



普通高等教育“十三五”规划教材

数据结构实践教程 (C语言版)

主编 袁嵩



华中科技大学出版社
<http://www.hustp.com>

离散数学基础
办公自动化高级教程
Office高级应用案例教程
计算机基础教程
计算机英语实用教程
Visual Basic程序设计教程
Visual FoxPro 8.0程序设计基础
C语言程序设计
C语言程序设计实验指导与习题选解
C语言程序设计案例教程
C++程序设计案例教程
C++程序设计精讲与实训
Visual C++程序设计教程
Visual C++知识点详解与算法程序精练
C#程序设计案例教程
C#程序设计基础
JAVA程序设计
Java基础案例教程
计算机网络
计算机网络技术
计算机组成原理教程
计算机组成原理实验指导教程
数据结构
数据结构（C语言版）
数据结构（C语言版）——从应用到实现
✔ 数据结构实践教程（C语言版）
数据结构实验指导与习题解析（C语言版）
微机原理与接口技术

微型计算机原理与接口技术
Access数据库基础与案例教程
SQL Server数据库原理及应用
SQL Server 2016数据库应用教程
MySQL数据库实用教程
数据库原理与应用
数据库原理与应用实践教程
数据库原理与应用（SQL Server 2012）
Oracle 11g数据库教程
软件工程
软件建模技术与应用
Photoshop平面设计实用教程
Linux操作系统基础及应用
Android应用开发案例教程
PHP动态网页设计与网站架设
Java EE应用开发教程
ASP.NET Web应用开发教程
Web前端开发技术
IT项目管理
机器人编程设计与实现
物联网技术导论
物联网RFID技术与应用
人工智能基础教程
大数据导论
云计算应用与实践——基于OpenStack架构
IDC基础运维



策划编辑：康 序

责任编辑：史永霞

ISBN 978-7-5680-5858-2



9 787568 058582 >

定价：38.00元



普通高等教育“十三五”规划教材

数据结构实践教程 (C语言版)

主 编 袁 嵩

副主编 马庆槐 熊 莹 徐 嘉



华中科技大学出版社

<http://www.hustp.com>

中国·武汉

此为试读, 需要完整PDF请访问: www.ertongbook.com

内 容 简 介

本书是《数据结构(C语言版)》配套的实验指导和习题集,本书内的所有语言均采用C/C++语言,所有应用程序均在Microsoft Visual Studio 2010集成开发环境下编译并通过。

本书结合企业常用的实际案例、应用环境和软件开发技术,对线性表、队列、二叉树、图、查找等多个重要的数据结构知识单元进行了分析和设计,并为每个知识单元设计了实验指导和丰富的习题,力求使学生充分理解这些知识并通过实验指导、习题和综合实践来验证所学的知识。全书分为三大部分:第一部分为实验指导,共安排了六个实验项目,可对应课堂练习或实验教学,重点训练每一知识单元的编程实现技巧;第二部分为习题,可对应课堂练习或课外作业,进一步巩固和检验各单元知识点的掌握情况;第三部分为综合实践,可对应课程设计,该部分利用一个游戏项目综合训练数据结构核心知识和算法在企业实际项目中的应用。

本书适合各类高等院校计算机专业学生作为数据结构实践教学参考用书,也适合作为对数据结构与算法应用开发感兴趣的学习者的指导用书或参考书。

图书在版编目(CIP)数据

数据结构实践教程:C语言版/袁嵩主编. —武汉:华中科技大学出版社,2019.10
普通高等教育“十三五”规划教材
ISBN 978-7-5680-5858-2

I. ①数… II. ①袁… III. ①数据结构-高等学校-教材 ②C语言-程序设计-高等学校-教材
IV. ①TP311.12 ②TP312.8

中国版本图书馆CIP数据核字(2019)第234045号

数据结构实践教程(C语言版)

Shuju Jiegou Shijian Jiaocheng(C Yuyan Ban)

袁嵩 主编

策划编辑:康序

责任编辑:史永霞

封面设计:孢子

责任监印:朱玢

出版发行:华中科技大学出版社(中国·武汉)

