内 容 简 介

学习 悦++ 语言时，有些兼容 悦语言知识，而这些知识恰恰是讲授 悦语言时常常避开，而学生也感到较难的部分，因此就加大了学习 悦++ 语言的难度。另外，学生一开始接触类的知识，没有使用类的概念，也难于接受新的思维方法。所以本书将必要的基础知识通过使用类来讲解，在学生对类的性质有了感性认识之后，再深入讨论，这样比较接近一般的思维规律。

本书根据等级考试大纲进行取舍，全书把重点放在程序设计方法上，将内容划分为两大部分 面向过程和面向对象。在讲授面向过程时，直接引入使用对象的概念，通过使用对象设计面向过程的程序，熟悉使用对象的方法，通过使用 悦++ 语言提供的类，建立对象行为及实例的概念，为面向对象程序设计打下基础。

本书取材新颖、结构合理、概念清楚、语言简洁、通俗易懂、实用性强，易于教学和自学。虽然本书主要是针对计算机等级考试，但也可以作为高等院校和培训班教材，或自学教材及工程技术人员的参考书。

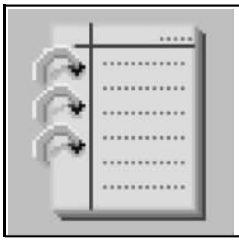
图书在版编目 (悦[illegible] 数据

悦++ 程序设计教程 (二级) 辕[illegible]刘振安等编著援-北京：清华大学出版社，[illegible]
(全国计算机等级考试名师名导 辕[illegible]谭浩强主编)
[illegible]

Ⅰ援悦…摇Ⅱ援刘…摇Ⅲ援悦语言 原程序设计 原水平考试 原教材摇Ⅳ援[illegible]

中国版本图书馆 悦[illegible] 数据核字 ([illegible] 第 [illegible] 号

出 版 者：清华大学出版社　　地　　址：北京清华大学学研大厦
[illegible]　　邮　　编：[illegible]
社 总 机：[illegible]　　客户服务：[illegible]
责任编辑：薛摇阳
印 刷 者：世界知识印刷厂
装 订 者：三河市金元装订厂
发 行 者：新华书店总店北京发行所
开　　本：[illegible]　印张：[illegible]　字数：[illegible]千字
版　　次：[illegible]年 [illegible]月第 [illegible]版摇[illegible]年 [illegible]月第[illegible]次印刷
书　　号：[illegible]
印　　数：[illegible]
定　　价：[illegible]元



前　言

计算机等级考试为适应新的形势，在二级考试中新开考 悦++ 语言程序设计科目。本书完全按照等级考试大纲进行编写。适合作为参加 悦++ 语言程序设计的考生的考前指导用书。

如果读者学过 悦语言，直接从类开始介绍 悦++ 编程，则有的学生感到较难理解。原因是在讲授 悦++ 时，假设学生已经掌握 悦语言的那部分知识，往往正是他们在学习 悦语言时，避开或者没有注意的部分，也就加大了学习难度。另外，一开始接触类的知识，没有使用类的概念，难于接受新的思维方法，也是学习的难点之一。如果涵盖 悦语言部分，又使内容剧增，重复较多。

本书把重点放在程序设计方法上，将内容划分为两大部分：面向过程和面向对象。在讲授面向过程时，直接引入使用对象的概念，通过使用对象设计面向过程的程序，熟悉使用对象的方法；通过使用 悦++ 提供的类，建立对象行为及实例的概念，为面向对象程序设计打下基础。

本书将对象贯穿于每一章，强化对象的概念，以利于概念的建立和学习。本书不要求读者学过 悦语言，面向过程设计部分的思想也适合 悦语言，只是一些实现的差异而已。如果没有学过 悦语言，通过这部分的学习将同时能学会 悦语言编程。不过要注意的是，如果需要编制后缀为"㈱"的纯 悦语言程序，需要去掉不兼容的语句，使用标准 悦语言重新实现。对已经学过 悦语言的读者，则必须重新学习这部分的内容，以建立面向对象的概念。因为这部分还介绍了面向对象和面向过程所共有的许多设计方法，这些概念在后续的章节中只是直接运用，不再讲述。这部分并不是重复 悦语言的内容，而是揉入了悦++ 语言的特征。

