数据结构算法

Visual C++ 6.0

程序集

侯识忠　等编著

包含108个完整的可执行程序及源代码

中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书收集了当前国内几本比较流行的数据结构算法书中的算法，在 Visual C++ 6.0 平台上将其编写成可以直接运行的程序。对有些算法，还给出了多种程序设计方案实现，以迅速提高程序员的算法与程序设计能力。

本书遵循数据结构算法书中的体系，将全书分为九章对应之，以便阅读。它们是：顺序存储结构的表、堆栈和队列，链式存储结构的表、堆栈和队列，数组、串和广义表，递归，树和二叉树，图，排序，查找，文件。本书的光盘中含有 108 个完整的程序源代码、可执行文件。

本书适合于想要通过编程实例学习数据结构的 C++程序员，也可以作为高校师生学习计算机编程与数据结构的参考书，尤其适合报考计算机软件专业研究生的人员与参加信息奥赛班学习的人员参考。

本书配有 PowerPoint 制作的电子教案，教师可以根据情况任意修改，如有需要，可从中国水利水电出版社网站（http://www.waterpub.com.cn/softdown/）下载。

图书在版编目（CIP）数据

数据结构算法：Visual C++ 6.0 程序集 / 侯识忠等编著．—北京：中国水利水电出版社，2005

ISBN 7-5084-2956-7

Ⅰ．数… Ⅱ．侯… Ⅲ．①数据结构②算法分析③C 语言－程序设计 Ⅳ．TP311.12

中国版本图书馆 CIP 数据核字（2005）第 050421 号

书　　名	**数据结构算法——Visual C++ 6.0 程序集**
作　　者	侯识忠　等编著
出版　发行	中国水利水电出版社（北京市三里河路 6 号　100044）
	网址：www.waterpub.com.cn
	E-mail：mchannel@263.net（万水）
	sales@waterpub.com.cn
	电话：（010）63202266（总机）、68331835（营销中心）、82562819（万水）
经　　售	全国各地新华书店和相关出版物销售网点
排　　版	北京万水电子信息有限公司
印　　刷	北京蓝空印刷厂
规　　格	787mm×1092mm　16 开本　23.75 印张　529 千字
版　　次	2005 年 5 月第 1 版　2005 年 5 月第 1 次印刷
印　　数	0001—5000 册
定　　价	45.00 元（含 1CD）

前　言

计算机专业和其他与计算机技术关系密切的许多专业课程都与数据结构这门课程有着紧密联系，这些课程的学习效果如何，与数据结构息息相关。

本人多年从事计算机语言课程的教学工作，如何让学生尽快学会、熟练掌握计算机编程，需要进行大量的工作，并提供一定的方法。仅仅让学生了解、掌握计算机语言是不够的，还必须掌握数据组织、存储和运算的一般方法，掌握一定的算法。然而，相当多的数据结构教材给出的算法一是抽象，二是不能直接用于上机操作，学生掌握起来有一定困难。

基于此，作者编写了这本《数据结构算法——Visual C++ 6.0 程序集》。编写本程序集的目的有二，一是让学习数据结构的读者更容易掌握其中的算法；二是让学习计算机语言编程的读者尽快学会、熟练掌握数据结构的相关算法，迅速提高编程能力。

本书对数据结构问题的描述具有相对独立性，每一个程序都完整地包含了该数据结构问题的各种算法。本书采用 C++语言描述具体的数据结构问题，所有程序都用 C++Builder 6.0 调试通过，并输出程序运行结果。

本书第 1 章至第 6 章、第 8 章由侯识忠同志编写，第 7 章、第 9 章由侯凌翀同志编写，杜正宽、杜希平、杜琨同志做了大部分程序校对与录入工作，最后由侯识忠同志统稿。

本书的 108 个完整的可执行程序、源代码都收录在随书光盘中，光盘中的应用程序除 5 个之外，其余的都可在光盘中直接执行，也可将光盘中的程序拷贝到硬盘 D 的子目录中运行。

本书适合于想要通过编程实例学习数据结构的 C++程序员，也可以作为高校师生学习计算机编程与数据结构的参考书，尤其适合报考计算机软件专业研究生的人员与参加信息奥赛班学习的人员参考。

侯识忠　湖南师范大学数学与计算机科学学院
侯凌翀　湖南省电信有限公司长沙市分公司

2005 年 3 月

目　　录

第 1 章　顺序存储结构的表、堆栈和队列

1.1　线性表的数组表示和实现

1.1.1　程序构思

本程序将 elem 表示成数据类型为结构体 ElemType 的数组。首先给出了线性表的类定义，然后给出了线性表中函数原型的实现部分，最后对线性表中的主要操作进行了测试，输出运行结果。

其中，对线性表按升序或降序输出 printlist(int mark)函数的形参 mark 为标志符，mark==0 为对线性表无序输出；mark==1 为对线性表升序输出；mark==-1 为对线性表降序输出。

1.1.2　线性表的类定义 linelist1.h

```
#define MaxListSize 20//顺序表的最大长度
#define EQUAL 1
typedef struct STU{
  char name[10];//姓名
  char stuno[10];//学号
  int age;//年龄
  int score;//成绩
}ElemType;
class List//线性表的类定义
{private:
  //线性表的数组表示
  ElemType elem[MaxListSize];//存线性表元素的数组
  int length; //线性表的当前长度
  int MaxSize; //线性表的最大长度
 public:
  //初始化顺序表
  void init(List **L,int ms);
  //删除顺序表
  void DestroyList(List &L){free(&L);}
  //将顺序表置为空表
  void ClearList(){length=0;}
  //判断顺序表是否为空表
  bool ListEmpty()
   {return length==0;}
```

```
  //判断顺序表是否为满
  bool ListFull()
   {return length==MaxSize;}
  //决定返回表中元素 pre_e 的前驱
  ElemType PriorElem(ElemType cur_e,ElemType &pre_e);
  // 决定返回表中元素 next_e 的后继
  ElemType NextElem(ElemType cur_e,ElemType &next_e);
  //从线性表中删除表头、表尾或等于给定值的元素
  bool ListDelete(int,ElemType &e);
  //遍历顺序表
  void ListTraverse();
  //返回顺序表的长度
  int ListLength();
  // 获取顺序表中第 i 个元素
  void GetElem(int,ElemType *);
  // 判断顺序表两元素是否相等
  bool EqualList(ElemType *,ElemType *);
  // 判断顺序表两元素是否不等
  bool Less_EqualList(ElemType *,ElemType *);
  //顺序表的查找算法
  bool LocateElem(ElemType,int);
  //更新线性表中的给定元素
  bool UpdateList(ElemType& e,ElemType);
  //顺序表的合并算法
  void MergeList(List *,List *);
  //顺序表的插入算法
  bool ListInsert(int,ElemType &);
  //顺序表的联合算法
  void UnionList(List *,List *);
  //对线性表按升序或降序输出
  void printlist(int);
};
```

1.1.3 线性表的实现 linelist1.cpp

```
#include "linelist1.h"
void List::init(List **L,int ms)
{*L=(List *)malloc(sizeof(List));
 (*L)->length=0;
 (*L)->MaxSize=ms;
}
int List::ListLength()
{return length;}
ElemType List::PriorElem(ElemType cur_e,ElemType &pre_e)
{for(int i=0;i<length;i++)
  if(i!=0&&strcmp(cur_e.name,elem[i].name)==0)
   {pre_e=elem[i-1];return pre_e;}
```