第1章

概　　述

【知识结构图】

第1章知识结构参见图1-1。

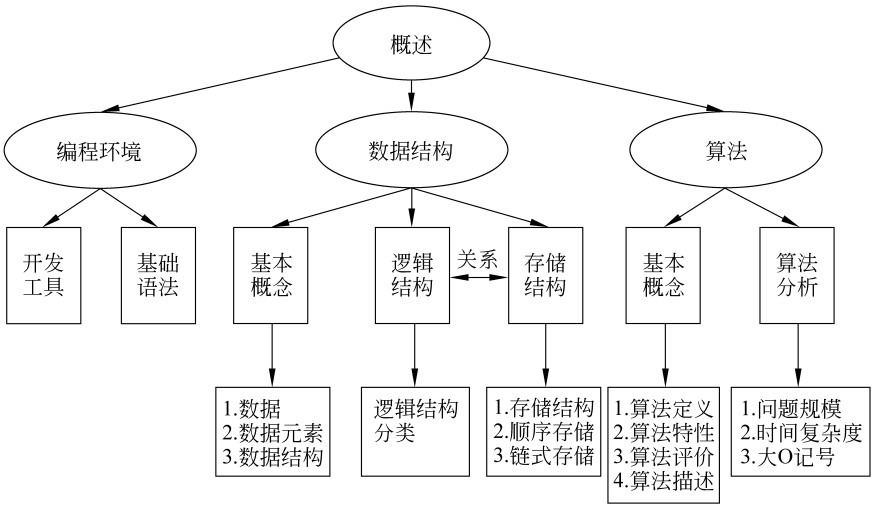


图1-1　知识结构图

【学习要点】

本章知识点包括三条主线：①编程环境，包括高级语言基础语法；②数据结构，包括相关基本概念及含义；③算法，包括算法的基本概念、描述方法及时间复杂度的分析方法。

学习数据结构主要抓住两个方面：逻辑结构和存储结构，需重点把握两者之间的关系。学习算法以基本概念和特性为基本点，重点是算法效率分析。

1.1　编程环境

1.1.1　Java编程环境

Java的开发工具很多，而且各有优缺点。目前，比较流行的Java开发工具有EditPlus、Jcreator、Eclipse、MyEclipse、Jbuilder、NetBeans等，本书使用的开发工具为MyEclipse。

MyEclipse是一个非常优秀的用于开发Java、J2EE的Eclipse插件集合，功能强大，支持的开发工具十分广泛，例如Java Servlet、AJAX、JSP、JSF、Struts、Spring、Hibernate、