电话:(027)81321913

武汉市东湖新技术开发区华工科技园

邮编:430223

录排:武汉三月禾文化传播有限公司

印刷:武汉华工鑫宏印务有限公司

开本:787mm×1092mm 1/16

印张:16

字数:410千字

版次:2019年10月第1版第1次印刷

定价:38.00元



华中科大

本书若有印装质量问题,请向出版社营销中心调换
全国免费服务热线:400-6679-118 竭诚为您服务
版权所有 侵权必究

前言

PREFACE

数据结构是计算机专业的必修主干课程之一,是一门实践性很强的课程。为了满足计算机类各专业学生对数据结构课程的上机、知识巩固和实践指导需求,特编写了本书。本书旨在方便学生通过大量的实验和习题的练习,充分掌握数据结构的基本知识,并培养学生运用所学理论来分析和解决实际问题的能力以及严谨、求实的编程作风。为了帮助学生学会如何把现实世界的问题转化为计算机内部的表示和处理,我们为几种最常用的数据结构和算法精心设计了实验项目。这些实验项目全部按企业项目开发思路进行分析、设计和编程实施。同时,针对教材中典型知识点,我们还编写了丰富的配套习题供学生进行知识点检验和训练。除此之外,本书还设计了1个游戏案例,并且结合C++、MFC Dialog 可视化界面和GDI绘图等实用开发技术,提高数据结构和算法应用实践能力。在实践中,引导读者理解数据结构和算法中知识单元与项目需求如何进行技术对接,并同时采用迭代开发思路进行每一个功能开发。

全书主要分为实验指导、习题和综合实践三部分。

实验指导部分:针对开发工具、线性表、队列、树、图、查找设计了实验项目,以软件开发的形式加以呈现,包括实验目标、实验任务和实验实施等内容。

习题部分:针对重点理论知识,编排了练习题,包含单项选择题、填空题、判断题、综合题以及重点章节的算法设计题,并配有参考答案,供学生检验各知识点的掌握情况。

综合实践部分:设计了1套“连连看游戏”实践案例,通过完整实践数据结构和算法核心知识,阐释了数据结构在企业项目中的应用,同时详细讲解实践项目迭代开发过程。

本书由武汉科技大学袁嵩担任主编,由武汉市软酷网络科技有限公司马庆槐、武汉科技大学熊莹、武汉晴川学院徐嘉担任副主编,全书由袁嵩统稿,熊莹进行了审核。在本书编写过程中,软酷网(www.ruanko.com)为本书提供实践参考资料,其中,马庆槐总监为

我们提供了项目资源和企业项目实施过程资料,郑婕和王博宜项目经理负责项目的开发和测试,并完成部分图表绘制及文档排版工作,在此对他们表示衷心感谢。同时也特别感谢在本书出版过程中给我们支持与帮助的华中科技大学出版社的相关编辑和工作人员。

由于编者的水平和时间有限,本书难免出现错误,对于本书的任何问题,恳请读者批评指正。

编者

2019年6月

目录

CONTENTS

概述/1

7 查找/132

8 排序/138

9 参考答案/143

第一部分 实验指导

第三部分 综合实践(连连看游戏)

1 Visual Studio 开发工具/4

1 实践目标/189

2 线性表(一元多项式相加)/7

2 需求分析/189

3 队列(银行叫号系统)/18

3 系统设计/191

4 树和二叉树(随机地图生成器)/32

4 创建工程/195

5 图(公交线路图)/56

5 主界面/198

6 查找(字符统计程序)/71

6 定义游戏数据/202

7 实验报告要求/86

7 绘制游戏地图/206

第二部分 习题

8 一条直线消子/215

1 绪论/87

9 两条直线消子/231

2 线性表/91

10 三条直线消子/235

3 栈和队列/100

11 判断胜负/240

4 数组和广义表/104

12 重排/242

5 树和二叉树/108

13 提示/246

6 图/121

概 述

介绍

本书根据《数据结构(C语言版)》的内容,围绕数据结构和算法课程知识,拟通过实验指导、习题和综合实践(连连看游戏),综合训练学生对数据结构和算法的应用,培养学生运用所学理论知识分析和解决实际问题的能力,树立严谨、求实的工作作风,并达到以下目标:

- 理解 Microsoft Visual Studio 2010 工具,应用 C/C++ 编写应用程序。
- 掌握线性表、队列、树和二叉树、图、查找等重要数据结构知识的特点、存储表示、运算方法以及在计算机学科中的基本应用。
- 理解企业软件开发过程,理解系统需求分析和设计,应用迭代开发思路进行实践。
- 养成良好的编码习惯和培养软件工程化思维,综合应用程序设计、数据结构和算法等知识,开发“连连看游戏”项目,达到掌握和应用数据结构核心知识的目的。

实施安排

本书的内容主要分为实验指导、习题和综合实践 3 大部分,先通过实验专题编程训练数据结构和算法中的知识单元内容,再通过习题巩固知识点,最后通过综合实践,训练运用各知识单元内容综合解决企业应用项目的能力。

实验指导采用控制台应用程序,聚焦数据结构中的知识单元训练;综合实践部分采用桌面应用程序,聚焦整个数据结构核心知识训练。

1. 实验指导部分

选取典型的开发工具、线性表、队列、树和二叉树、图、查找等内容,安排 6 个技术专题,分别训练相关知识,如表 0-1 所示。

表 0-1

编号	实验	技术专题	训练知识点
1	开发工具	第一个 C++ 程序	1. 安装 Visual Studio 2010 2. 编写标准 C++ 程序
2	线性表	一元多项式相加	1. 线性表的链式存储结构 2. 线性表的基本操作
3	队列	银行叫号系统	1. 队列的定义 2. 队列的链式存储结构 3. 队列的基本操作
4	树和二叉树	随机地图生成器	1. 树和二叉树的定义 2. 二叉树的链式存储结构 3. 二叉树的遍历和基本操作

续表

编号	实验	技术专题	训练知识点
5	图	公交线路图	1. 图的定义 2. 图的邻接表存储结构 3. 深度优先搜索算法
6	查找	字符统计程序	1. 查找定义 2. 二叉排序树

2. 习题部分

针对数据结构课程的重难点,配套了 8 章内容(见表 0-2)的习题,题型包括单项选择题、填空题、判断题、综合题及算法设计题,并配有参考答案。

表 0-2

编号	章节	知识点
1	绪论	数据结构的逻辑结构和存储结构、算法和复杂度分析
2	线性表	顺序表、单链表、循环链表、双向链表的表示和实现
3	栈和队列	顺序栈、循环队列、链队列的表示和应用
4	数组和广义表	数组的顺序表示、矩阵的压缩存储、广义表的定义
5	树和二叉树	树与二叉树的定义和实现,二叉树的遍历,树、森林与二叉树的关系,哈夫曼树及其应用
6	图	图的数据类型定义,图的实现(邻接矩阵、邻接表),图的遍历,图的应用(最小生成树、拓扑排序、关键路径、最短路径)
7	查找	顺序查找、折半查找、二叉排序树、哈希表构造、处理冲突的方法和分析
8	排序	直接插入排序、希尔排序、快速排序、选择排序、堆排序、归并排序、基数排序

3. 综合实践部分

应用迭代增量思想,进行功能迭代开发,迭代根据功能和训练知识进行划分,每一个迭代内容包括工作任务和编码实现。“连连看游戏”迭代内容如表 0-3 所示。

表 0-3

编号	功能/迭代	实践内容	训练知识点
1	创建工程	搭建开发环境,创建项目解决方案和工程	1. Microsoft Visual Studio 2010 开发工具 2. MFC Dialog 程序的创建
2	主界面	创建主界面,添加按钮控件和菜单	1. 窗体属性 2. 按钮控件 3. 菜单

续表

编号	功能/迭代	实践内容	训练知识点
3	定义游戏数据	设计和定义游戏地图、元素图片和游戏区域的存储结构	1. 数组 2. 结构体 3. 类向导
4	绘制游戏地图	单击“开始游戏”时,使用多张图片,按照固定的顺序产生游戏地图,游戏地图为10行16列	1. 消息响应函数 2. GDI绘图 3. 透明位图的绘制 4. BitBlt()和掩码
5	消子判断	判断单击的两个图片是否满足一条、两条和三条直线消子,并绘制提示框和提示线	1. 鼠标单击事件 2. 消子算法 3. 界面和地图更新
6	判断胜负	判断棋盘子是否消完,并提示用户获胜,更新数据	胜负判断算法
7	重排	对游戏地图中剩下的图片进行重新排列,并利用重排实现随机开局	1. 重排算法 2. 界面和地图更新
8	提示	系统提示可以消除的一对图片,在该对图片间绘制连接线	1. 提示算法 2. 界面和地图更新