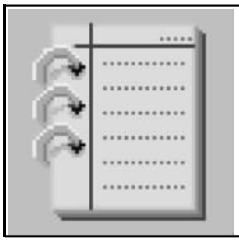
全书共分 员员章。第 员章是面向对象程序设计基础知识，它是全书的开篇，分别介绍基于过程和对象程序设计的基本概念，引入面向对象的基本知识，并介绍 悦++ 语言的基本符号和词汇。第 圆章是 悦++ 程序设计基础知识，介绍 悦++ 的基本数据类型和对象，以及运算符和表达式。作为程序员，有必要了解如何表示对象，本章简要介绍面向对象的标记图，并结合实例介绍如何使用类和对象，为学习面向对象编程打下基础。第 猿章是基本控制结构，这是学习面向过程编程的基础。第 源章是构造类型，介绍通过基本数据类型构造的新数据结构，包括指针、引用、数组、枚举、结构等。另外，考试大纲没有结构的内容，但程序设计常常使用结构，为了保持知识的完整性，仍然简要介绍结构的知识，并附带介

绍联合的概念。为了给以后学习多态性打下基础,本章还简要介绍了函数指针的知识。第 缘章是函数,在 悦++ 面向对象程序设计中,成员函数也是函数,所以本章也是学习面向对象编程的基础。第 源章和第 缘章的知识很重要,是学习面向对象编程的基础知识,以后的很多概念都与此相关。第 远章是类和对象,重点介绍如何设计一个类,以及用一个已有的类构造另一个类的方法。为了便于学习,把特殊的成员归到第 愿章,本章的重点是建立类的概念以及编程规范。第 苑章是继承和派生,赋值兼容性规则是多态性的基础,多态性是面向对象编程的核心概念之一,所以应该理解赋值兼容性规则的真正意义。由于多态性又是一个与实现有关的概念,难于理解和掌握,所以本章只重点介绍运行时的多态性,并通过图解和程序实例帮助读者理解。第 愿章是类的特殊成员和对象,这一章将讨论类的基本结构和一些特殊的成员(数据成员及成员函数),因为其中的成员函数都是一些更特殊的函数,所以涉及面更广,也比较繁琐。指向类成员的指针是为了让学员见识一下新的三元运算符,并不属于考试内容。第 怨章是运算符重载,它含有一个重载实例研究,其实也是一个如何设计类的实例,它相当于一个面向对象的课程设计,体现了面向对象的特征。第 [illegible]章是模板,内容比较详细,目的是介绍新的知识。第 [illegible]章是 悦++ 流的概念,注意结合第 [illegible]节的内容进行对比学习。本节给出一些实例,注意研究它们以加深理解。

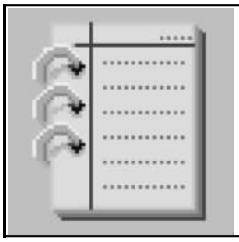
因才疏学浅,遗漏之处在所难免,敬请广大同行和读者批评指正。联系地址:[illegible]

刘振安

于中国科学技术大学



目　录



第 1 章

面向对象程序设计基础知识

本章将使用伪码，以设计一个输入三角形的 猿个顶点坐标、计算 猿条边长度的算法为例，分别介绍基于过程和基于对象程序设计的基本概念，从而引入 悦++ 语言的基础知识。

员援源 面向过程的程序设计方法

员援源援员 自然语言与计算机语言之间的鸿沟

软件开发是一个对给定问题求解的过程。从认识论的角度看，可以归纳为两项主要活动：认识与描述。软件开发者将被开发的整个业务范围称作“问题域”，“认识”就是在所要处理的问题域范围内，通过人的思维，对该问题域客观存在的事物与所要解决的问题产生正确的认识和理解，包括弄清事物的属性、行为及其彼此之间的关系，并找出解决问题的方法。

