本书的过程中如有问题，可与下列 E-mail 联系：dlhm@263.net

编 者
2006 年 3 月

目 录

第 1 章 概述	1
1.1 Java 简介	1
1.1.1 什么是 Java	1
1.1.2 Java 的主要特点	1
1.2 课程设计指导	3
1.2.1 问题定义	3
1.2.2 可行性研究	3
1.2.3 需求分析	3
1.2.4 总体设计	4
1.2.5 详细设计	4
1.2.6 编码	5
1.2.7 综合测试	5
1.2.8 维护	6
1.3 课程设计选题及考核	7
1.3.1 课程设计选题	7
1.3.2 课程设计考核	7
第 2 章 贪食蛇游戏	9
2.1 需求分析	9
2.1.1 系统概述	9
2.1.2 系统运行环境	9
2.1.3 功能需求描述	9
2.2 总体设计	10
2.2.1 开发与设计的总体思想	10
2.2.2 系统模块结构图	10
2.2.3 模块设计	10
2.2.4 系统流程描述	11
2.2.5 界面设计	12
2.3 详细设计	13
2.3.1 游戏区模块	13
2.3.2 游戏控制模块	26
2.3.3 级别设置模块	27
2.4 小结	29
第 3 章 俄罗斯方块游戏	31
3.1 需求分析	31

3.1.1	系统概述	31
3.1.2	系统运行环境	31
3.1.3	功能需求描述	31
3.2	总体设计	32
3.2.1	开发与设计的总体思想	32
3.2.2	系统模块结构图	32
3.2.3	模块设计	32
3.2.4	系统流程描述	34
3.2.5	界面设计	34
3.3	详细设计	35
3.3.1	主游戏区模块	35
3.3.2	游戏控制模块	55
3.3.3	初始级别设置模块	56
3.4	小结	58
第 4 章	个人信息助理	59
4.1	需求分析	59
4.1.1	系统概述	59
4.1.2	系统运行环境	59
4.1.3	功能需求描述	59
4.2	总体设计	60
4.2.1	开发与设计的总体思想	60
4.2.2	系统模块结构图	60
4.2.3	模块设计	60
4.2.4	系统流程描述	62
4.2.5	界面设计	62
4.2.6	数据库设计	64
4.3	详细设计	65
4.3.1	系统主控平台	65
4.3.2	日常记事模块	69
4.3.3	通讯簿模块	76
4.3.4	密码备忘模块	81
4.4	小结	85
第 5 章	家庭理财工具	87
5.1	需求分析	87
5.1.1	系统概述	87
5.1.2	系统运行环境	87
5.1.3	功能需求描述	87
5.2	总体设计	88
5.2.1	开发与设计的总体思想	88

5.2.2	系统模块结构图	88
5.2.3	模块设计	88
5.2.4	系统流程描述	89
5.2.5	界面设计	90
5.2.6	数据库设计	92
5.3	详细设计	93
5.3.1	系统主控平台	93
5.3.2	家庭收支管理模块	97
5.3.3	家庭资产管理模块	111
5.4	小结	117
第 6 章	图书管理系统	119
6.1	需求分析	119
6.1.1	系统概述	119
6.1.2	系统运行环境	119
6.1.3	功能需求描述	119
6.2	总体设计	120
6.2.1	开发与设计的总体思想	120
6.2.2	系统模块结构图	120
6.2.3	模块设计	120
6.2.4	系统流程描述	122
6.2.5	界面设计	122
6.2.6	数据库设计	124
6.3	详细设计	125
6.3.1	系统主控平台	126
6.3.2	用户管理模块	129
6.3.3	借阅管理模块	133
6.3.4	书库管理模块	142
6.3.5	读者管理模块	144
6.4	小结	146
第 7 章	人事档案管理系统	147
7.1	需求分析	147
7.1.1	系统概述	147
7.1.2	系统运行环境	147
7.1.3	功能需求描述	147
7.2	总体设计	148
7.2.1	开发与设计的总体思想	148
7.2.2	系统模块结构图	148
7.2.3	模块设计	148
7.2.4	系统流程描述	149