EJB3、JDBC 数据库链接工具等,几乎囊括了目前所有的主流开源产品。图 1-2 为 MyEclipse 2013 的编辑界面。

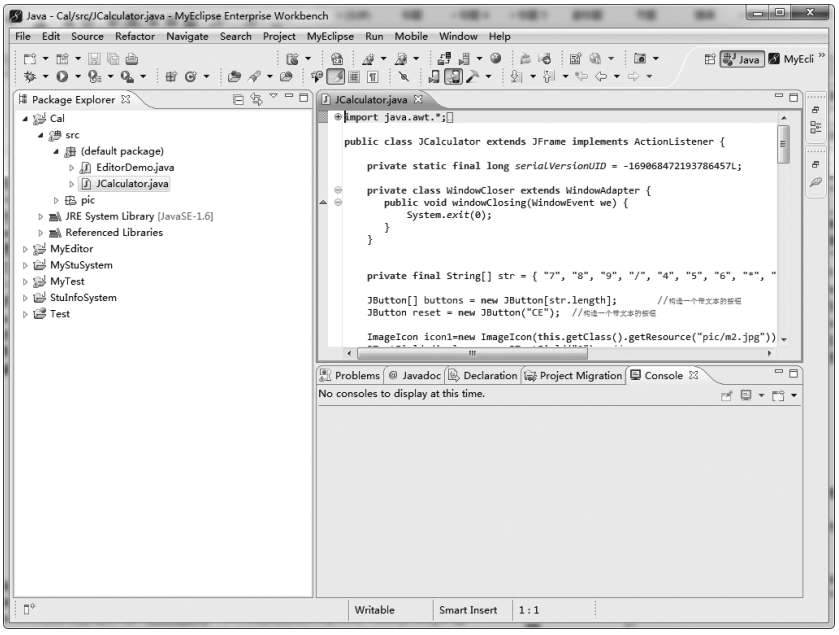


图 1-2　MyEclipse 2013 的编辑界面

在进行实验时,启动 MyEclipse 后要进行以下几步准备工作。

(1) 新建 Java Project,如图 1-3 所示,输入 Project name。

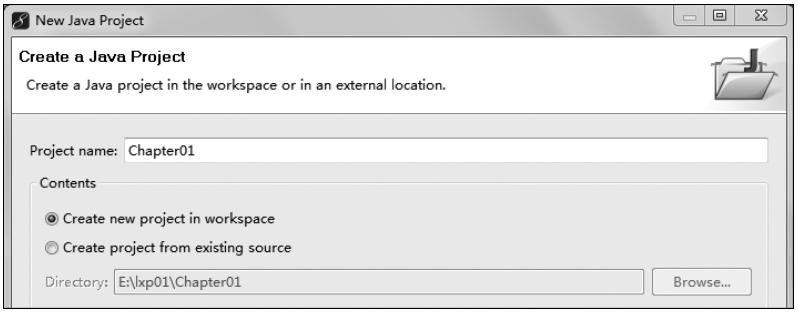


图 1-3　新建 Java Project

【本书命令规则】 第 1 章实验项目名命名为 Chapter01,第 2 章实验项目名命名为 Chapter02,以此类推。

(2) 新建 Java Package,如图 1-4 所示,输入 Package name。

【本书命令规则】 第 1 章第一个实验包命名为 com.gdlg.sy1_1,第二个实验包命名为 com.gdlg.sy1_2;第 2 章第一个实验包命名为 com.gdlg.sy2_1,第二个实验包命名为 com.gdlg.sy2_2,以此类推。