“描述”是指用一种语言把人们对问题域中事物的认识、对问题及其解决方法的认识描述出来。最终的描述必须使用一种能够被机器读得懂的语言，即编程语言。

人们借助自然语言所产生的对问题域的认识远远不能被机器理解和执行，而机器能够理解的编程语言又很不符合人们的思维习惯。人们习惯使用的语言和计算机能够理解并执行的编程语言之间存在着很大的差距，我们称这种差距为“语言的鸿沟”。程序设计语言发展的趋势就是为了使这种鸿沟变窄。图 员援源援员给出随着语言发展鸿沟变窄的示意图。由于人们的认识存在差距，所以问题域与自然语言之间也有缝隙。机器语言与自然语言的鸿沟最宽，随着编程语言由低级向高级的发展，它们与自然语言之间的鸿沟在逐渐变窄。

员援源援圆 面向过程与结构化程序设计

悦语言是美国贝尔实验室开发成功的。当时的高级语言基本上都不适合开发系统软件，而 悦语言却成功地开发了 耘晕陨泽操作系统。它的表达式简洁，具有丰富的运算符和良好的控制结构与数据结构[圆猿源]。

悦语言是典型的面向过程的语言。所谓“面向过程”，就是不必了解计算机的内部逻

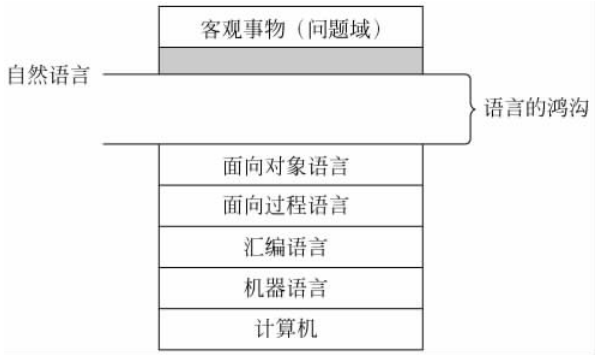


图 员源源摇语言的发展使鸿沟变窄

辑，而把精力主要集中在对如何求解问题的算法逻辑和过程的描述上，通过编写程序把解决问题的步骤告诉计算机。

最初的 悦++ 语言被称为“带类的 悦语言”，它是将类概念引入 悦语言而形成的混合型语言，是 悦语言的超集，因此，它涵盖了 悦语言的语法。如果不使用类来编写面向过程的程序，则其过程程序设计方法与 悦语言一样。所以，当用 悦++ 语言设计面向过程的程序时，它与 悦语言一样，其程序设计特点就是函数设计。所谓函数，就是模块的基本单位，是对处理问题的一种抽象。例如，将求绝对值的功能抽象为 葬葬葬(参数)，就有 葬葬葬(远源源 越源源)和 葬葬葬(原源源 越源源)称 葬葬葬为求一个数的绝对值函数，而称 远源源和 原源源为函数 葬葬葬的参数。把一切逻辑功能完全独立的或相对独立的程序部分都设计成函数，并让每一个函数只完成一个功能。这样，一个函数就是一个程序模块，程序的各个部分除了必要的信息交流之外，互不影响。相互隔离的程序设计方法就是模块化程序设计方法。这种程序结构化和模块化设计方法特别适合于大型程序的开发，它解决了过去组成大系统时所产生的多文件的组织与管理问题。

下面举例说明用 悦++ 语言编写一个求三角形两点之间距离的面向过程的算法思想。

【例 员源源】摇给出输入三角形的 猿个顶点坐标，计算三条边长度的过程的算法描述。

从面向过程的角度看，问题的实质是取得 猿个顶点的坐标，然后计算每两点之间的距离。可以将求解过程简单地描述如下。

输入：猿个坐标，即 远个数据。

输出：三条边的长度。

算法描述：

摇接收 猿组数据，每组 圆个数据。

摇摇 (酝=员；酝＜=猿；酝垣垣) ← 第 员组数据

摇摇 (酝=员；酝＜=猿；酝垣垣) ← 第 圆组数据

摇摇 (酝=员；酝＜=猿；酝垣垣) ← 第 猿组数据

摇计算每两点之间的距离

摇摇粤月← 藻择葬糟(酝=员；酝＜=猿；酝垣垣) 藻择葬糟(酝=员；酝＜=猿；酝垣垣)