7.2.5	界面设计	149
7.2.6	数据库设计	152
7.3	详细设计	154
7.3.1	系统主控平台	154
7.3.2	权限验证模块	158
7.3.3	口令维护模块	160
7.3.4	基本档案管理模块	163
7.3.5	人事变动管理模块	172
7.4	小结	175
第 8 章	学生成绩管理系统	177
8.1	需求分析	177
8.1.1	系统概述	177
8.1.2	系统运行环境	177
8.1.3	功能需求描述	177
8.2	总体设计	178
8.2.1	开发与设计的总体思想	178
8.2.2	系统模块结构图	178
8.2.3	模块设计	178
8.2.4	系统流程描述	180
8.2.5	界面设计	180
8.2.6	数据库设计	183
8.3	详细设计	186
8.3.1	系统主控平台	186
8.3.2	成绩管理模块	190
8.3.3	考生名单管理模块	200
8.3.4	基础数据管理模块	208
8.3.5	数据库管理模块	213
8.4	小结	216
附录 A	Java 运行环境的安装与配置	217
A.1	JDK 的安装与配置	217
A.2	ODBC 数据源的配置	218
参考文献		220

第 1 章 概 述

1.1 Java 简介

1.1.1 什么是 Java

Java 是由 Sun Microsystem 公司开发的新一代编程语言，其最早被设计用于开发消费类电子产品的内部软件。1995 年后 Java 被重新设计，Java 程序可以嵌入 HTML 网页，给 Web 用户带来生动的界面和交互方式。由于 Java 的免费，以及简单、跨平台、面向对象等诸多优点，迅速得到了业界的认同，已经成为目前应用领域最广泛的编程语言之一。

早期 Java applet 已经让广大互联网用户为之动容，但 Java 同样也可以和其他编程语言一样开发独立应用程序。目前，由于 Java 语言的可移植性、多线程机制以及网络性能，它在开发中间件方面有其得天独厚的优势，是其最主要的应用领域。同时 Java 也被广泛应用于嵌入式系统开发，它正在逐渐编程开发手持设备、互联网信息站、车载计算机等应用的标准。总之，Java 以其独到的特性已经成为广大程序员的首要选择。

1.1.2 Java 的主要特点

“一次编写，到处运行”是人们形容 Java 语言时最常用的一句话，也是 Java 最主要的特点之一。在 Sun 公司的 Java 技术白皮书中对 Java 的特点有一段更全面的描述：Java 是简单的（simple）、面向对象的（object-oriented）、分布式的（distributed）、解释型的（Interpreted）、健壮的（robust）、安全的（secure）、结构中立的（architecture-neutral）、可移植的（portable）、高效的（high-performance）、多线程的（multithreaded）、动态的（dynamic）。下面对这些特点简要分析一下。

1. Java 是简单的

Java 的简单是与 Java 之前最主流的 C++ 相比而言的，Java 是在 C++ 基础上发展而来的，它保留了 C++ 的大部分优点，同时也进行了简化和改进，剔除了一些很少使用、难于理解的以及易混淆的特性。比如 Java 去掉了指针，并取消了类的多重继承，转而在更简单的接口来取代多重继承。另外，Java 也改进了 C++ 在内存方面的不足，Java 采用了自动内存分配和回收，使得编写一个正确的程序变得更简单了。

2. Java 是面向对象的

目前的程序设计语言主要分两类：面向过程和面向对象。比如 Pascal 语言和 C 语言就是典型的面向过程式的，而 Java 是面向对象式的。面向对象与面向过程相比，面向对象的程序设计更符合人的自然思维方式，程序设计更容易，效率也更高。