(3) 新建 Java Interface,如图 1-5 所示;或新建 Java Class,如图 1-6 所示。

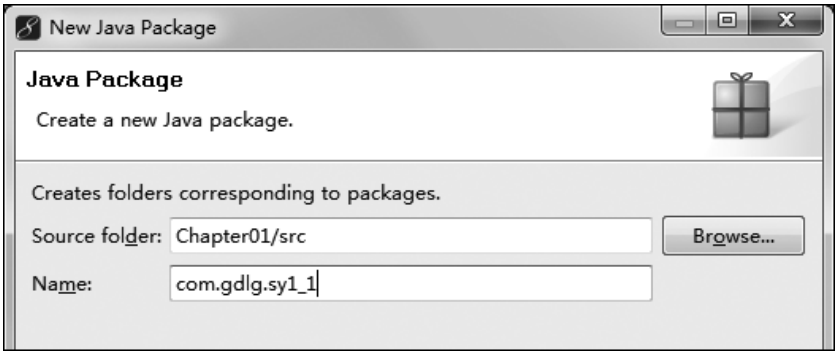


图 1-4　新建 Java Package

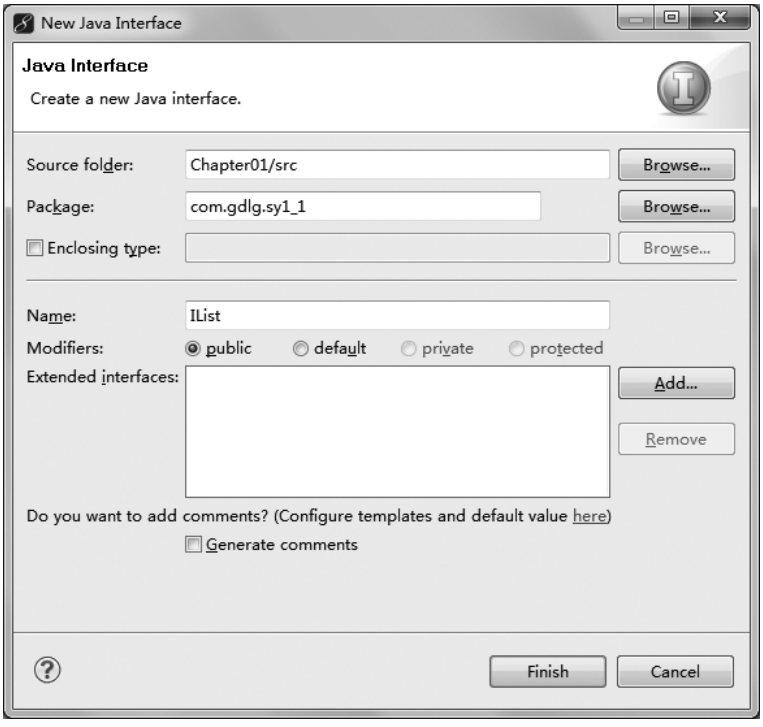


图 1-5　新建 Java Interface

1.1.2　C++ 编程环境

C++ 的开发工具很多，而且各有优缺点。目前，比较流行的 C++ 开发工具有 kDevelop、Anjuta Devstudio、code block、Visual-MigGW、Ideone、Eclipse CDT、Compiler、Code lite、Netbeans C++、Dev C++、Ultimate++ 、DigitalMars、C-Free、Tiny C Compiler、Visual Studio 等。本书使用的开发工具为 Visual Studio 2015。

Visual Studio 2015 是一款由开发人员工作效率工具、云服务和扩展组成的集成套件，适用于 Web、Windows 商店、桌面、Android 和 iOS 的应用程序和游戏。图 1-7 为 Visual Studio 2015 的新建项目界面。