第一部分 实验指导

1 Visual Studio 开发工具

1.1 实验目标

- (1) 掌握 Microsoft Visual Studio 2010 工具的使用。
- (2) 掌握 C++ 程序的组成。
- (3) 掌握 C++ 程序中 cin 和 cout 的用法。
- (4) 了解程序的编辑、编译、连接和执行的过程。
- (5) 综合运用以上知识,开发 C++ 程序。

1.2 实验任务

安装和使用 Microsoft Visual Studio 2010 开发工具,编写控制台程序。功能是根据用户输入内容,输出对应的内容。具体功能如下:

- (1) 提示用户输入文字“Please input some words:”,如图 1-1 所示。
- (2) 用户在控制台下输入一段字符数据,例如“HelloWorld”,如图 1-2 所示。
- (3) 按回车键将输出“Output words:HelloWorld”,如图 1-3 所示。

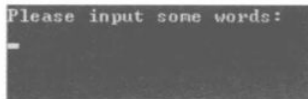


图 1-1

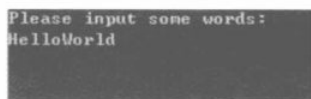


图 1-2

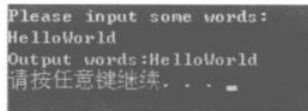


图 1-3

1.3 实验实施

使用 Microsoft Visual Studio 2010 开发工具创建一个 Win32 控制台程序,通过输入输出流实现数据的输入和输出功能,工程名为 HelloWorld。

(1) 利用 Microsoft Visual Studio 2010 创建一个空白的解决方案,解决方案名为 HelloWorldCPro。

(2) 在解决方案中,利用 Microsoft Visual Studio 2010 工具的 Win32 应用程序向导,创建一个空的 Win32 控制台工程,工程名为 HelloWorld。

(3) 在 HelloWorld 工程中添加一个 Main.cpp 文件,在该文件中添加程序的入口函数 main()。

(4) 在 main()函数中使用 cin 接收控制台输入内容,cout 向控制台输出内容。

步骤一:下载和安装

(1) 下载 Microsoft Visual Studio 2010 安装包。

下载地址为 <http://www.microsoft.com/visualstudio/zh-cn/download>。

(2) 安装 Microsoft Visual Studio 2010。

打开安装包即可进行安装。在选择安装模式时,可以选择全部安装,也可以选择自定义安装。在自定义安装时,注意要选 Visual C++。

关于安装的路径,最好选择默认的安装路径。

步骤二:创建解决方案

(1) 启动 Microsoft Visual Studio 2010,如图 1-4 所示。

(2) 选择左上角的“File”菜单,单击“New Project”,弹出新建对话框。

(3) 选择工程类型为“Other Project Types -> Visual Studio Solutions -> Blank Solution”,输入工程名称 HelloWorldCPro,选择工程保存路径,单击“OK”按钮,完成解决方案的创建,如图 1-5 所示。



图 1-4

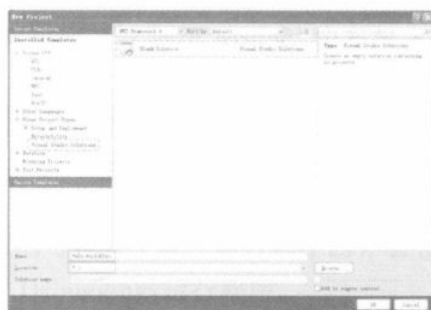


图 1-5

步骤三:创建工程

(1) 选择左上角的“File”菜单,单击“New Project”,弹出新建对话框。

(2) 选择工程类型为 Visual C++ -> Win32 -> Win32 Console Application,输入工程名 HelloWorld,选择 Add to solution,如图 1-6 所示。



图 1-6

(3) 单击“OK”按钮,进入 Win32 应用程序向导。选择应用程序类型为 Console application, Additional options 属性选择 Empty project, 单击“Finish”按钮,完成工程的创建,如图 1-7 所示。

