

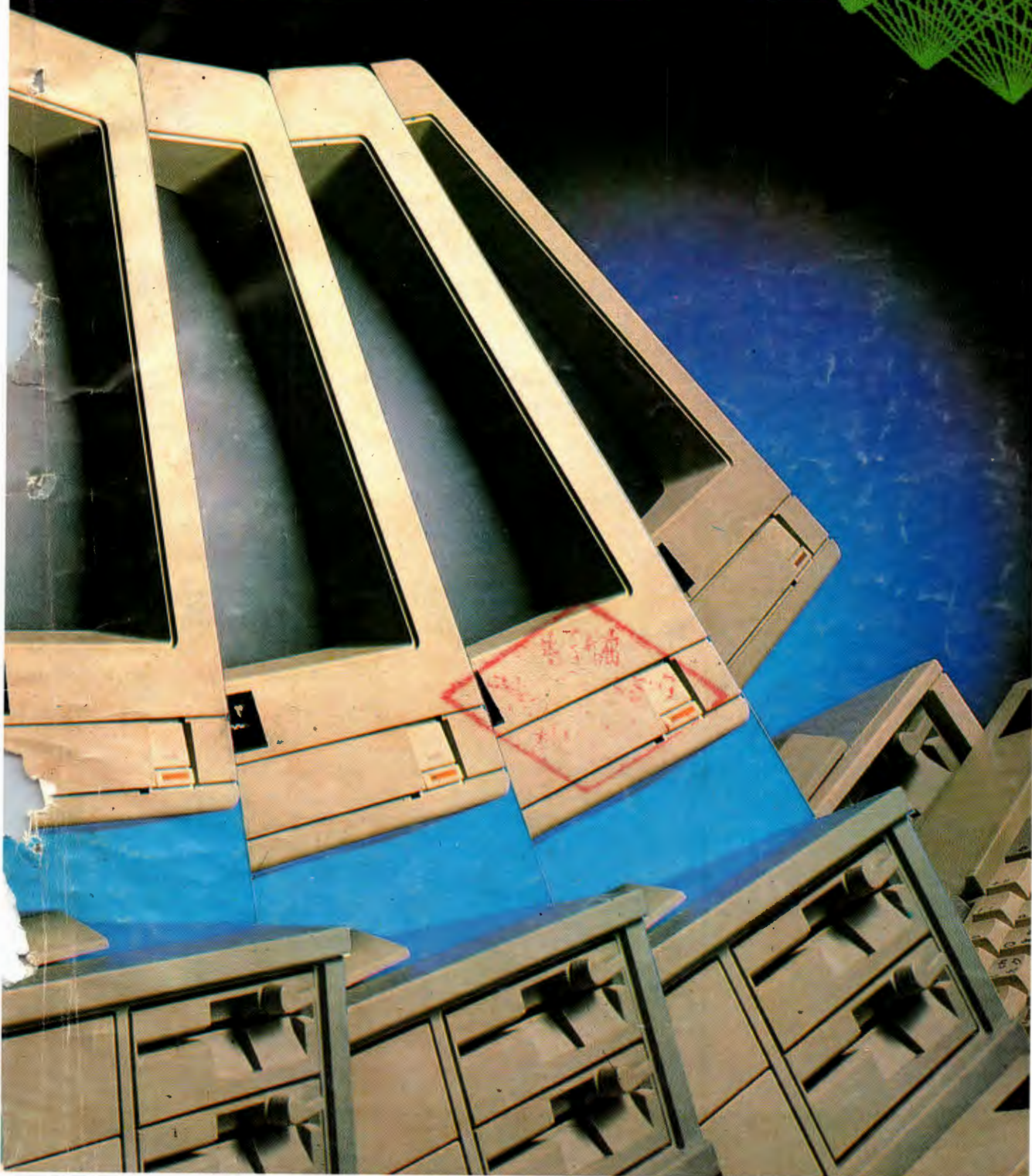
ISSN 1001-3695

计算机應用研究

1993

2

APPLICATION RESEARCH OF COMPUTERS 《计算机應用研究》雜誌社



新潮電腦

XINCHAO'S COMPUTERS

時代新潮

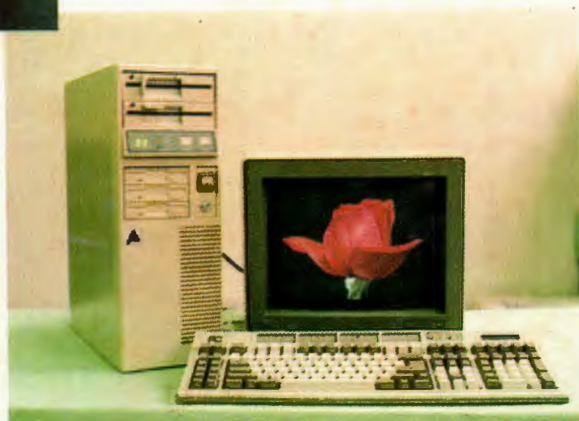
GIVE YOU A STYLISH FEELING IN YOUR WORK

新潮系列微機

高科技的象征

國產微機的新里程

- 先進的設計和工藝
- 高度的兼容性
- 廣大的維修服務網



● 攝像機及其應用系統

● 網絡工程

● 開關電源

● 亞森 UPS 卡

● 系列工業控制計算機

● 稱量儀表

● 微機電子皮帶秤



四川新潮計算機產業集團公司
SICHUAN XINCHAO COMPUTER ENTERPRISES GROUP CO.

地址：四川成都新濤路四號

電話：448100 443074

傳真：(028)444115 電挂：4615

郵政信箱：成都 606 信箱

郵政編碼：610051

四川新潮计算机产业集团公司

四川新潮计算机产业集团公司经四川省人民政府批准于1988年10月成立。集团公司是从事计算机类产品开发、生产、销售和技术服务及进出口贸易的全民所有制企业集团。

集团公司以新潮系列微机和开关电源、软件等为主导产品,迄今已先后开发、生产了新潮系列微机XC—PC、XC—286、XC—386、XC—486、开关电源10余种及卡式UPS电源,CCD摄象器及其应用系统,100多项软件产品等。

集团公司的高级技术人员在STD工业控制机、网络工程、通讯系统、机房工程、轻印刷系统、工程工作站、CAD等方面,进行了卓有成效的开拓、研究和服务。迄今为止,上述各方面已先后荣获国家、部、省的几十次奖励。

集团公司拥有一支技术层次较高的技术服务队伍,多年来在计算机的销售、维修、技术培训等方面为社会各界提供了优质的全面服务,为社会各界维修各类微机4万台次以上,培训计算机技术人才9235名。同时,集团公司在全国及省内有重庆、绵阳、自贡等数十个分部和维修站的支撑,极大地方便了各界用户。

集团公司十分注重同国际间的经济技术合作,已先后在深圳、蛇口等地建立了华德、雅德、鼎峰、新欣、合力等8家中外合资企业,引进了国外先进的技术,先进的设备,先进的管理方法。美国著名的IBM计算机就指名用华德的开关电源,它获得UL登记和加拿大CSA以及法国TUV证书。所属雅德公司年产40多万块OEM级别的计算机主机板以及各种功能卡,全部销往欧、亚、美洲各国。

集团公司拥有全国第一条完整的表面组装(SMT)生产线和具有国际水平的电源测试设备,具有创亿元以上产值的生产能力,年出口创汇2500多万美元。在全国计算机行业中名列前茅。

集团公司以“开拓、创新、团结、奉献”为企业精神,立足国内,面向世界。真诚欢迎国内外各界朋友与我们携手合作。集团公司将以精湛的技术、优秀的工作质量,高品质的产品,期待着为您服务的机会。

总经理:吕金才

电话:(028)443074,448100 传真:(028)444115

地址:成都新鸿路4号 邮编:610051

新潮计算机产业集团公司最新推出彩色电脑画像机

该公司采用最新软、硬件技术,在国内独家推出商品化彩色电脑画像机。所画图像生动逼真,色彩丰富,艺术效果显著,可作为彩色图像输入输出设备,更可作为彩色画像机,广泛应用于各风景名胜旅游点、街头闹市,为客户带来丰厚利润,只需三个月,即可回收全部投资。

该系统采用分离式组装,升级、维护方便,各部件均可独立用于其它目的。

系统使用方便、灵活、易学易用、功能强、图像尺寸可选,既支持彩色打印,亦支持单色打印,打印图像亮度可调。该公司将为新老用户推出彩色电脑画像机新的型号和新的软件版本。

欢迎订购,诚征代理!