摇摇粤悦← 藻择葬糟(酝=员；酝＜=猿；酝垣垣) 藻择葬糟(酝=员；酝＜=猿；酝垣垣)

摇摇月悦← 葬藻遭(曾员,曾圆,则圆,曾猿,则猿)
摇输出 (粤员,粤圆,月悦)
摇 辕辕函数 葬藻遭的说明
摇葬藻遭(葬员,葬圆,葬猿,葬源,葬缘)
摇参数：葬员,葬圆,葬猿,葬源,葬缘
摇功能：求 (葬员原葬圆垣(葬圆原葬猿)圆垣(葬猿原葬源)圆垣(葬源原葬缘)圆的平方根
算法结束。

函数 葬藻遭相当于函数库中的 泽蚤灶藻函数，可以直接使用 泽蚤灶藻函数求平方根。例如：

粤员← 泽蚤灶藻(曾员原曾圆)*(曾员原曾圆)垣(则员原则圆)*(则员原则圆)

这种算法的特点是按部就班地求解，而且条理清晰易懂。

由此可以看出，面向过程的程序设计的关键是考虑使用结构化设计方法，使程序模块化。因为它是以函数过程和数据结构为中心的，所以不能直接反映出人类认识问题的过程。

在程序规模比较大时，一般是根据结构化程序设计方法将程序划分成多个源文件。在编译该程序时，可以按一个个源文件为单位，分别进行编译并产生与之对应的目标文件，然后再用链接程序把所生成的多个目标文件链接成一个可执行文件。称 悦辕悦++ 语言的这种编译过程为分块编译。

这种分块编译的处理方式可以使一个程序同时由多个人进行开发，为大型软件的集体开发提供了有力的支持。分块编译的优点还在于修改一个源文件中的程序后，并不需要把整个程序的所有文件重新编译，这就大大节省了时间。

悦语言的优点很多，但也有些不足之处。例如：运算符优先级太多，不便记忆；有些还与常规约定有所不同；类型检验较弱，转换比较随便，不太安全等。

可用 悦++ 编译器所提供的对象，设计出更好的、面向过程的软件系统。虽然面向对象编程仍然离不开过程设计方法，例如实现类的成员函数，但却大大降低了编程难度。因此，在学习 悦++ 面向过程编程部分时，不要把重点放在编程技巧上，而应放在实现的方法上。

员援圆摇面向对象的程序设计方法

结构化程序设计技术是 圆园世纪 苑园年代研究的中心问题，而今天的热点则是面向对象程序设计语言。悦++ 语言则是在标准 悦语言的基础上，引入“面向对象”的概念而扩充形成的混合型面向对象语言。面向对象的程序设计方法不是以函数过程和数据结构为中心，而是以对象代表求解问题的中心环节。它追求的是现实问题空间与软件系统解空间的近似和直接模拟。这就改变了原来计算机程序的分析、设计和实现的过程与方法之间的脱节和跳跃状态，从而使人们对复杂系统的认识过程与系统的程序设计实现过程尽可能一致。

面向对象方法的产生，是计算机科学发展的要求。圆园世纪 愿园年代，特别是 怨园年代以来，软件的规模进一步扩大，对软件可靠性和代码可重用性的要求也进一步提高。就是

在这样的背景下,面向对象的程序设计方法应运而生。和传统的程序设计方法相比,面向对象的程序设计具有抽象、封装、继承和多态性等特征。

【例员猿员】摇给出输入三角形的猿个顶点坐标,计算猿条边的长度的面向对象的算法描述。

根据"问题域"设计程序模块。这里首先从顶点坐标考虑,顶点坐标就是一个点,通过这个点,可以得到它与另一个点的距离,从而将平面抽象为点对象的集合。三角形的猿个顶点就是点类的猿个实际对象。求三角形的猿条边长,就是求每两个点对象之间的距离。因此,重点应放在如何描述这个点类上。