(4) 完成创建后,界面如图 1-8 所示。



图 1-7

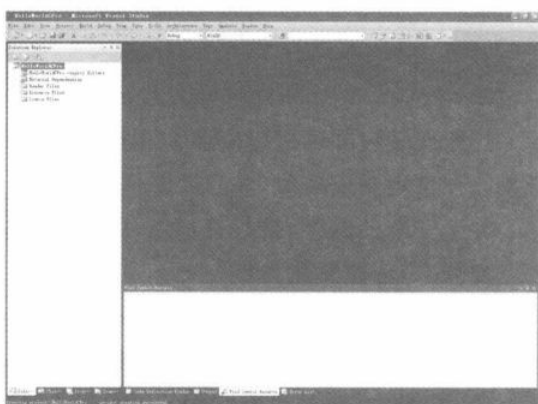


图 1-8

步骤四:添加文件

(1) 右键选中“Source Files”,在弹出菜单中选择“Add”→“New Item”,如图 1-9 所示,打开新建文件对话框。

(2) 在新建文件对话框中,选择文件类型为 C++ File(.cpp),输入文件名为 Main,选择文件保存路径为工程文件夹,如图 1-10 所示。



图 1-9

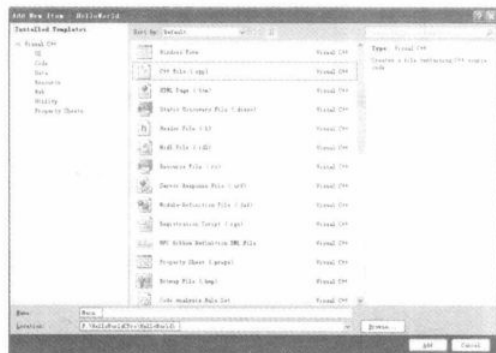


图 1-10

(3) 单击“Add”按钮,完成文件的添加,如图 1-11 所示。

步骤五:编辑程序

(1) 在 Main.cpp 文件中,添加入口函数 main() 函数。

```
int main(int argc, char*argv[])
{
    return 0;
}
```

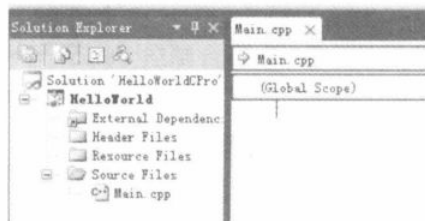


图 1-11

(2) 导入标准输入输出流头文件 `iostream`, 使用 `std` 命名空间。

```
#include <iostream>
using namespace std;
```

(3) 编写 `main()` 函数。

① 定义字符数组:

```
char ch[20];
```

② 输出提示信息到控制台:

```
cout << "Please input some words:" << endl;
```

③ 接收输入的数据:

```
cin >> ch;
```

④ 输出用户输入的数据并回车:

```
cout << "output words:" << ch << endl;
```

步骤六: 编译和运行程序

1) 编译程序

Microsoft Visual Studio 2010 将编译和连接的功能放到一起, 直接编译就可以产生可执行文件。

利用 Microsoft Visual Studio 2010 工具, 有两种方法可以对源文件进行编译连接。

方法一: 单击菜单 `Build` → `Build Solution`, 完成编译。

方法二: 使用快捷键 `F7`, 完成编译。

编译程序, 在解决方案目录和工程目录下面都会生成一个 `Debug` 文件, 在工程目录下的 `Debug` 文件夹中, 保存了编译程序时生成的一些文件, 比如目标文件, 如图 1-12 所示。

2) 运行程序

源文件经过编译连接后, 生成以 `.exe` 为后缀的可执行文件。运行程序的方法有两种:

方法一: 单击菜单 `Debug` → `Start Without Debugging`, 运行程序。

方法二: 使用快捷键 `Ctrl+F5`, 运行程序。

运行程序, 显示控制台界面, 输出提示信息, 如图 1-13 所示。

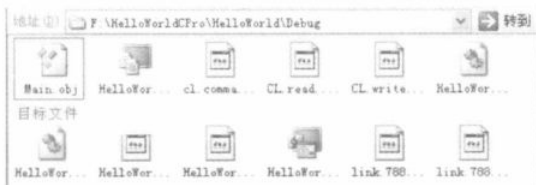


图 1-12



图 1-13

2 线性表(一元多项式相加)