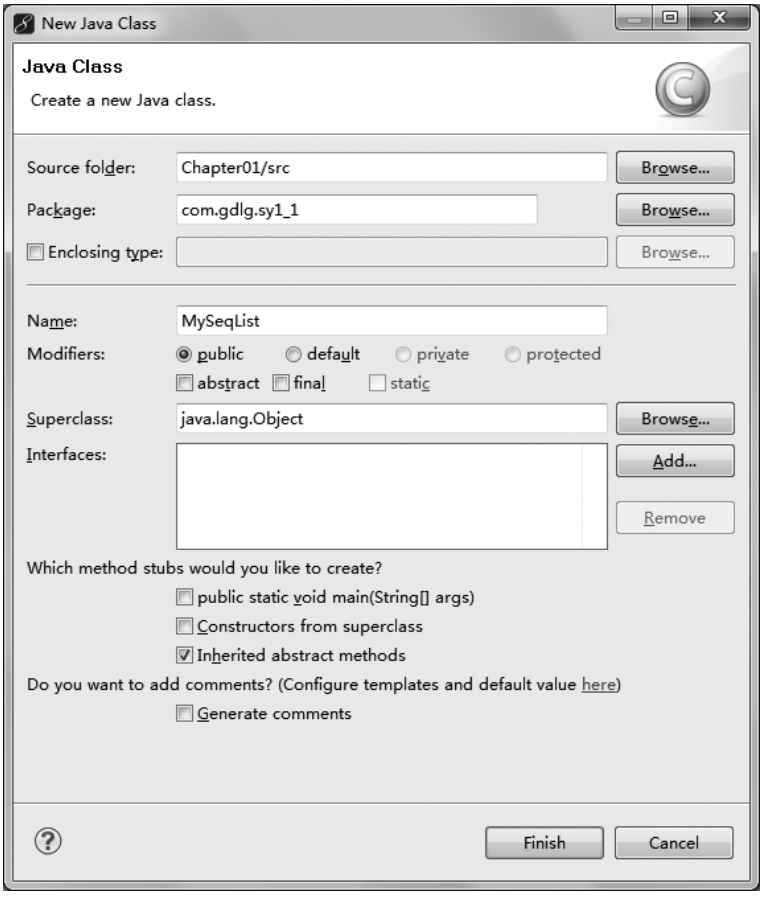


图 1-6　新建 Java Class

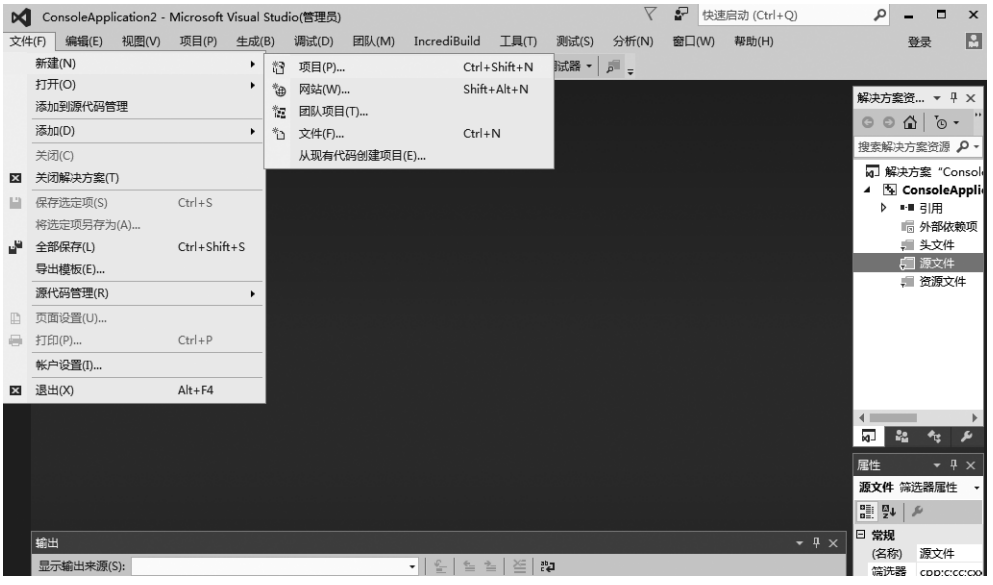


图 1-7　新建项目

在进行 C++ 实验时，启动 Visual Studio 2015（以下简称 VS2015）后，可以通过建立 Win32 控制台应用程序来调试 C++ 程序。

（1）单击 VS2015 的“文件”→“新建”→“项目”，如图 1-7 所示。

（2）选择 Visual C++ →“Win32 控制台应用程序”，输入相应的应用程序名称和应用程序位置，单击“确定”按钮，如图 1-8 所示。

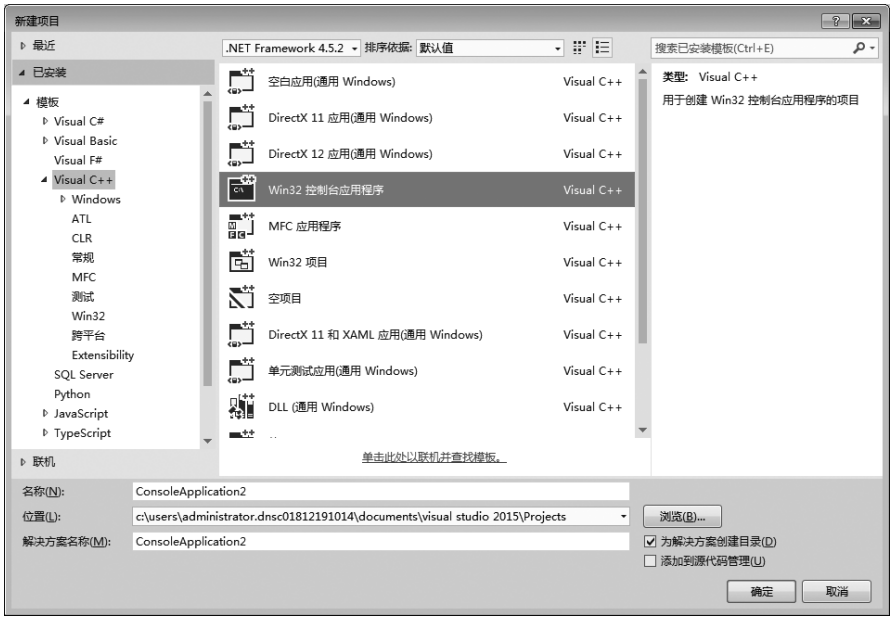


图 1-8　选择 Win32 控制台应用程序

（3）进入程序向导，单击“下一步”按钮，如图 1-9 所示。

图 1-9　应用程序向导

(4) 选择控制台应用程序和空项目,如图 1-10 所示。

图 1-10　选择应用程序类型

(5) 在完成上面的步骤后,在右边的“解决方案管理器”中选择“源文件”,右击选择“添加”→“新建项”,如图 1-11 所示。

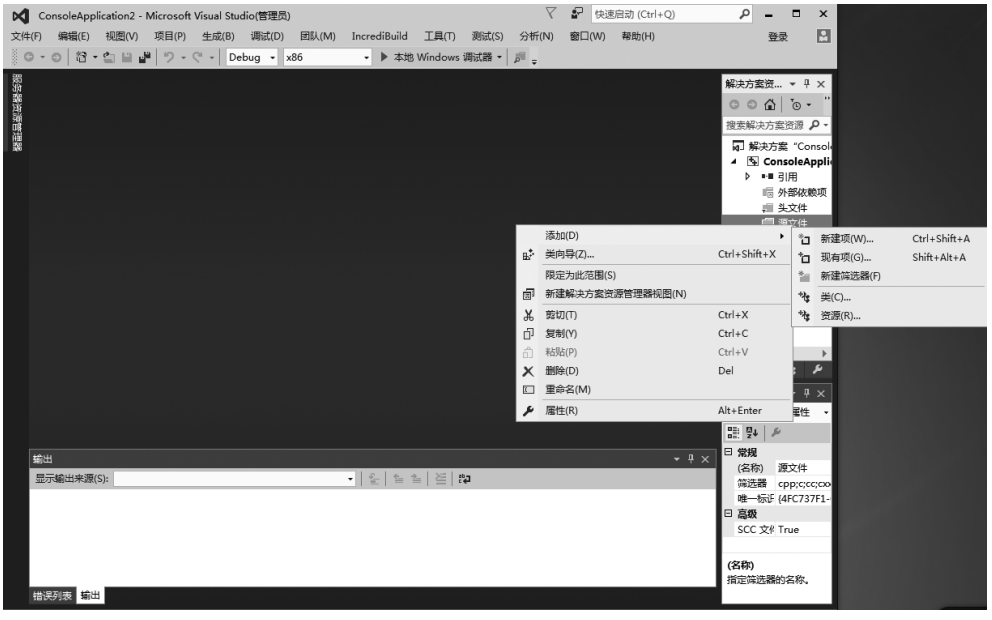


图 1-11　新建源文件

(6) 选择“C++ 文件(.cpp)”,单击“添加”按钮,如图 1-12 所示。

图 1-12　新建 C++ 文件

(7) 打开应用程序开发界面,可以编写代码,如图 1-13 所示。