《计算机应用研究》杂志办刊单位

四川省电子计算机应用研究中心
贵州省科委计算中心
安徽省计算中心
吉林省计算中心
内蒙古电子计算中心
青海省测试计算中心
四川省电子学会

新疆电子计算中心
甘肃省计算中心
广西计算中心
山东省计算中心
河南省计算中心
云南省电子计算中心

《计算机应用研究》杂志社董事会

董事长:周赛渝

董 事:孙传江 陆慰椿 闫长荣 王升亮
李天健 郑国基 叶大卫 乔中南
冯德成 朱 华 黎瑰常

《计算机应用研究》杂志编辑委员会

主任委员:张执谦

副主任委员:李泽民

委 员:贾洪钧 曾光初 龚宇清 罗海鹏
张湘金 张国栋 范德元 李文华
刘启茂 崔振远 刘铁军 杨剑波
余 凯

1993年第2期(总第52期) 出版日期:1993年3月 责任编辑:唐大利

计算机应用研究(双月刊) (公开发刊)

刊 号:CN51-1196/TP(国内)
ISSN 1001-3695(国外)

发行代号:62-68(国内)
BM4408(国外)

主 编:张执谦

副 主 编:李泽民

编 辑 出 版:《计算机应用研究》杂志社

通 讯 地 址:成都市人民南路4段11号附1号

邮 政 编 码:610041

印 刷:成都市新都华兴印务有限公司

订 阅 处:全国各地邮政局

国内总发行:成都市邮政局

国外总发行:中国国际图书贸易总公司

计算机应用研究 第 10 卷第 2 期(总第 52 期)

目 录

研究探讨

计算机在数论中的应用	颜松远(1)
高级矢量汉字库的生成及使用	邹希平(10)
最小覆盖算法及正确性证明	金银秋(39)
软件重用的模板方法	郭 江(46)
局域自适应卡尔曼滤波器的图像复原	贾英江(49)
MHS 中 P7 协议的设计与实现	邱大钧(52)

应用实践

FOXBASE 通用窗口式数据库文件装载机	张 懿(13)
一种易于维护的 FOXBASE 通用下拉式菜单	曹玉俊 刘路放(18)
在西文 DOS 操作系统下显示汉字文本文件	明 军(19)
微机普通显示器图像处理系统概述	午锁平 薛和华 刘叶梅(20)
多用户帐务处理系统的设计与实现	齐英武(22)
FOXBASE+2.10 报表通用打印程序	贾士超(24)
PC 系列微机总线与任天堂系列游戏机卡槽总线的接口	王让定 倪志新 冯百明(37)
陶瓷生产线计算机实时控制系统	唐 敏 唐 明(42)
工作站间调用:一种局部网应用开发方法	王慧强(44)
CGA 显示器低分辨率屏幕图形设置技术	姚运生(55)
计算机绘制和打印电路图的新方法	金 波(57)
TVGA256 色图象截取技术	季军杰(59)

维护维修

联机不打印故障分析	刘景明(33)
IT286DH 微机键盘故障维修一例	林勇奇(27)
小结硬盘不能自举的故障分析、排除及保护措施	汪文娟(28)
带字库打印机的打印针的故障检测	马良渝(29)
ROM BIOS 系统参数在 IBM PC 机维修中的应用二例	甘早斌(30)
爱华 AH1488D 单色显示器行扭及无显示故障的排除	赵明生(41)
GW—500 显示器维修两例	刘亮生 陈名海(62)

综述评论

计算机软件的发展趋势和潜在的突破	王庆春(31)
人工智能与专家系统	张鸿鸣(34)

动态简讯

简讯	(封二、12、21、27、33、62)
----------	---------------------

APPLICATION RESEARCH OF COMPUTERS

Vol. 10 No. 2 (Total 52)

CONTENTS

RESEARCH

The Applications of Computers in Number Theory	Yan Song yuan(1)
Generating and Operating Advanced Vector Base of Chinese Characters	Zou Xi ping(10)
Minimal Cover Algorithm and Proof of Program Correctness	Jin Yin qiu(39)
The Template Method of Software Reusing	Guo Jiang(46)
Adaptive Regional Kalman Filtering for Image Restoration	Jian Ying jiang(49)
Design and Implementation of P7 Protocol in MHS	Qiu Da jun(52)

APPLICATION

A Current Window Type of Data Base File Loader in FOXBASE	Zhang Yi(13)
A Current Push—down Menu of FOXBASE by Easily Maintenancing	Cao Yu jun et al. (18)
Displaying Texts and Files of Chinese Characters Under the Operating System of DOS in the Western Language	Ming Jun(19)
A System of Processing Picture on Common Display of Microcomputers	Wu Suo ping et al. (20)
Designing and Realizing the Processing System of Account with Multi Users	Qi Ying wu(22)
A Current Printing Program of Report Forms in FOXBASE+ 2.10	Jia Shi chao(24)
The Interface of the Bus on Microcomputers of PC and the Bus of Card Groove on Gamers of Ren Tian Tang's Name	Wang Rang ding et al. (37)
Microcomputer Control System for Ceramic Line	Tang Min Tang Ming(42)
Interworkstation Reference: An Approach to LAN Application Development	Wang Hui qiang(44)
The Technique for Installing Screen Figure of Low Resolving Power on the Display of CGA	Yao Yun sheng(55)
A New Method of Drawing and Printing Circuit Diagrams on Computers	Jin Bo(57)
Technique for Intercepting Image of TVGA 256 Colours	Ji Jun jie(59)

MAINTENANCE

Analyzing Fault for On—line not Printing	Liu Jing ming(33)
An Example about Maintenancing Fault of Keyboard on Microcomputer of IT 286 DH	Lin Yong qi(27)
Analyzing and Debugging Faults and Preseving of Brief Summary of Hard Disk not Self—starting	Wang Wen juan(28)
Testing Fault of Printing Needles on Printers with Character Base	Ma Liang yu(29)
Two Examples of Application for System Parameter of ROM BIOS in Maintenancing IBM PC	Gan Zao bin(30)
Debugging Line Swinging and Not Displaying on Monochrome Display of Aihua AH1488D	Zhao Ming sheng (41)
Two Examples of Maintenance on Display of GW—500	Liu Liang sheng et al. (62)

SURVEY

Developing Trend of Software on Computers and Latent Breakthrough	Wang Qing chun(31)
Artificial Intelligence and Expert System	Zhang Hong ming et al. (34)

INFORMATION

News in Brief	(Cover 2,12,21,27,33,62)
---------------------	--------------------------

计算机在数论中的应用

英国约克大学数学系 颜松远

摘要 计算机与数论有着十分密切的联系。一方面,计算机在数论中有着广泛的应用;另一方面,数论也在计算机科学中有着深入的应用;在本文中,我们力图提纲挈领地、简明扼要地介绍计算机在数论中的应用。具体而言,我们要介绍计算机在素数测试、整数分解、孙子定理以及哥德巴赫猜想等数论领域中的应用。