2.1 实验目标

- (1) 掌握线性表的链式存储结构。
- (2) 掌握链表的基本操作, 并能进行应用实践。

(3) 使用 C/C++ 语言和线性表实现“一元多项式相加”专题。

2.2 实验任务

本程序是一个控制台程序,用户可以根据自己的需求分别输入两个一元多项式,并且能够实现显示输入的一元多项式,再将这两个一元多项式相加,输出结果。

本程序的输入需求是按照指数从小到大进行输入,并且项数必须为正整数,指数需为整数,系数为双精度型且不能为 0。

例如:

第一个多项式 $1.2 + 2.4x + 3.6x^3$,

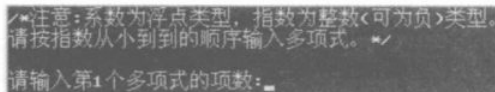
第二个多项式 $0.2x^{-1} + 1.2x^3$,

相加之后的结果为 $0.2x^{-1} + 1.2 + 2.4x + 4.8x^3$ 。

本程序具体实现的功能如下:

(1) 输出程序界面,提示用户输入第一个多项式的项数,如图 1-14 所示。

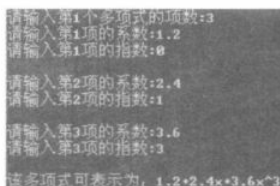
(2) 根据用户输入的项数,提示用户依次输入每一项的系数和指数,并打印出第一个多项式,如图 1-15 所示。



```

/*注意:系数为浮点类型,指数为整数(可为负)类型。
请按指数从小到大的顺序输入多项式。*/
请输入第1个多项式的项数:
  
```

图 1-14



```

请输入第1个多项式的项数:3
请输入第1项的系数:1.2
请输入第1项的指数:0

请输入第2项的系数:2.4
请输入第2项的指数:1

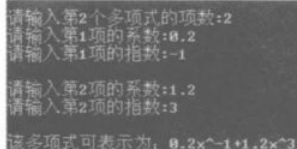
请输入第3项的系数:3.6
请输入第3项的指数:3

该多项式可表示为: 1.2+2.4x+3.6x^3
  
```

图 1-15

(3) 接着会提示输入第二个多项式的项数,按照上述方法输入第二个多项式的相关参数,打印该多项式,如图 1-16 所示。

(4) 输出最终相加的和多项式,如图 1-17 所示。



```

请输入第2个多项式的项数:2
请输入第1项的系数:0.2
请输入第1项的指数:-1

请输入第2项的系数:1.2
请输入第2项的指数:3

该多项式可表示为: 0.2x^-1+1.2x^3
  
```

图 1-16



```

多项式之和为: 0.2x^-1+1.2+2.4x+4.8x^3
请按任意键继续...
  
```

图 1-17

2.3 实验实施

设计思路

使用 Microsoft Visual Studio 2010 创建一个 Win32 Console Application 工程,利用结构体、链表等数据结构,使用 C++ 语言开发一元多项式相加,工程名为 PolynAddCPro。

1) 程序层次结构

(1) 在工程中添加一个 PolynAdd.cpp 文件作为程序的主文件。

(2) 在工程头文件中添加 StructDefine. h,用于存储用结构体定义链表的结点。

(3) 在工程头文件中添加一个 Polyn. h 文件,在源文件中添加一个 Polyn. cpp 文件,用于声明和实现一元多项式的相关函数。

程序层次结构如表 1-1 所示。

表 1-1

文件	主要函数	说明
PolynAdd. cpp	int main()	程序入口函数
	bool GetInt()	输入并验证输入是否为整数
	void InputData	输入一元多项式数据并排序输出
StructDefine. h		使用结构体定义链表的结点
Polyn. h、Polyn. cpp	void CreatePolyn()	创建空链表
	void ListInsert()	将结点插入链表尾部
	PLinkList ListSort()	将链表按指数从小到大排序
	void PrintPoly()	打印输出单个结点
	void PrintPolyn()	打印输出整个一元多项式
	PLinkList AddPolyn()	将两个一元多项式相加
	void FreePolyn()	释放链表