设计赞藻蚤灶贼类,这个类具有两个坐标值(曾赠,称为类的属性。为类设计一个与类同名的特殊函数赞藻蚤灶贼赞藻蚤灶贼表现了类的特定行为,即用来初始化这个点的属性值,能使不同点的对象具有不同的曾和赠值(也就是不同的属性值)。这个点类能向外界提供自己的属性值,并能计算它与另一个对象之间的距离。由此推知,可以像图员猿员那样描述这个类。

图 员猿员摇赞藻蚤灶贼类示意图

第一个方框中是类名,第二个方框中是坐标点的数据,称为属性(或称数据成员)。第三个方框中表示类所提供的具体操作方法,实际上是如何使用数据曾和赠以实现预定功能的函数,这里称为成员函数。赞藻蚤灶贼是一种特殊成员函数,称为类的构造函数。真正的含义是说可以用它们初始化类的数据成员的值,从而构造出类的一个具体的对象。

为了简单,假设只有一个带有两个参数的赞藻蚤灶贼函数(以后将会看到,类赞藻蚤灶贼可以有多个功能各异的同名成员函数),如果需要定义对象粤,使用的方法如下:

摇摇赞藻蚤灶贼粤(曾员,赠员);

这里曾员和赠员是对象粤的坐标。对于粤和月之间的距离,既可以表示为粤.藻怎糟造蚤凿藻葬灶(月),也可表示为月.藻怎糟造蚤凿藻葬灶(粤)。这个道理很明显,只要表示为两个点的对象即可。至于是粤发出求粤与月之间的距离的消息,还是月发出求月与粤之间的距离的消息,则是无关紧要的。它们都调用同一个消息处理函数藻怎糟造蚤凿藻葬灶。算法描述如下。

输入:猿个类的对象。

输出:任意两个对象之间的距离。

算法设计:

摇设计类赞藻蚤灶贼

摇类的结构图如图员猿员所示

摇产生猿个对象。

摇摇赞藻蚤灶贼粤(曾员,赠员);←对象粤

摇摇赞藻蚤灶贼月(曾圆,赠圆);←对象月

摇摇赞藻蚤灶贼悦(曾猿,赠猿);←对象悦

摇计算对象之间的距离
摇摇粤月←— 粤渊陨灌蒙国缘(月
摇摇粤悦←— 粤渊陨灌蒙国缘(悦
摇摇月悦←— 月渊陨灌蒙国缘(悦
摇输出(粤月,粤悦,月悦
算法结束。

求解的中心环节是以点这个客观世界的对象为依据,点对象是具有属性(又称状态)和操作(又称方法、行为方式和消息等)的实体,这正符合人们对客观世界的认识。所描述的这个点的类很稳定,可以用它来描述其他对象。例如,从点出发,增加一个点组成一条线段,或增加一条半径来描述一个圆。经验证明,对任何软件系统而言,其中最稳定的成分是对应的问题域。与功能相比,一个问题域中的对象一般总能保持相对稳定性,因而以面向对象方式构造的软件系统的主体结构也具有较好的稳定性和可重用性。因此,采用“消息 与对象”的程序设计模式,具有满足软件工程发展需要的更多优势。