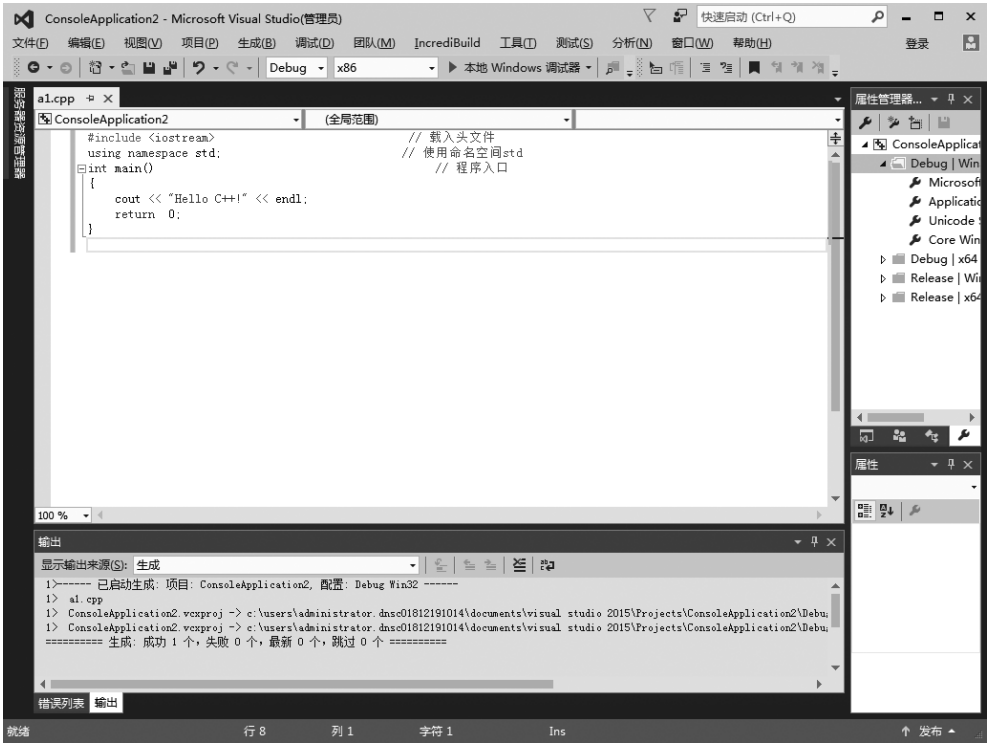


图 1-13　打开应用程序开发界面

(8) 编写好代码后,单击调试按钮或者按 Ctrl+F5 组合键进行调试。

1.2 数据结构

数据结构是一门研究非数值计算的程序设计问题的操作对象,以及它们之间的关系和操作等相关问题的学科。1968 年美国教授高德纳(Donald E.Knuth)在《计算机程序设计艺术》第一卷《基本算法》中系统地阐述了数据的逻辑结构和存储结构及其操作,开创了数据结构课程体系。同年,数据结构作为一门独立的课程,在计算机科学的课程体系中开始出现。数据结构对于程序设计来说非常重要,占据重要地位。

程序设计=数据结构+算法

1.2.1 基本概念

1. 数据

数据是所有能被输入计算机中,且能被计算机处理的符号的总称,如实数、整数、字符(串)、图形和声音等。数据是计算机操作对象的集合。数据不仅包括整型、浮点型这些数值型数据,还包括声音、视频、图像等多媒体数据。

2. 数据元素

数据元素是数据(集合)中的一个“个体”,是数据结构中讨论的基本单位,不同场合也叫结点、顶点、记录。比如,在人类中数据元素是人,禽类的数据元素有猪、牛、马、羊等。

3. 数据项

数据项是数据结构中讨论的最小单位,不可再分割,一个数据元素由若干个数据项组成。比如,人这个数据元素,可以包括眼、鼻、嘴、耳等数据项,也可以包括姓名、年龄、性别等数据项。

4. 数据对象

数据对象是性质相同的数据元素的集合,是数据的子集。在某些场合,将数据对象简称为数据。