关键词 计算机 数论 应用 素数 整数 孙子定理 哥德巴赫猜想

一、引言 数论是数学中最古老、最纯粹、最重要的一个分支,它的初等部分便是大家所熟知的算术。在今天的社会里,即便是一个文盲,也略懂一些算术知识。相对于数学而言,计算机科学则是一个十分年轻的学科,它从出现到至今也不过短短四、五十年的时间。不过,尽管它们的“年龄”相差悬殊,但它们却有着十分密切的联系,并且这种联系完全是双向的。一方面,计算机在数论中有着广泛的应用,如在素数测试、整数分解、孙子定理以及哥德巴赫猜想等许多问题的研究中,都直接或间接地要依赖于计算机的帮助。另一方面,数论也在计算机科学中有着深入的应用,如在计算机算术、计算机设计、可计算性理论以及计算复杂性理论等计算机科学领域中,都需要大量用到数论的知识。更为重要的是,计算机科学与数论互相渗透、交叉影响,并逐步地改变着各自的面貌。比如目前蓬勃兴起的学科——计算数论,就是计算机科学与数论相互渗透、互相影响而产生的一门新兴的边缘学科。这正是:

“算机数论两兄弟, 形影相吊不分离;

算中有论根基深, 论中的有算虎添翼”。

当然,在这篇短文中,我们不可能面面俱到地介绍计算机在数论中的所有应用,也不可能讨论计算数论中的所有内容。事实上,我们只是选择几个具有重要意义的问题,如素数测试、整数分解、孙子定理以及哥德巴赫猜想等,介绍计算机在数论中的应用。

二、素数测试 素数测试是当代数论中、尤其是当代计算数论中的一个十分重要的研究领域。在诸如计算机代数、数学密码、保密计算以及网络安全等应用领域中,素数测试几乎是不可缺少的一个重要工具。同时,它还是连结纯粹数学与应用学科的一座重要桥梁。因此,素数测试无论是在纯粹学科中,还是在应用学科中,都具有十分重要的意义。但是,素数测试又是一个很困难的问题,具有计算上的难解性之特点,因此千百年以来,许多一流数学家以及近

代的许多优秀计算机科学家都曾涉足这个领域,这其中包括十七世纪法国某著名数学家麦什涅和费尔马、十八世纪著名瑞士数学家欧拉,以及享有数学王子之称的十九世纪德国数学家高斯。可是,尽管有这么多的名人来从事这项工作,但素数测试问题始终没有得到彻底解决。

所谓素数测试(或素性测试),就是测定一个整数的素合性,也即给定一个大于1的整数N,要求测定出该N到底是素数还是合数。根据数论的基本知识,我们知道,对于任意一个大于1的整数N,它要么是素数,要么是合数,二者必居其一且仅居其一。用比较形式一点的话讲,就是:

输入:给定一个大于1的整数N;

输出:如果N为素数,输出Yes;否则输出No。

这个问题从表面上看,似乎极简单,但它却是一道著名数学难题。

最早期最简单的素数测试算法大概要算“试除法”。所谓试除法,就是用1与N之间的整数去遍除N。即(我们将余数为零记之为0,将余数不为零记之为1):

$$\frac{n}{2} \Rightarrow 1/0$$

$$\frac{n}{3} \Rightarrow 1/0$$

...

$$\frac{n}{n-1} \Rightarrow 1/0$$

只要其中有一个除式的余数等于零,N就是合数。如果所有的除式之余数都不为零,那么N便是素数。当然,在实际上,我们完全不必要测试到n-1。下面我们引出一条重要的数论定理,据此可以设计出较好的试除算法来。

定理1: 设 $n > 1$ 。如果n没有小于或等于根号n的素因子,那么n就是素数。

根据这个定理,如果我们要测试整数n是不是素数,那么我们不必测试在1与根号n之前的所有

整数,而仅需测试在1与根号 n 之间的素数便可。例如,测试任何一个小于1000000的整数,我们至多仅需进行168次试除,因为根号1000000等于1000,而在1000之内只有168个素数。由此可知,这个方法还是比较快的。从数学上讲,这个方法不仅是可靠的,而且也是完备的(也即它说 n 是素数, n 就一定素数;它说 n 不是素数, n 就一定不是素数)。值得一提的是,这个定理不仅可以用来测试素数,而且可以用来指导生成在 n 之前的所有素数。著名的素数生成算法—Eratosthenes筛法,实际上与这个定理的思想是一致的。

下面这个适合于在计算机上实现的算法,就是根据上述定理而设计的:

算法1(试除测试算法):

begin

while $d_k \leq \sqrt{n}$

begin

if $d_k | n$

Output n is composite, goto Exit

else $k \leftarrow k+1$

end

output n is prime

Exit;

end

算法中的 $d[k]$ 为已知的从2到根号 n 的素数集合。我们用此集中的数去遍除 n 。

作为一个例子,我们来看看如何用这种试除法来确定一个大于1的整数 n 是不是素数。假定欲被测试的数是198371,我们用从1至445(即根号198371之近似值)之间的素数去除198371。首先,2不能整除198371,3、5、7、11等素数都不能,因此我们必须继续试,一直试到443为止,因为443是最接近445的素数。可是当我们试到163这个素数时,它可整除198371。这也就是说,198371除以163之余数为零。因此我们断定198371是一个合数而不是一个素数。相反,如果我们欲测定整数3989,那么,我们必须用在1与63之间的素数去试除3989。由于从2到61为止的所有素数都不能整除3989,因此3989是一个素数。

定理2(素数定理):设 $n>1$ 。如果 n 是一个很大的整数,则其小于或等于 n 的素数的个数 $\pi(n)$ 大约等于 $\frac{n}{\ln n}$ 。更精确的讲,我们有下列的式子:

$$\lim_{n \rightarrow \infty} \frac{\pi(n)}{\frac{n}{\ln n}} = 1$$

定理2所给定的素数个数的估计既非常简单又比较精确。下面我们给出十个根据定理2得出的估

计值,据此读者可以作一番比较:

n	$\frac{n}{\ln n}$	$\pi(n)$
101	4.3...	4
102	21.7...	25
103	144.7...	168
104	1085.7...	1239
105	8685.8...	9592
106	72382.4...	78498
107	620420.7...	644579
108	5428681.0...	5761455
109	48254942.4...	50847478
1010	434294481.9...	455052512
...

根据定理1,为了测定 n 的素合性,我们需要用小于等于根号 n 的所有素数去遍除 n ,再根据定理2,我们知道小于等于根号 n 的素数大约有如下这么多个:

$$\frac{\sqrt{n}}{\ln \sqrt{n}} = \frac{2\sqrt{n}}{\ln n}$$

假定我们在最现代化的计算机上计算,那么这种计算机执行一个试除至多仅需 $\frac{\ln n}{10^6}$ 秒钟。因此,我们最多仅需

$$\frac{2\sqrt{n}}{\ln n} \times \frac{\ln n}{10^6} = \frac{2\sqrt{n}}{10^6} \text{ 秒钟去测定 } n \text{ 的素合性。}$$

直至本世纪七十年代,试除法一直是最基本最常用的一种素数测定法。就是在今天,试除法也还有一定市场。不过当被测之整数很大时,这种方法是无效的。例如,当被测之整数 n 是一个30位的整数时,用这种方法大约要63年才能知道 n 是不是素数。

下面我们介绍一种稍好点的算法—费尔马法。

所谓费尔马法,就是基于费尔马小定理的一种测试素数的方法(费尔马小定理是法国数学家费尔马在1640年发明的)。

定理3(费尔马小定理):如果 P 为素数,且 $\gcd(a, p)=1$,则 $a^{p-1} \equiv 1 \pmod{p}$ 。根据这个定理,我们可以很容易地得出如下推论:

推论1:如果 $\gcd(a, N)=1$,且 $a^{N-1} \not\equiv 1 \pmod{N}$ 。则 N 为合数。

再根据这个推论,我们可以很容易地确定出 N 是不是合数。可能读者会问,当 N 很大时,计算 $a^{N-1} \pmod{N}$ 之复杂性是不是也会很大呢?有幸的是,在计算机上计算 $a^{N-1} \pmod{N}$ 之复杂性仅为 $O((\log_2 n))$,由于这是一个对数级的复杂性,因此这当然没有任何计算上的困难性。注意,前面刚介绍的试除法之计算复杂性为 $O(2^{\log_2 n/2})$,由于这是一个指数复杂性的算法,因此当 N 很大时,它无法在计算机上实现。

作为费尔马法的一个例子,我们来看看它如何确定 1736 这个整数是不是合数。首先我们选定 $a=2$ 。由于:

$$2^{1763-1} \equiv 742 \pmod{1763} \\ \not\equiv 1 \pmod{1763}$$

因此 1763 是一个合数而不是一个素数。如果我们用试除法来确定 1763 的素合性,那么必须用 1763 去遍除 2, 3, 5, 7, ..., 直至 41, 才能知道 1763 是合数而不是素数,因为 41 是第一个可以整除 1763 的素因子。

从数学上讲,费尔马法是可靠的,但不是完备的。也即它说整数 n 是合数,那么 n 肯定是合数;但若它说 n 是素数(即不是合数),那么我们就打个问号了,因为有时它会指鹿为马,不是素数它也是素数,关于这一点,很多人不理解,他们认为反正对于一个大于 1 的整数 n ,它不是素数就是合数,不是合数就是素数,就象一个数的奇偶性一样,它不是奇数就是偶数,不是偶数就是奇数。不过我们要强调的是,数的素合性可比奇偶性复杂多了。我们还是通过例子来说明这个问题吧,比如我们要测定 1387。由于:

$2^{1387-1} \not\equiv 1 \pmod{1387}$, 因此根据费尔马小定理,这不是一个合数而是一个素数。可是事实上 1387 根据不是素数而是合数,如果用试除法我们可以很容易得到 1387 的第一个素因子 19。因此我们称 1387 是伪素数(或基于 2 的伪素数),也即它看起来象素数可其实它并不是素数。由此我们可以得出一个基本结论,即如用费尔马法测定 n 是合数,那么 n 一定是合数;如用费尔马法测定 n 为素数,那么 n 可能真是素数,也可能不是素数。这完全是一个概率,也许正因为费尔马法具有一定的概率性,因此,它也可被称之为概率法,不过尽管此法有一定概率性,但在未能找到有效的素数测试算法之前,它仍不失为是一个好方法。并且,相对来说,出现这种概率性的错误机会是很小的。即便有错,我们还有一定的手段与办法来补救它。事实上,很多近代的一些基于概率的素数测试方法如强伪素数测试和随机测试等方法,都是在此基础上发展壮大起来的,并且这些方法的基本出发点都是为了增强其可靠性与完备性。若用现代概率测试法来测定 n ,如果 n 是一个具有 100 个十进制位的整数,那么它仅需 33 秒钟,200 位的整数需 8 分钟,就是 1000 位的整数,也只不过一个星期。由此可知,尽管概率法不太准确,但其速度倒确实是很快的,是试除法所望尘莫及的!当然,概率法并非最好的素数测试方法。事实上,我们今天已经有了很多基于深刻数学理论(如代数几何)的方法。

前面介绍的试除法及费尔马法是属于通用型方法,对一些具有特殊形式的整数,如麦什涅数,我们当然不必用通用型方法测试,而应采用特殊方法测试。下面我们简单介绍适应于麦什涅素数测试的 Lucas—Lehmer 方法:

所谓麦什涅数,是指形为如下之数。即: $M_p = 2^p - 1$, 其中 p 为素数,如果形为 $M_p = 2^p - 1$ 的麦什涅数是一个素数,那么,我们就称其为麦什涅素数。

Lucas—Lehmer 方法主要基于如下定理:

定理 4: 设 q 为奇素数,并用如下规则定义序列 $\langle Ln \rangle$:

$$L_0 = 4, L_{n+1} = (L_n^2 - 2) \pmod{2^q - 1}.$$

则 $2^q - 1$ 为素数且仅当 $L_{q-2} = 0$ 。

根据这个定理,我们可以很容易地设计出测试麦什涅素数的算法,即:

```
L=4
For i=3 to p do
  L=(L^2-2)mod(2^p-1)
  If L=0(mod 2^p-1)
    then 2^p-1 is prime
    else 2^p-1 is composite
```

这是一个很有效的测试麦什涅素数的算法。在具体计算时,为了加快其速度,一般都采用 Schonhage—Strassen 之快速富里埃速乘法来计算 L^2 (L^2 是最影响速度的一个计算项目)。

到目前为止,人们已经发现了前 32 个麦什涅素数。即(我们仅列出 $2^p - 1$ 中 p 之值):

$p = 2, 3, 5, 7, 13, 17, 19, 31, 67, 89, 107, 127, 521,$
 $607, 1279, 2203, 2281, 3217, 4253, 4423, 9689,$
 $9941, 11213, 19937, 21701, 23209, 44497, 86243,$
 $110503, 132049, 216091, 756839.$

其中 $p = 756839$ 这个麦什涅素数是刚于 1992 年 3 月间在英国 Cray 研究中心的 Cray—2 巨型计算机上发现的,这是当今世界上最大的素数,它具有 227832 个十进制数位。上面这个表列中所列出的比较大的几个素数,基本上都是用 Lucas—Lehmer 方法发现的。不过有一点需要强调的是, Lucas—Lehmer 法只适合于麦什涅素数的测试,对于具有一般形式的整数的素性测试,它是无能为力的。因此,对于一般形式的整数,就目前来讲,主要还得靠概率法或其它更现代化的方法。

解决素数测试问题的关键是研制出具有多项式复杂性的测试算法。可是遗憾的是,到目前为止,人们还没有找到这样的一种算法。当然,也无人能否定说没有这种算法存在的可能性。不过理论性的结果已经表明:在广义黎曼假设的情况下,素数测试问题属于 P (即多项式);如果消掉这个假设,那么它属于

$co-RP$ (即随机多项式的补集)和 NP (即非确定型多项式);另外,它还可以用确定型算法在 $O(n^{O(\log \log n)})$ 计算步骤内得到解决。最新的结果则更表明素数测试问题在 RP 与 ZPP ($ZPP = R \cap co-R$) 之间。这也就是说,素数测试问题可以在概率多项式时间内得到解决。总而言之,虽然我们还没有找到确定型多项式复杂性的素数测试算法,但我们找到了测试素数的随机(即概率)多项式算法,尽管随机多项式算法(RP)没有确定型多项式算法(P)好,但它却比非确定型多项式算法(NP)及指数复杂性算法要好得多。由此可知,尽管素数测试问题还没有得到彻底解决,但也是比较接近彻底解决了。

由于素数有无穷多个(根据欧几里德定理),因此寻找特大的素数、尤其是寻找具有一般形式的特大素数,是当前素数测试的一个重要研究方向,因为具有一般形式的特大素数在密码设计等领域中有重大的应用价值。

三、整数分解 整数分解是当前计算数论中最重要的研究课题之一。就其重要性、应用性以及影响力而言,数论中所有与计算机相关的问题,没有一个可以与整数分解相提并论。

大家知道,数论中的许多问题,都具有一个显著的特点,即看起来似乎很容易但解起来却非常的困难。整数分解就是属于这类问题中的一个。到目前为止,人们还没有找到一种分解整数的有效的算法,但又没有人能否定这种法算存在的可能性。也许正因如此,从古至今,这个问题才吸引了许许多多的数学家以及计算机科学家,其中包括历史上的名家费尔马、欧拉以及高斯。与我们前面刚介绍的素数测试相比,整数分解可就要更困难得多、更重要得多了。