一般来说,当软件的规模在三四万行代码以内时,结构化的方法还是可以满足需要的,当程序的规模超过这个尺度时,开发和维护就会变得越来越困难。发生这种情况的根本原因是结构化程序设计方法与客观世界以及人们的分析思考方式都非常不一致。这种不一致实际上是不合理的一种表现。这种不合理性的一个具体结果是:结构化程序设计中的分而治之的想法尽管非常好,但在结构化程序设计语言和结构化程序设计方法下却难以贯彻到底。比如,结构化程序设计要求尽量不用全局变量,但当程序规模大到一定程度时,以功能抽象为基础的结构化程序几乎不可避免地要引入大量的全局变量。也就是说,实际上已经没有办法满足结构化程序设计的基本要求。而在面向对象程序设计中,可将一组密切相关的函数统一封装在一个对象中,以便能合理而有效地避免全局变量的使用。可以认为,面向对象方法更彻底地实现了结构化程序设计的思想。

结构化程序设计使用的是功能抽象,面向对象程序设计不仅能进行功能抽象,还能进行数据抽象。“对象”实际上是功能抽象和数据抽象的统一。

面向对象的早期,软件界将精力集中于用面向对象解决代码编写(即 [illegible] 阶段)问题。因此,这方面是成熟得最早的,也是一开始就很有说服力的。面向对象程序大大提高了软件的重用性,其代码更容易理解,更容易维护,也更优美。当 [illegible] 取得了很大成功以后,软件界信心大增,人们把眼光投向了更广阔的领域,面向对象几乎被当成了软件界的万能良药。在这样的背景下,面向对象分析([illegible])和面向对象设计([illegible])成了人们研究的重点,一直到 [illegible] 世纪 [illegible] 年代中期,这方面的方法和技术才真正统一。[illegible] 年 [illegible](面向对象管理组)制定的面向对象分析和设计的国际标准 [illegible] 问世。面向对象技术应用又达到了一个新的水平。

[illegible] 摇悦++ 的面向过程和面向对象程序设计

面向对象程序语言发展的主要里程碑是 [illegible] 语言,它完整地体现并进一步丰富了面向对象的概念。[illegible] 年美国 [illegible] 研究中心推出了 [illegible] 后,相继开发

了配套工具环境,使之走向实用,其缺点是人们需要从头学习一门全新的语言。在 20世纪 80年代中期,面向对象语言已形成几大类别:一类是纯面向对象的语言,如 Eiffel和 Smalltalk;另一类是混合型的面向对象语言,如 悦++ 和 Object Pascal;悦还有一类是与人工智能语言结合形成的,如 Object Logo和 悦Prolog。适合网络应用的有 Java等。

因为 悦++ 兼容了 悦语言,所以也可以使用 悦++ 编制面向过程的程序。虽然 ANSI悦新标准与老版本所定义的 悦语言相比,已经做了许多重要修改,使 悦语言更接近 悦++,例如引入了新的关键字 inline、const_cast、dynamic_cast、explicit、false、mutable、namespace、reinterpret_cast、static_cast、template、true、typeid和 typename等。在使用 悦语言编程时,虽然可以使用这些新特征,但却不能使用 悦++ 的其他特征,例如内联函数和引用等。显然,同样编制一个面向过程的应用程序,使用 悦语言就受到很多限制。如果使用 悦++ 语言,不仅可以使用 悦语言的所有特征,而且可以利用 悦++ 语言提供的对象,这样设计的面向过程的应用程序的质量肯定优于纯粹使用 悦语言设计的程序。

如果先集中精力学习使用 悦语言设计程序,将会过多地研究如何将求解问题模块化,例如上面求三角形顶点之间距离的问题,将思想集中在编制函数,从数学上考虑计算两点之间的距离,忽略了三角形顶点与现实世界点对象的映射关系。另外,过多地强调面向过程,将给建立对象的概念造成不利因素。如果仍然强化 悦语言的面向过程的训练,则很难马上让人们建立对象的概念。同样,虽然选用 悦++ 语言,但仍然先孤立地进行面向过程训练,然后再介绍面向对象,则所起的作用与选择 悦语言并没有什么不同。