3. Java 是分布式的

所谓分布式系统是指多台计算机通过网络连接在一起工作并彼此分享资源。由于 Java 从一开始就设计了网络工作能力，在 Java 中编写网络程序就如同在本地文件中存取数据一样简单，因此使用 Java 使得分布式计算变得更容易，Java 本身就支持分布式计算。

4. Java 是解释型的

Java 程序在运行之前需要先将 Java 源代码编译成字节码 (Bytecode)，此种字节码与机器无关，然后通过一个解释器来运行字节码。Java 编译成字节码之后，能够在任何具有 Java 解释器的机器上运行。

5. Java 是健壮的

虽然 Java 语言不能保证用 Java 编写的程序绝对可靠，但同 C++ 相比，Java 程序的可靠性要高得多。Java 在编译阶段能够检查出大多数错误。此外，它去掉了指针类型，能够避免因重写内存而造成数据破坏的可能性。而且，Java 的异常处理机制强制程序员编写处理异常的代码，能够最大程度的提高程序的健壮性。

6. Java 是安全的

当一个 Java 程序执行时，JVM (Java Virtual Machine) 可以监视它的动作，只要它有异常动作，JVM 就会阻止，这使得 Java 比 C++ 程序更安全。比如，在 Java applet 中就不允许对本地系统的文件进行读写。

7. Java 是结构中立的

体系结构中立，也叫做平台无关，是 Java 最主要的优势之一。通过 Java 可以编写在任何平台上都能够运行的程序。

Java 的成功基础在于有网页设计能力，使用 Java 编写的 applet 小程序，可以在任何嵌有 Java 虚拟机的 Web 浏览器上运行，利用 Java 解释器，也可以编写在任何操作系统上独立运行的 Java 程序。

8. Java 是可移植的

可移植也就是人们常说的“Java 程序一次编写，处处运行”。Java 程序编译成字节码之后，可以在任何具有 Java 解释器的操作系统上运行，具有很强的可移植性。

9. Java 是高效的

Java 的执行速度的确不如 C++ 快，主要原因是 Java 是解释型的，字节码不是直接在操作系统上运行，而是通过解释器运行。然而，Java 的执行速度并不是让人无法忍受，事实上 Java 的速度能够满足大多数交互程序的需要，而且，CPU 在这几年中已有惊人的进步，并且这种趋势还将继续下去。Java 本身在其运行效率方面也有长足的进步，它利用实时技术将 JVM 的执行效率提高 10 到 20 倍，已经接近 C++。

10. Java 是多线程的

多线程是指一个程序具有同时执行多个任务的能力。多线程对于 Java 而言是与生俱来的，它本身很好地集成了多线程技术，而其他语言必须调用操作系统特定的过程才能实现。

11. Java 是动态的

Java 被设计为能够适应发展的环境，可以任意在原类基础上增加新的方法或者属性而不影响原来程序的运行，而且，Java 根据运行需要来动态装载需要的类。

1.2 课程设计指导

在课程设计过程中，应该遵循软件工程的思想，按软件工程的流程来指导学生工作的开展。软件工程是指导计算机项目开发与维护的工程学科。软件项目的开发应该遵循软件工程标准，这样可以提高软件开发的效率，减少软件开发与维护中的问题。

一个计算机软件，从开始构思起，经过该软件开发成功投入使用，直到最后决定停止使用，并被另一个软件代替时为止，称为该软件的一个生命周期。典型的软件生命周期包括以下列阶段。

1.2.1 问题定义

问题定义阶段必须回答的问题是：你要解决的问题是什么？具体包括：项目的名称、背景、开发该系统的现状、项目的目标等。

1.2.2 可行性研究

可行性研究的目的是用最小的代价确定在问题定义阶段所确定的系统的目标和规模是否能实现，所确定的问题是否可以解决，系统方案在经济上、技术上和操作上是否可以接受。

典型的可行性研究有下列具体步骤：

- (1) 确定项目规模和目标。
- (2) 研究正在运行的系统。
- (3) 建立新系统的高层逻辑模型。
- (4) 导出和评价各种方案。
- (5) 推荐可行的方案。
- (6) 编写可行性研究报告。