如表 1-1 所示,学生成绩表是一个数据对象,表中的一行称为数据元素,表中的一列称为数据项。

表 1-1 学生成绩表

学号	姓名	高等数学	计算机导论	大学英语
2019001	张三	90	56	89
2019002	李四	80	87	67
2019003	丁一	67	67	87
2019004	马二	98	90	67
2019005	王五	56	87	68

5. 数据结构

数据结构是相互之间存在一种或多种关系的数据元素的集合。在计算机中,数据元素

之间存在的一种或多种特定关系，也就是数据的组织形式。按照视点的不同，把数据结构分为逻辑结构和存储结构，逻辑结构是面向问题的，而存储结构是面向计算机的。

数据结构可用二元组形式定义为：

```
Data_Structures=(D, R)
```

其中，D 是数据元素的有限集合，R 是 D 上关系的有限集合。

1.2.2 逻辑结构

数据的逻辑结构是指各个数据元素之间的逻辑关系，是呈现在用户面前的、能感知到的数据元素的组织形式。它与数据的存储无关，是独立于计算机的。按照数据元素之间的逻辑关系可将数据结构划分为线性结构和非线性结构两大类，非线性结构又分为树结构、图结构和集合结构，如图 1-14 所示。

1. 线性结构

线性结构中数据元素之间存在“一对一”的关系，即如果结构非空，则有且仅有一个首结点和尾结点，首结点没有前驱但有一个后继结点，尾结点没有后继但有一个前驱结点，其余结点有且仅有一个前驱和一个后继结点，如图 1-15 所示。