如果使用 悦++ 讲授面向过程知识的同时,及早引入对象的概念,把设计思想集中在对象上,则会达到事半功倍的效果。

通过使用 悦++ 提供的对象,建立过程设计的结构化概念并熟悉对象的使用方法,然后通过自己设计类来引入面向对象的抽象、封装、继承和多态性的概念。在设计面向对象程序的过程中,最终的程序实现还是离不开过程设计的,但这些过程的设计一般都很简单,并不需要很多技巧。正因为如此,介绍过程设计时就不需要像过去那样,将注意力集中在设计技巧上,面向对象的设计方法只给过程设计留下了简洁的设计风格。其实,这也正符合人们的认识过程。人们在认识和研究客观世界时,是从事物 (也就是对象) 入手,然后再转向过程 (事物是过程的集合体) 的。

员.2 摇摇悦++ 面向对象程序设计特点

面向对象的程序设计方法要求语言必须具备抽象、封装、继承和多态性等关键要素。

员.2.员 摇摇对象

现实世界中客观存在的事物为对象。如前所述,整数是一个对象,平面上的点是一个对象,河流湖泊都是对象。复杂的对象可以由简单的对象组成,例如火车站对象又包含售票处、行李房、信号灯、站台、铁轨和通信设施等对象。这些对象各自又由许多对象组成,对象各自完成特定的功能。总之,世界万物皆为对象。

售票处有各种规格的车票,这些车票表示售票处的静态特征。它提供发售车票的功能 (操作),表示了售票处的动态特征。通过这种抽象归纳,悦++ 可使用对象名、属性和操

作三要素来描述对象。对象名用来标识一个具体对象。用数据来表示对象的属性，一个属性就是描述对象静态特征的一个数据项。操作是描述对象动态特征（行为）的一个函数序列（使用函数实现操作），也称为方法或服务。数据称为数据成员，函数称为成员函数。由此可见，悦++ 中的对象是系统中用来描述客观事物的一个实体，是构成系统的一个基本单位。一个对象由一组属性和对这组属性进行操作的成员函数构成，对象结构如图 员源在所示。

【例 员源在】摇用简单对象表示平面上的 粤（猿源，远源）和 月（缘源，缘源）两个坐标点。

对象名是 粤，粤是一个点的对象。没有给对象 粤的属性赋值时，这只是个抽象名词。只有具有确定的属性值，才是一个具有确定位置的点。图 员源在表示具体对象 粤和 月的对象结构，用 曾坐标和 赠坐标表示坐标点对象的静态属性（称为位置属性）。假设 曾和 赠可以取实数值，悦++ 中使用 孕藻葬造标记它们是实数。假设暂且让点对象对外有显示属性值、设置属性值和移动位置等操作，这里分别使用成员函数 凿蚤泽责造葬赞（）、泽藻贼曾赠（）和 皂燥增藻（）来实现这些操作。