1.2.3 需求分析

需求分析是指开发人员要准确理解用户的要求，进行细致的调查分析，将用户非形式的需求陈述转化为完整的需求定义，再由需求定义转换到相应的形式功能规约（需求规格说明）的过程。需求分析虽处于软件开发过程的开始阶段，但它对于整个软件开发过程以及软件产品质量是至关重要的。

需求分析的基本任务是要准确地定义新系统的目标，回答系统必须“做什么”的问题。

结构化分析（Structure Analysis，简称 SA），是面向数据流进行需求分析的方法。SA 也是一种建模活动，该方法使用简单易读的符号，根据软件内部数据传递、变换的关系，自顶向下逐层分解，描绘出满足功能要求的软件模型。

SA 分析步骤

(1) 了解当前系统的工作流程，获得当前系统的物理模型。当前系统指目前正在运行的系统，可能是需要改进的正在计算机上运行的软件系统，也可能是人工处理系统。

(2) 抽象出当前系统的逻辑模型。物理模型反映了系统“怎样做”的具体实现，去掉物理模型中非本质的因素，抽象出本质的因素。所谓的本质因素是指系统固有的、不依赖运行环境变化而变化的因素，任何实现均可这样做。非本质因素不是固有的，它随环境不同而不同，随实现不同而不同。

(3) 建立目标系统的逻辑模型。目标系统指待开发的系统。分析、比较目标系统与当前系统逻辑上的差别，然后对“变化的部分”重新分解，分析人员根据自己的经验，采用自顶向下逐步求精的分析策略，逐步确定变化部分的内部结构，从而建立目标系统的逻辑模型。

(4) 进一步补充和优化。为了完整描述目标系统，还要做一些补充：说明目标系统的人机界面；说明至今尚未详细考虑的细节。

1.2.4 总体设计

软件总体设计的基本任务如下。

1. 设计软件系统结构（简称软件结构）

为了实现目标系统，首先进行软件结构设计，具体为：

- (1) 采用某种设计方法，将一个复杂的系统按功能划分成模块。
- (2) 确定每个模块的功能。
- (3) 确定模块之间的调用关系。
- (4) 确定模块之间的接口，即模块之间传递的信息。
- (5) 评价模块结构的质量。

2. 数据结构及数据库设计

(1) 数据结构的设计。采用逐步细化的方法设计有效的数据结构，将大大简化软件模块处理过程的设计。

(2) 数据库设计。数据库设计指数据存储文件的设计，主要进行概念、逻辑和物理三方面的设计。

(3) 编写概要设计文档。

1.2.5 详细设计

详细设计阶段主要确定每个模块的具体执行过程，其主要任务有：

- (1) 为每个模块进行详细的算法设计。
- (2) 为模块内的数据结构进行设计。
- (3) 对数据库进行物理设计，即确定数据库的物理结构。
- (4) 其他设计。根据软件系统的类型，可能还要进行代码设计、输入/输出格式设计和人机对话设计。
- (5) 编写详细设计说明书。

(6) 评审。

1.2.6 编码

编码是用选定的程序设计语言实现详细设计。通常选择程序设计语言考虑的因素如下。

1. 项目的应用领域

这是选择语言的关键因素，项目应用领域一般为以下几种类型：

- (1) 科学与工程计算。可供选用的语言有：FORTRAN 语言，Pascal 语言，C 语言，PL/I。
- (2) 数据处理与数据库应用。可供选用的语言有：Cobol 语言，SQL，4GL
- (3) 实时处理。可供选用的语言有：汇编语言，Ada 语言。
- (4) 系统软件。如果编写操作系统、编译系统等系统软件时，可选用汇编语言、C 语言、Pascal 语言和 Ada 语言。
- (5) 人工智能。如果要完成知识库系统、专家系统、决策支持系统、推理工程、语言识别、模式识别、机器人视角、自然语言处理等人工智能领域内的系统，应选择 Prolog 语言、Lisp 语言。