早在两千年前的欧几里德时代,人们就知道任何一个大于1的整数,都可以表示成素数的乘积形式。当然这个问题的最终的精确描述与理论证明,则是1801年在数学王子高斯手上完成的。这就是我们今天人所皆知的“算术基本定理”。

定理5(算术基本定理):任何一个大于1的整数,都可以唯一地表示成如下之素数的乘积形式,即: $n = p_1^{e_1} p_2^{e_2} \cdots p_r^{e_r}$, 其中, $p_1 < p_2 < \cdots < p_r$ 为素数, $e_1 < e_2 < \cdots < e_r$ 为正整数。

例如,123456789是一个大于1的整数,因此,它可以唯一地表示成如下的素数的乘积形式,即:

$$123456789 = 3^2 \cdot 3607^1 \cdot 3803^1$$

其中3,3607和3803均为素数。

在此值得特别一提的是,当大于1的这个整数是素数时,那么我们认为它已被分解成了素数的乘积形式,此乘积形式只有一个因子,就是它自己。例

如,素数1234567之素数乘积形式可以简单地写成 $123457 = 123457$ 。当整数 n 不是很大时,比如说只是一个具有五、六位或七、八位十进制数的整数,我们可以手算。但是当这个数是一个比较大的整数时,要手算是很困难的,甚至是不可能的。因此,我们必须依靠计算机,必须研究出适合于在计算机上实现的有效分解算法。

因此,根据算术基本定理,所谓整数分解,就是要将一个大于1的整数分解成素数的乘积形式,即:

输入:给定一个大于1的整数 N ;

输出:如果 N 为合数,输出 N 的所有素因子;

否则指出 N 是素数。

与素数测试类似,最简单最原始的计算机分解整数的算法,也就是基于如下定理的试除法:

定理6. 如果 n 为合数,那么其素因子必定都小于或等于根号 n 。

根据定理6,我们可以将算法设计成:假定欲被分解的那个整数为 n 。该法用从2开始直至根号 n 为止的所有素数去遍除 n ,只要该素数能被 n 整除,那么这个素数就是 n 的一个因子。当然,在具体实现上,还要把握住两个循环,即一个外循环和一个内循环。外循环当然是指从2到根号 n 的那个循环,而内循环则在被找到的那个因子上。例如,对于16,首先找到因子2,此时并不急于从2跳到3上去执行外循环,而应在2上执行内循环。事实上,对于16024只需在2上再执行9次内循环,便可得到 $1024 = 2^{10}$ 。外循环它实际上只执行了一次。

下面是关于这个算法的形式描述:

算法2(试除分解算法):

```

input n                                (输入欲被分解的整数)
k=0, t=0,                             (置初值)
A: if n=1 go to Exit                   (如果 n=1, 则算法终止)
B: q=n/d[k]                           (q 为 n 除以 d[k] 之商)
r=n mod d[k]                           (r 为 n 除以 d[k] 之余数)
if r != 0 go to c
else t=t+1                             (找到一个因子)
p[t]=d[k]
q=n
print p[t]
go to A
C: if q>0
k=k+1
go to B
t=t+1
p[t]=n                                (n 是一个素数)
print p[t]
Exit:                                  (算法之出口)

```

在这个算法中,我们假定 $d[0]=2, d[1]=3, d[3]=5, \dots, d[k] \leq \sqrt{n}$ 为至少直至根号 n 的所有素数。在具体实现上,这些素数可以作为一个外部文件读入程序中。由于这种算法采用试凑性的方法用所有可能的素因子去遍除 N , 因此该法一般被称之为试除法。该算法的运行时间与 $\max(p_{i-1}, \sqrt{pt})$ 成比例, 即其算法复杂性为 $O(\max(p_{i-1}, \sqrt{p_i}))$ 。

试除分解算法的可靠性与完备性是显然的, 只要时间空间足够, 它就可以将 N 的因子全部分解出来。但问题是当被分解的数的位数增多时, 其算法的复杂性以指数倍数增长。根据计算复杂性的理论, 计算机只能执行具有多项式或小于多项式复杂性的算法。因此, 对于这个算法, 它没有能力分解那些比如说大于 20 位的整数。由此看来, 对于整数分解, 更重要的问题不在于找到一个算法, 而在于找到一个有效的算法。所谓有效的算法, 是指那些具有多项式复杂性的算法, 简称为多项式算法。当然, 这种算法是不太好找的。事实上, 到目前为止, 人们还没有找到一种多项式的整数分解算法, 但人们又不能否定这种算法存在的可能性。但有一点是可以肯定的, 我们至少可以找到比算法 1 更精细更有效的算法。比较现代一点的算法包括基于概率论的 RHO 方法、基于连分数的 CFRAC 方法、基于代数几何的 ECM 法以及基于代数数论的数域筛法 NFS 等等。我们在此便不多叙。

前面我们曾提及, 到目前为止, 人们还没有找到一种适合于在计算机上执行的“有效”的整数分解算法(即确定型多项式算法)。就今天的算法与机器条件而言, 人们分解一个随机的 100 位的整数问题不大(注意, 直到本世纪七十年代, 人类分解整数的能力只限于 40 位以下), 但对于一个 150 位的整数, 就已经相当困难了, 当然并非不可能, 但对于一个 200 位的整数, 目前人们是无论如何也分解不出来的, 这也就是说, 目前人们还没有找到一种有效的整数分解算法。但是, 目前没有找到并不意味着就不存在。事实上, 数学家们一直在寻求这种有效的整数分解算法, 而这种寻求工作又突出地集中在一种称之为费尔马的数上, 因这种数具有特殊形式, 相对来讲分解起来要容易些。事实上, 数学家们是想以费尔马数为突破口, 通过分解这种特殊型式的数寻找规律, 研究技巧, 为从特殊到一般奠定基础, 并期望最终找到一种通用的有效的整数分解算法。

所谓费尔马数, 就是以十七世纪法国著名数学家费尔马的名字命名的, 具有如下形式的数:

$$F_n = 2^{2^n} + 1, n \geq 0$$

早在 1640 年, 费尔马便曾猜测所有这种形式的数均为素数, 因为他已经知道 F_0 的前五个数全为素数。即:

$$F_0 = 2^{2^0} + 1 = 3,$$

$$F_1 = 2^{2^1} + 1 = 5,$$

$$F_2 = 2^{2^2} + 1 = 17,$$

$$F_3 = 2^{2^3} + 1 = 257,$$

$$F_4 = 2^{2^4} + 1 = 65537,$$

全为素数。

当然, 我们今天都知道, 费尔马的猜测是错误的, 因为早在 1732 年, 瑞士著名数学大师欧拉就已验证 F_5 是合数而不是素数, 因为他将 F_5 分解成了素数的乘积形式。即:

$$F_5 = 2^{2^5} + 1 = 4294967297 = 641 \cdot 6700417$$

事实上, 就人们今天所掌握的情况而言, 我们也仅知道这前五个费尔马数是素数。至于其它的费尔马数, 它们要么是合数, 要么其素合性至今仍确定不了。在欧拉后的 148 年间(即 1880 年), 法国数学家 Landry 和 Lasseur 彻底分解了第六个费尔马数 F_6 。即他们得到:

$$F_6 = 2^{2^6} + 1 = 2^{64} + 1 = 274177 \cdot 67280421310721$$

不过他们是用什么方法分解出这个数的, 人们并不清楚, 因为他们没有发表他们所用的方法。

第七个费尔马数 F_7 是美国数学家 Morrison 和 Brillhart 在 1970 年应用连分数算法(即 CFRAC 法)分解出来的。

$$\text{即: } F_7 = 2^{2^7} + 1$$

$$= 2^{128} + 1$$

$$= 59649589127497217 \cdot 5704689200685129054721$$

连分数法是一个比较通用的分解算法, 它也适用于其它形式的整数的分解。

对于第八个费尔马数, 即:

$$F_8 = 2^{2^8} + 1 = 2^{256} + 1 = 1238926361552897 \times$$

$$9346163971535797776916355819960689658405123754$$

$$16381885180280321$$

人们早在 1909 年就知道它是一个合数, 但直到 1980 年才由 Brent 和 Pollard 应用一种基于概率论方法的蒙特卡罗算法(即改进的 rho 方法)将它完全分解成素数的乘积形式的。