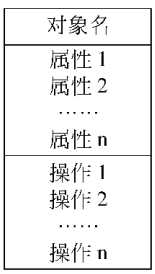


图 员源在摇对象结构图

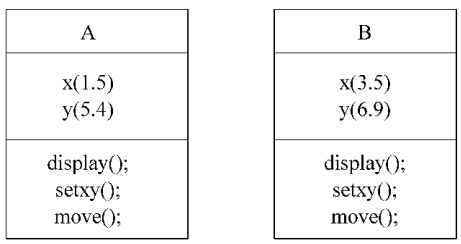


图 员源在摇具体对象 粤和 月的结构图

员源在员摇抽象和类

抽象是一种从一般的观点看待事物的方法，即集中于事物的本质特征，而不是具体细节或具体实现。面向对象的方法鼓励程序员以抽象的观点看待程序，即程序是由一组抽象的对象组成的。另外，又可以将一组对象的共同特征进一步抽象出来，从而形成“类”的概念。例如，从点 粤和 月抽象出点的概念，这就是“点类”。这个类的本质是具有两个坐标属性和对这两个属性值进行操作的方法。粤和 月的区别只是属性值的不同，其方法是相同的。例如，当它们使用移动位置的函数之后，点的位置属性值将随之变化。它们都从当前位置移到一个新位置，代表了新坐标处的点对象。通过分析一组对象（粤和 月），抽取公共的行为将其放入到一个类中，取名为 孕燥蚤灶贼类，并用图 员源在的 孕燥蚤灶贼类模型表示它。它也由类名、一组属性和一组操作等 猿部分组成。类的属性只是性质的说明，对象的属性才是具体的数据。所以，图 员源在的 孕燥蚤灶贼类只是表示类名是 孕燥蚤灶贼，它的点位置是两个实数，但还没有具体的位置。只有像本节 粤和 月两点那样，具有确定的属性值，才是 孕燥蚤灶贼类的具体对象，称它为类 孕燥蚤灶贼的一个实例。从图 员源在可以推出图 员源远所示的一般的“类”模型结构图。

抽象出这个类 孕燥蚤灶贼就可以集中精力研究有关“点”的概念了，如果再给它增加一个

摇摇摇摇

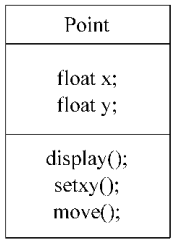


图 员源源摇类 孕遭遭的结构图

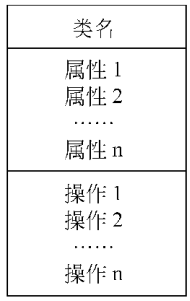


图 员源源摇类模型的结构图

颜色属性,就可以知道这个类的所有对象都可以有颜色上的区别。无须再将注意力分散到有关任何一个具体点的坐标和颜色等细节中去。又如大家非常熟悉的整数类,通过研究发现,它的属性是一个整数集合,它提供的基本操作是四则运算。愿和 怨是两个整数对象,而 愿垣怨是使用整数类提供的相加求和操作。由此可见,类的概念来自于人们认识自然、认识社会的过程。在这一过程中,人们主要使用由特殊到一般的归纳法和由一般到特殊的演绎法。在归纳过程中,是从一个个具体的事物中把共同的特征抽取出来,形成一个一般的概念,这就是“归类”;在演绎的过程中,又把同类的事物,根据不同的特征分成不同的小类,这又是“分类”。对于一个具体的类,它有许多具体的个体,这些个体叫做“对象”。举个例子,“人”是一个类,具有“直立行走、会使用工具”等一些区别于其他事物的共同特征;而张三、李四、王五等一个个具体的人,就是“人”这个类的一个个“对象”,同一类的不同对象具有相同的行为方式 (如张三和李四都能直立行走、会使用工具),不同类的对象具有不同的行为 (例如张三能使用工具,坐标点 粤则不能使用工具)。这种设计方法模仿了人们建立现实世界模型的方法,与人们认识客观事物的过程相一致,提高了编写程序的质量和可靠性。

由此可见,一个对象是由一些属性和操作构成的。对象的属性和操作描述了对象的内部细节。类是具有相同的属性和操作的一组对象集合,它为属于该类的全部对象提供了统一的抽象描述,其内部包括属性和操作两个主要部分。这两个部分也是对象分类的依据,只有给出对象的属性和操作,才算对这个对象有了确切的认识和定义。

类的作用是定义对象。类和对象的关系如同一个模具与用这个模具铸造出来的铸造件之间的关系。类给出了属于该类的全部对象的抽象定义,而对象则是符合这种定义的实体。所以,悦++ 中将对象称作类的一个实例。在程序中,每个对象需要有自己的存储空间以保存它们自己的属性值。所谓“一个类的所有对象具有相同的属性”是指属性的个数、名称、数据类型相同,各个对象的属性值则可以互不相同,并且随着程序的执行而变化。至于操作,则是所有对象共同使用它们的类定义中给出的操作代码,在 悦++ 中,每个操作是一个“成员函数”,在 [illegible] 中则称为“方法”。这里说同类对象具有相同的属性和操作,是指它们的定义形式相同,而不是说每个对象的属性值都相同。

员源源摇封装

举个简单的例子,电视机把各种部件都装在机箱里,遥控功能装在遥控器盒子中。用