2. 软件开发的方法

有时编程语言的选择依赖于开发的方法，如果要用快速原型模型来开发，要求能快速实现原型，因此宜采用 4GL。如果是面向对象方法，宜采用面向对象的语言编程，面向对象的语言主要有：C++、Java。

1.2.7 综合测试

软件测试的目的是为了发现程序中的错误而执行程序的过程。一个好的测试用例能够发现至今尚未发现的错误，一次成功的测试应该是发现了至今为止尚未发现的错误。

1. 测试的指导原则

在软件测试中，应注意以下指导原则：

- (1) 测试用例应由输入数据和预期的输出数据两部分组成。
- (2) 测试用例不仅选用合理的输入数据，还要选择不合理的输入数据。
- (3) 除了检查程序是否做了它应该做的事，还应该检查程序是否做了它不应该做的事。
- (4) 应制定测试计划并严格执行，排除随意性。
- (5) 长期保留测试用例。
- (6) 对发现错误较多的程序段，应进行更深入的测试。
- (7) 程序员避免测试自己的程序。

2. 测试方法

软件测试方法一般分为两大类：动态测试方法与静态测试方法。而动态测试方法中又根据测试用例的设计方法不同，分为黑盒测试与白盒测试两类。

(1) 静态测试

静态测试指被测试程序不在机器上运行，而是采用人工检测和计算机辅助静态分析的手段对程序进行检测。

(2) 动态测试

动态测试指通过运行程序发现错误。一般意义上的测试大多是指动态测试。对软件产品进行动态测试时，也采用两种方法，分别称为黑盒测试法和白盒测试法。

a) 黑盒测试法

该方法把被测试对象看成一个黑盒子，测试人员完全不考虑程序的内部结构和处理过程，只在软件的接口处进行测试，依据需求规格说明书，检查程序是否满足功能要求。因此，黑盒测试又称为功能测试或数据驱动测试。

通过黑盒测试主要发现以下错误：

- ① 是否有不正确或遗漏了的功能。
- ② 在接口上，能否正确地接受输入数据，能否产生正确的输出信息。
- ③ 访问外部信息是否有错。
- ④ 性能上是否满足要求等等。

b) 白盒测试法

该方法把测试对象看做一个打开的盒子，测试人员需了解程序的内部结构和处理过程，以检查处理过程的细节为基础，对程序中尽可能多的逻辑路径进行测试，检验内部控制结构和数据结构是否有错，实际的运行状态与预期的状态是否一致。

1.2.8 维护

软件维护是在软件交付使用以后对它所做的改变。软件维护的内容有四种：校正性维护、适应性维护、完善性维护和预防性维护。

1. 校正性维护

在软件交付使用后，由于在软件开发过程中产生的错误并没有完全彻底地在测试中发现，因此必然有一部分隐含的错误被带到维护阶段上来。这些隐含的错误在某些特定的使用环境下会暴露出来。为了识别和纠正错误，修改软件性能上的缺陷，应进行确定和修改错误的过程，这个过程就称为校正性维护。这种维护活动占整个维护工作的 21%。

2. 适应性维护

为了使应用软件适应某种变化而修改软件的过程称为适应性维护。这种维护活动占整个维护活动的 25%。

3. 完善性维护

在软件漫长的运行时期中，用户往往会对软件提出新的功能要求与性能要求。这是因为用户的业务会发生变化，组织机构也会发生变化。为了适应这些变化，应用软件原来的功能和性能需要扩充和增强。这种增加软件功能、增强软件性能、提高软件运行效率而进行的维护活动称为完善性维护。这种维护性活动数量较大，占整个维护活动的 50%。