2) 数据设计

(1) 一元多项式的结点类型。

一个一元多项式可以写成下列形式：

$$P_n(X) = c_1 x^{e_1} + c_2 x^{e_2} + \dots + c_n x^{e_n}$$

若用一个长度为 n 且每个元素都有两个数据项(系数项和指数项)的线性表解决多项式相加,必须要有多项式,所以必须首先建立两个多项式。在这里采用链表存储一元多项式的数据,其每一个结点都代表一元多项式的一项,所以将结点结构体定义为:

序数 coef	指数 expn	指针域 next
---------	---------	----------

例如多项式为 $7 + 3x + 9x^8 + 5x^{17}$,则用链表可描述为图 1-18 所示。

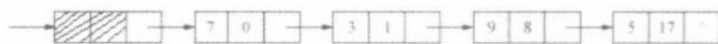


图 1-18

用结构体表示其结点为:

```
typedef struct PLNODE{
    double dbCoef;           //系数域
    int nExpn;              //指数域
    struct PLNODE*next;     //指针域,指向下一个结点
}PLNODE;
```

(2) 一元多项式的创建过程如图 1-19 所示。

(3) 两个一元多项式相加。

本程序的一元多项式是按照从小到大的顺序排列的。具体相加的描述如下：

假设指针 pa 和 pb 分别指向多项式 A 的第一项和多项式 B 的第一项,和多项式为 C。比较两个结点中的指数项的大小,在 pa 和 pb 的指向均不为空的时候,有三种情况：

① pa 所指结点的指数值 < pb 所指结点的指数值：

pa 的结点插入 pc 中,把 pa 指向下一个结点。

② pa 所指结点的指数值 = pb 所指结点的指数值：

```
if(这两个系数相加=0)
```

```
{//把 pa、pb 均指向下一个结点}
```

```
else
```

```
{//将它们系数相加的值和它们的指数赋给一个新的结点,插入 pc
```

```
//把 pa、pb 均指向下一个结点}
```

③ pa 所指结点的指数值 > pb 所指结点的指数值：

pb 的结点插入 pc 中,把 pb 指向下一个结点。

例如：

$$A = 5x^2 + 2x^3 + 4x^4 + 8x^6$$

$$B = 3x + 4x^2 + 6x^4$$

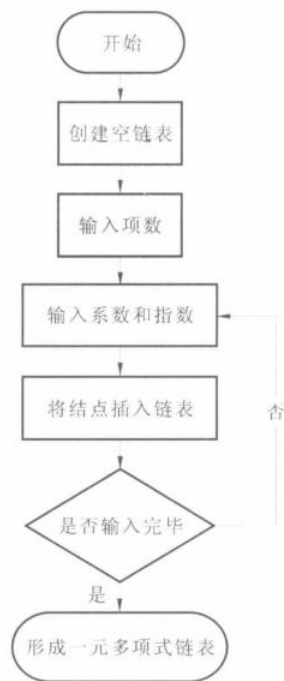


图 1-19

说明：

图 1-20 中红色箭头(①②③④⑤⑥)表示 A、B 当前所指结点指针 pa、pb,蓝色字体(打“√”)表示当前需要操作的结点数据。

这样遍历 pa、pb 两个一元多项式之后,有可能会有一个多项式没有遍历完,需要继续遍历,这个时候剩下的结点的指数值都比 pc 已有的结点的指数值要大,直接把剩余的结点从前到后一个一个插入 pc 尾部即可,如图 1-21 所示。

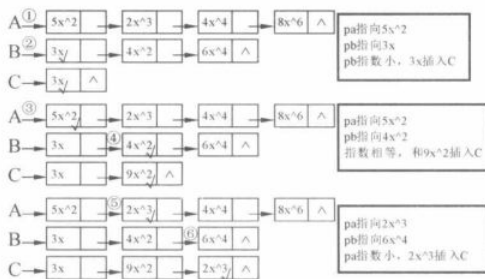


图 1-20

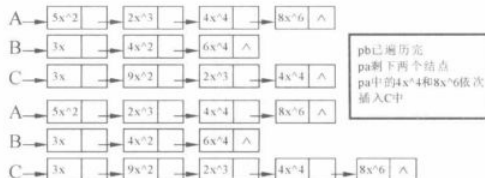


图 1-21

最后 $C = 3x + 9x^2 + 2x^3 + 4x^4 + 8x^6$ 。