时隔十年后(即 1990 年), 英美荷等好几个国家的数学家联合作战, 并采用当代最先进的基于代数数论的数域筛法(即 NFS 法), 以及计算机网络技术, 尤其是电子邮件技术(即 e-mail)将分布在世界各地的 700 多台计算机相联, 最终又在一台巨型机上汇总, 前后总共用了 4 个多月, 才彻底分解了第九个费尔马数。即:

$$F_9 = 2^{2^9} + 1 = 2^{512} + 1$$

= 2424833. p_{49} , p_{99} (p_{49} 与 p_{99} 分别为具有49位与99位的素数)

在这里值得一提的是,分解 F_9 所用的 NFS 算法虽然是一种特殊的方法,但它可望扩展成一般的方法,使之适合于分解一般形式的整数。

第 11 个费尔马数是澳大利亚计算机科学家 Brent 于 1988 年采用椭圆曲线法(即 ECM 法)在 VP100 巨型机上彻底分解出来的。即:

$$F_{11} = 2^{2^{11}} + 1$$

$$= 2^{2048} + 1 = 319489.974849.$$

$$167988556341760475137.3560841906445833$$

$$920513. p_{564} (p_{564} \text{为一具有 } 564 \text{ 位数的素数})$$

目前未被分解的最小的费尔马数,便是第十个费尔马数 F_{10} ,因此,分解第十个费尔马数是当今国际计算数论界的一个主攻课题。

四、“孙子定理” 我国是一个具有几千年历史的文明古国,是数学的故乡。我们中华民族的先哲为古代数学的创造与发明做出了杰出的贡献,写出了许多优秀的、最早期的数学著作,如《周髀算经》以及《九章算术》等等。《孙子算经》也是我国古代的一部数学巨著。《孙子算经》共分上、中、下三卷。上卷叙述筹算的乘除法,中卷叙述筹算的分数算法及开方方法。下卷则收集了一些算术难题,如已知头数和足数的“鸡兔同笼”问题等等。不过在《孙子算经》中,最有名的要算下卷中的第 26 题,即所谓的“孙子问题”。这个问题的原文为:

“今有物不知其数,三三数之余二,五五数之余三,七七数之余二,问物几何?”

这就是名扬天下、誉满全球的“孙子定理”之肇源。“孙子定理”在国外则被尊称为“中国剩余定理”或“中国余数定理”。事实上,只要打开国际上任何一本标准数论讲义,都少不了有一个专门的章节来讲述“孙子定理”。稍微现代一点的数论著作则更要大书“孙子定理”在快速计算、计算机设计、密码设计、数值分析、数据库设计以及查询检索等方面的应用。由此可知,“孙子定理”在当代国际数学界中的重要地位。

“孙子问题”可以用代数语言来作更一般的描述,即:

“现有一数 X ,以 m_1 除之余 r_1 ,以 m_2 除之余 r_2 ,以 m_3 除之余 r_3 ,问 X 是多少?”

假如我们用同余式 $x \equiv r_1 \pmod{m_1}$ 来表示 x ,以 m_1 除余 r_1 ,则这个问题相当于求解一个一次同余方程组,也即求出满足如下方程:

$$\begin{cases} x \equiv r_1 \pmod{m_1} \\ x \equiv r_2 \pmod{m_2} \\ x \equiv r_3 \pmod{m_3} \end{cases}$$

的解 X 来。

“孙子问题”在我国民间流传甚广,并有“秦王暗点兵”、“韩信点兵”、“鬼谷算”、“剪管术”、“神奇妙算”、“物不数”、“大衍求一术”,以及“隔墙算”等许多不同的名字。在宋代(420—479 年间)的一本算术笔记中,还曾把它的解法写成下面四句诗:

“三岁孩儿七十稀,五留二一事尤奇,
七度上元重相会,寒食清明便可知。”

到了 1583 年,明代数学家程大位则在他的数学著名《算法统宗》中进一步将其叙述为如下的四句诗,即:

三人同行七十稀,五树梅花廿一枝,
七子团圆月正半,除百零五便得知。”

如把这首诗翻译成白话文,便是:

“以三三除之余数乘七十,
五五除之余数乘二十一,
七七除之余数乘十五,
总加之并减去 105 的倍数,
便为所求。”

若列成算式,则为:

$$2 \cdot 70 + 3 \cdot 21 + 2 \cdot 15 - 210 = 23.$$

程大位的这个解法是非常奇妙的,尤其是 70, 21, 15 这三个数,更是设计得巧妙极了,真可谓是神奇妙算!不过,程氏的这个解法并不是凭空捏造的,而是以一定的天文现象作背景的,这大概就是为什么我国古代的很多数学家,如祖冲之父子等,同时也是天文学家之缘故吧!另外,用诗的形式来解答数学问题,这也算是我们祖先的一大发明!

“孙子问题”的进一步发展,是将它总结成如下这个“孙子定理”。

定理 7(孙子定理): 设 m_1, m_2, \dots, m_n 均为 >1 的两两互素的正数, a_1, a_2, \dots, a_n 为任意整数,则下列同余方程组:

$$\begin{cases} x \equiv a_1 \pmod{m_1}, \\ x \equiv a_2 \pmod{m_2}, \\ \dots, \\ \dots, \\ x \equiv a_n \pmod{m_n} \end{cases}$$

具有一个解 x ,如果有两个解 x 和 x' 的话,那么 $x \equiv x' \pmod{M}$, 其中, $M = m_1 m_2 \dots m_n$.

关于这个定理的证明,读者可以查阅任何一本数论书,我们便不在此赘述。关于这个定理的重要性,我们要强调三点:

1. 虽然孙子本人并没有形式地提出这个定理,但这个定理的思路完全是从“孙子问题”中得来的,

因此我们完全可以说这个定理是孙子(注:国外文献中尊称孙子为孙大师)发明的。

2. 这个定理的证明的本身就提供了一种求解同余方程组的完整算法,并且这种证明完全可以是构造性的,因此它极易在计算机上实现。从这个角度讲,孙子也是一名出色的计算机科学家(因为计算机科学的核心是研究算法),尽管那时还没有计算机。美国著名计算机科学家 Knuth 称古巴比伦人是优秀的计算机科学家,事实上,我们中华民族的先哲也是优秀的计算机科学家。

3. 孙子定理具有很强的应用性,它在数值分析,计算机设计,计算机算术,密码设计,最优 hash 函数设计,数据库保密,快速计算以及环论等许多领域中都有重要应用。例如当代最有名的 RSA 密码技术,实际上它翻来复去就是用孙子定理而已,孙子定理成了它不可缺少的一个重要工具。据作者统计分析,孙子定理完全可以与古希腊人发明的著名的欧几里德算法相提并论、平起平坐,甚至比它还重要,还常用。