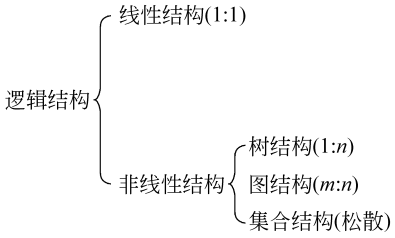


图 1-14　逻辑结构

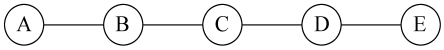
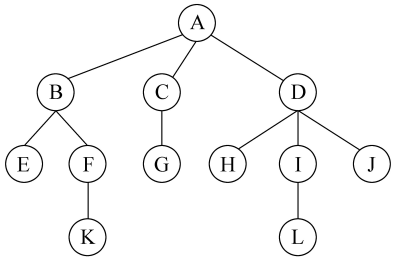


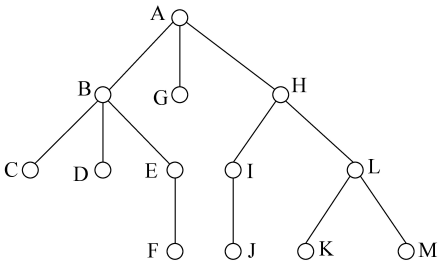
图 1-15　线性结构

2. 树结构

树结构中数据元素之间存在“一对多”的关系，即如果结构非空，则有一个根结点，该结点没有前驱结点，其余结点有且仅有一个前驱，所有结点都可以有多个后继结点，如图 1-16 所示。



(a) 一棵树



(b) 另一棵树

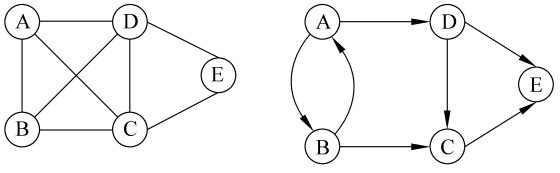
图 1-16　树结构

3. 图结构

图结构中数据元素之间存在“多对多”的关系,即如果结构非空,则任何结点都可以有任意个前驱和后继,如图 1-17 所示。

4. 集合结构

集合结构中数据元素之间是一种松散的关系,元素之间除了“同属于一个集合”之外无其他关系,如图 1-18 所示。



(a) 无向图　　(b) 有向图

图 1-17　图结构

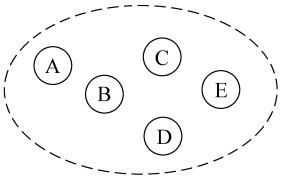


图 1-18　集合结构

1.2.3　存储结构

存储结构也称物理结构,是逻辑结构在计算机中的表示,即逻辑结构在存储器中的实现,依赖于计算机。基本的存储结构有顺序存储结构和链式存储结构两种。另外,还有索引存储结构和哈希存储结构两种。

1. 顺序存储结构

顺序存储结构以相对的存储位置来表示数据元素之间的逻辑关系,逻辑位置关系与存储位置关系是一致的。如图 1-19 所示,a_1 是 a_0 的后继结点,是 a_2 的前驱结点。

图 1-19　顺序存储结构

这种结构的主要特点:所有元素存放在一片连续的存储单元中,整个存储结构中只含数据元素值本身的信息。

在程序设计语言中,数组就是这样的存储结构。例如,当要建立一个长度为 6 的整型数组时,计算机就在按照一个整型所占的空间大小乘以 6,在内存中开辟一段空的连续的地址空间,第一个数据元素存入第一个位置,第二个元素存入第二个位置,依次存放所有的元素。

2. 链式存储结构

链式存储结构以附加信息(指针)表示后继关系,即每个结点除了存储数据元素值本身之外,还需存储表示逻辑关系的指针。这种结构的主要特点:所有元素存放在可以不连续的存储单元中,逻辑上相邻的元素其存储位置不一定相邻,如图 1-20 所示,其中 h 表示头指针。

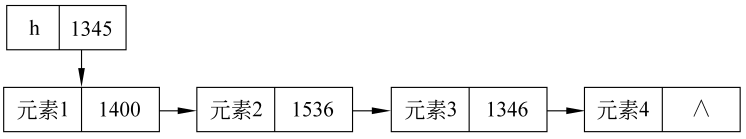


图 1-20　链式存储结构