4. 预防性维护

为了提高软件的可维护性和可靠性而对软件进行的修改称为预防性维护。这种维护活动只占很小的比例，为4%。

1.3 课程设计选题及考核

1.3.1 课程设计选题

恰当的选题是开展好课程设计的前提，选题要结合所学专业，对所选课题，要认真进行调查研究、系统分析。选题应当符合以下要求：

- (1) 选题必须符合计算机专业培养目标的要求、体现本专业的特色。
- (2) 所选课题应尽量使用最近学习的开发工具，并结合其教授课程的知识点、内容进一步延伸，在实用方面有更高、更强的针对性。
- (3) 题目要具有较好的代表性。选题应尽可能结合生产、科研、管理、教学等方面的实际需要，也可以选用符合教学要求的模拟题目。
- (4) 选题的难易程度要适当。以学生可以在2~4周的规定时间内完成为宜。
- (5) 选题一般由指导教师下达，可以每人独立完成相同的题目，也可以根据选题难度情况，对学生进行分组，比如5~6人一组，每组共同完成一题。

1.3.2 课程设计考核

在课程设计的课程考核中，一般采用五级（优、良、中、及格、不及格）评分制，最后成绩依据课程设计过程中学生完成情况、最后递交的软件源码和设计报告质量来进行评定。

在设计过程中，可以由老师进行两到三次的进度检查，记录学生的工作进展情况。采用进度检查的方式能够更好地控制学生设计工作的真实性，可以作为指导老师最后评定成绩的一个重要标准。

学生在设计课程结束后，上交一份课程设计报告，同时把个人完成的软件源码一起上交给指导老师，作为指导老师评定最后成绩和成绩复查时的资料。指导老师可以根据课程特点，要求学生上交其他文档和资料。

课程结束后，指导老师给出成绩，并根据情况填写课程设计总结报告。

第2章 贪食蛇游戏

2.1 需求分析

贪食蛇游戏是人们最熟悉的小游戏之一，它以其趣味性强、易上手等特点赢得了大众的认可。本游戏软件的目标是为用户提供一个可在普通个人电脑上运行的、界面美观的、易于操作的贪食蛇游戏。

2.1.1 系统概述

该游戏软件在 Windows 系统上为玩家提供传统贪食蛇游戏的基本功能，玩家可以通过键盘控制游戏区中贪食蛇的运动，当蛇头碰到系统产生的食物时加长蛇身，当蛇碰到墙壁或者自己身体时，游戏结束。游戏设定 3 个级别，级别越高蛇运动的速度越快、难度越大，玩家可以根据自己的需要自由设定初始游戏级别，当玩家取得的分数到达一定水平时游戏自动进入下一个级别，达到最高级别时不再增加。游戏以最终玩家获得的分数来判断玩家水平的高低。

2.1.2 系统运行环境

1. 硬件环境

- 处理器： Intel Pentium 166 MX 或更高
- 内存： 32MB 以上
- 硬盘空间：1GB 以上
- 显卡： SVGA 显示适配器

2. 软件环境

- 操作系统： Windows 98/ME/2000/XP

2.1.3 功能需求描述

本游戏软件的总体目标是按照特定游戏规则为玩家提供一个方便友好的游戏界面，同时满足不同级别的玩家的不同难度需求。游戏规则如下：

- 玩家通过键盘控制蛇在游戏区中运动。
- 当蛇头碰到软件提供的食物或者宝物时，得分并增加蛇身长度一个单位。
- 当蛇头碰到蛇身或墙壁时，如果没有相应的宝物则游戏结束。
- 游戏设定两种宝物，一种为穿身宝物，另一种为穿墙宝物。
- 记分规则：吃掉一个食物或者宝物加 100 分
- 晋级规则：累计得 2000 分自动进入下一个级别。