下面,我们给出“孙子定理”的一个 C 语言的程序实现:

```
#include <stdio.h>
#include <stdlib.h>
#define MAXi 100
int inv(a,n)
int a,n;
{
    int i,b,t,X,Y;
    int g[MAXi];
    int x[MAXi];
    int y[MAXi];
    g[0]=n;g[1]=a;
    x[0]=1;x[1]=0;
    y[0]=0;y[1]=1;
    i=1;
    while (g[i] != 0)
    {
        t=g[i-1]/g[i];
        g[i+1]=g[i-1]-t*g[i];
        x[i+1]=x[i-1]-t*x[i];
        y[i+1]=y[i-1]-t*y[i];
        i=i+1;
    }
    x=y[i-1];
    if (x<0) y=x;
    else Y=X+n;
    return (Y);
}
main()
{
    /* CRT(n,p[1],...,p[r],x[1],...,x[r])
```

```
(return x=[0,n-1]) */
int i,t,n,crt;
int x[MAXi];
int p[MAXi];
int y[MAXi];
while (1)
{
    n=1;
    crt=0;
    printf("\n Chinese Remainder Theorem");
    printf("\n Computing x=x[i] mod p[i]");
    printf("\n ----- \n");
    printf(" Enter Integers t:");
    scanf("%d",&t);
    for (i=1;i<=t;i++)
    {
        printf("Enter Integers x[i] and p[i]:");
        scanf("%d %d",&x[i],&p[i]);
        n=n*p[i];
    }
    for (i=1;i<=t;i++)
    {
        y[i]=0;
    }
    for (i=1;i<=t;i++)
    {
        y[i]=inv((n/p[i]),p[i]);
        printf(" inv=%d\n",y[i]);
    }
    for (i=1;i<=t;i++)
    {
        crt=(crt + (n/p[i]*y[i]*x[i]))%n;
        printf(" crt =%d\n",crt);
    }
}
```

通过运行这个程序,我们可以求解各种各样的同余方程组。比如我们可以用它来求解【孙子算经】中的“孙子问题”:

$$\begin{cases} x \equiv 2 \pmod{3}, \\ x \equiv 3 \pmod{5}, \\ x \equiv 2 \pmod{7}. \end{cases}$$

通过调用、执行上面的程序,我们可得如下结果:

```
Chinese Remainder Theorem
Computing x=x[i] mod p[i]
-----
Enter Integers t,3
Enter Integers x[i] and p[i]:23
Enter Integers x[i] and p[i]:35
Enter Integers x[i] and p[i]:27
x=23
```

其中的 Enter 部分表示需要用户输入信息,如方程组中方程的个数 t,以及诸方程中 x 与模的值等,最

后一行则是所求 x 之值。

五、“哥德巴赫猜想”早在1742年6月7日，德国数学家哥德巴赫便写信给瑞士数学家欧拉。在信中哥德巴赫猜测：任何一个大于4的偶数，都可以表示成两个素数之和。例如，下面这些等式显然满足哥德巴赫猜想：

$$6=3+3$$

$$8=3+5$$

.....

$$14=3+11=7+7$$

$$100=3+97=11+89=17+83=29+71=41+59=47+53$$

.....

以第一个式子为例，6(是一个偶数)可以表示成两个素数(即两个3,3是素数)之和。其余的式子均可依此类推。

欧拉并没有能够证明这个猜测的正确性，尽管欧拉号称是解难题的能手。在1922年，英国著名数学家哈代和李特伍德就曾指出：哥德巴赫的正确性是至信无疑的，但却谁也无法证明其正确性。

对于一般的科学家而言，如果我们做了成千上万次试验，结果都是一个，那么我们完全可以说这个结果是正确的。但是，对于数学家来讲，就是做了一百亿次试验，并且次次都正确，我们也不能说这个结果是正确的，因为在数学中，一百亿是个很小的数字。

到目前为止，关于哥德巴赫猜想的最好的理论结果是我国数学家陈景润在1966年得出的结果(尽管他的最终结果是迟至1973年才发表的)。他证明了任何一个充分大的偶数都可以表示成一个素数再加上两个素数之积(即所谓的 $1+2$)。由于两个素数之积并不是一个素数，例如2乘以3等于6就不是一个素数，尽管2和3都是素数，况且这个充分大的偶数到底有多大、到底是多少，目前我们并不知道，因此，陈景润教授的结果与哥德巴赫的原本猜想还相距甚远。

目前，在解决哥德巴赫猜想方面，有两种主要的方法：一种是解析数论的方法，另一种是计算数论的方法。就解析数论来讲，常规的圆法与筛法对哥德巴赫猜想已是力不从心了，必须寻求新的方法。就计算数论来讲，目前最好的结果，是由三个荷兰数学家计算出来的，他们于1989年在巨型向量计算机CDC CYBER 205上验证了哥德巴赫猜想一直到 $2 \cdot 10^{10}$ 都正确。可是，在数学上， $2 \cdot 10^{10}$ 是一个小的数字，对于比这个数还大的数是否也满足哥德巴赫猜想呢？目前人们并不知道。就是有一天人们能够突破(并不

是很困难!)这个数字即($2 \cdot 10^{10}$)而得到一个新的数字，但比这个新的数还大的那个数是否也满足哥德巴赫猜想呢？对于这些问题，目前人们可以说是根本无法回答。

为什么解决哥德巴赫猜想会如此之困难呢？首先，在数学理论上，人们还没有找到一种强有力的新的证明方法，该方法可以证明任何一个大于4的偶数都可以表示成两个素数之和。其次，在计算理论上，目前人们还没有找到一种有效的算法，该算法能够在多项式时间内，在计算机上予以解决。在这里，我们要特别强调，计算机只能执行具有多项式或比多项式还小如对数、线性复杂性的算法，而对于指数复杂性以及比指数复杂性还大(如阶乘复杂性)的算法，计算机是无能为力的。可是，目前人们所用于解决哥德巴赫猜想的一些计算机算法基本上都是基于穷举式的，其复杂性都远远超过了多项式，而在指数范围内，因此，这些算法当然不可能在现有的计算机条件下解决哥德巴赫猜想。

由于哥德巴赫猜想至少是具有计算上的难解性，即目前人们还没有找到具有多项式复杂性的计算机算法，因此，在人们还没有发明出新的强有力的数学工具之前、或在没有发明出新的有效的(即具有多项式复杂性的)计算机算法之前，哥德巴赫猜想是不可能得到解决的。在这里，我们认为值得一提的是，在数论中，一个问题两百多年没有得到解决是一点也不奇怪的，因为有的问题甚至花了两千多年才得到解决。

下面我们给出荷兰数学家们设计的几个计算哥德巴赫猜想的FORTRAN程序，供有兴趣的读者参考：

哥德巴赫算法 I :

```

C
C WE ASSUME THE INTEGER ARRAY PIND(N) HAS
BEEN INITIALIZED TO ZERO
C
DO 20 N = N1, N2, 2
C WE SUPPOSE THAT N1 AND N2 ARE EVEN;
C WE SEARCH FOR THE MINIMAL GOLDBACH PARTI-
TION OF N
C
DO 10 I = 1, IMAX1
IF (PRIME (N-PR(I))) THEN
PIND(N)=1
GO TO 20
ENDIF
10 CONTINUE
C NO GOLDBACH PARTITION N = P+Q FOUND WITH P
<= PR(IMAX1)
C INCREASE THE VALUE OF IMAX1
C
20 CONTINUE
哥德巴赫算法 II
C
C WE ASSUME THE INTEGER ARRAY PIND(N) AND

```



```

ODDPR HAVE BEEN INITIALIZED TO ZERO
C
DO 20 N=1,IMAX2
C WE SUPPOSE THAT N1 AND N2 ARE EVEN,
C WE SEARCH FOR THE MINIMAL GOLDBACH PARTI-
TION OF N
C
DO 10 I=1,IMAX1
PRI =PR(I)
DO 10 I=N1,N2,2
IF (PIND(N),EQ.0. AND. ODDPR(N-PRI).EQ.1) PIND
(N)=1
PIND(N)=1
10 CONTINUE
20 CONTINUE
C TREAT THE EVEN N FOR WHICH PIND (N) IS STILL
ZERO,
C I. E. FOR WHICH NO GOLDBACH PARTITION HAS
BEEN FOUND YET
C
CO 40 I=N1,N2,2
IF (PIND(N).GT.0)GOTO 40
DO 30 I=IMAX2+1,IMAX1
IF (ODDPR(N-PRI(I).EQ.1)THEN
PIND(N)=1
GOTO 40
END IF
30 CONTINUE
40 CONTINUE

```

六、结束语 在本文中,我们论述了计算机与数论之间的联系,指出它们是互相依赖、互相联系、互相影响、互相渗透、互相促进的。计算机在数论中有着广泛的应用,数论也在计算机科学中有着深入的应用,而计算数论则又是这两门学科交叉渗透而形成的一门边缘学科。就数论本身来讲,除了传统的初等数论外,它已逐步形成了诸如组合数论,代数数论,解析数论,概率数论,几何数论等等,而计算数论则与所有这些数论分支都有关系,因为它们都少不了计算,都离不开计算机。因此,计算数论就象一座桥梁一样,沟通了数论诸分支之间的联系。同时,它又大量应用着计算机科学中的研究成果,尤其是并行算法,巨型机方面的研究成果。计算机在数论中的应用是如此之广泛,而计算数论的内容又是如此之丰富,使得我们不可能在这样一篇文章中介绍很多的内容。我们只能提纲挈领地、简明扼要地介绍计算机在素数测试、整数分解、孙子定理以及哥德巴赫猜想等数论领域中的应用。至于数论在计算机科学中的应用,我们则基本上没有提及,我们期望本文能对那些对数论感兴趣的计算机科学与应用人员有所帮助。我们在下面列出了一些参考文献,供有兴趣的读者去进一步参阅。就目前来讲,有关计算数论方面的资料,大都散发在不同的刊物(杂志)之中,并且技术性很强,不易阅读,从这个角度讲,本文作者所著的【计算数论导引】一书,很值得大家阅读、参考。

参考文献

1. L. Adleman and k. s. mcCurley, Open Problems in Num-

ber Theoretic Algorithms, in: Discret Algorithms and Complexity, Academic Press, 1987, 237-262.

2. E. Bach, Intractable Problems in Number Theory, in: Advances in Cryptology — CRYPTO'89, Lecture Notes in Computer Science 403, Springer — Verlag, 1989, 77-93.
3. E. Bach, Number — Theoretical Algorithms, Annual Review of Computer Science, 1990, 119-172.
4. R. P. Brent, Parallel Algorithms for Integer Factorization, CMR-R49-89, Center for Mathematical Analysis, Australian National University, Canberra, 1989
5. J. D. Dixon, Factorization and Primality Tests, The American Mathematical Monthly, 91, 1984, 333-352
6. A. Granville, J. van de Lune and H. J. te Riele, Checking the Goldbach Conjecture on a Vector Copmputer, Report NM-R8812, Dept of Numerical Mathemaics, CWI, Amsterdam, 1988.
7. G. H. Hardy and E. M. Wright, An Introduction to the Theory of Numbers, Clarendos Press, Oxford, 5th Edition, 1979.
8. D. S. Johnson, The NP-Completeness Column, An Ongoing Guide, Journal of Algorithms, 7, 1986, 584-601.
9. A. K. Lenstra, H. W. Lenstra, Jr, M. S. Manasse and J. M. Pollard, The Number Field Sieve, in: proc. 22nd Annual ACM Symp. on Theory of Computation, Baltimore, May 14-16, 1990, 564-572.
10. A. K. Lenstra and M. S. Manasse, Factoring by Electronic Mail, in: Advances in Cryptology — EUROCRYPT'89, Lecture Notes in Computer Science, Springer-Verlag, 1990, 355-371.
11. D. E. Knuth, The Art of Computer Programming: Seminumerical Algorithms, 2nd Edition, Addison — Wesley, 1981.
12. H. J. J. te Riele, Parallel Processing in Number — Theoretical Problems, Report NM-R9010, Dept of Numerical Mathematics, CWI, Amsterdam, 1990.
13. R. L. Rivest, A. Shamir and L. Adleman, A Method for Obtaining Digital Signatures and Public Key Cryptosystems, Communications of the ACM, 21, 2, 1978, 120-126.
14. S. S. Wagstaff, Jr. and J. W. Smith, Methods of factoring Large Integers in: Number Theory, edited by D. V. Chudonvsky, G. V. Chudnovsky, H. Cohn and M. B. Nathanson, Lecture Notes in Mathematics, Springer-Verlag, 1984, 281-303.
15. S. Y. Yan, Introduction to Computational Number Theory, to be published in Gordon and Breach Science Publishers, London. (颜松远, 计算数论导引, 伦敦, 待出版)

高级矢量汉字库的生成及使用

中条山有色金属公司计算站 邹希平 (山西垣曲 043700)

利用计算机在绘图仪上做图是计算机管理信息系统中的—个重要组成部份,很多在屏幕上显示的图形如折线图、直方图等如能用绘图仪画出来,则有较好的保存价值。尤其是一些生产管理中的图形如矿山的深孔图等,如能在绘图仪上做出,则更具有实用价值。但是,现有的绘图仪不能使用点阵字库来写汉字确是一大憾事。为了解决这个问题,利用现有的点阵字库生成矢量字库是一个省时省力的方法。我们利用 2.13H 汉字系统中的高点阵字库生成了非常漂亮的矢量字库,使绘图仪的应用增色不少。现将其方法介绍如下:

一、高点阵汉字库的构成

为了进行汉字转换,清楚地了解点阵字库的构成是首要条件。2.13H 系统中有 4 个高点阵字库,分别为宋体、仿宋体、黑体和楷体。它们的结构是相同的,均为 40×36 的点阵,每个汉字占 180 字节,一个字库中有 72 个区(不包括 1—15 区)。为了打印方便,点阵字库采用竖排结构,其构成如附图(略),纵向共 40 排列,分为 5 部分,8 个点为一个字节,每部分占 36 个字节,一个汉字共占 180 字节。字节的排列顺序如附图中所标。

二、转换的思路及原则

把点阵字库转换为矢量字库的基本设想是:以一点为起点,沿某一方向搜索,直到没有点为止,即为一笔,搜索过的点应当删除。删除时,为了不将其它笔划断开,应考虑连线点的问题。以某点为中心,其左上和右下的点,左下和右上的点,上方和下方的点,左方和右方的点,均称为此点的连线点。删除某点时,如果此点有不在搜索方向上的连线点,则不删除,以免使其它笔划断线。总的搜索方向是从上排到下排,在一排中是从左到右。

三、由点阵字库向矢量字库的转换

把一个点阵汉字转换为一个矢量汉字的步骤如下:

1、首先从字库 HZK40F(以仿宋字体为例)中取一个汉字的点阵数据,共 180 字节。

2、将汉字点阵的数据放入一个 40×36 的数组中。一个位置有点,则数组中相应的位置上的数据为 1,否则为 0。

3、根据总的搜索原则先找到点阵中一点(即数

组中为 1 的元素)作为一笔的起点,记下它的坐标。如果此点没有除左右以外的连线点,则删除此点。

4、从此点开始按四个方向(左下、右下、右、下)搜索,若都没有点,则为孤立点,其起始坐标就是终点坐标。记下终点坐标。

5、如果四个方向中某一方向上有点,则以此方向为搜索方向继续搜索前进。如果新点除搜索(方向外没有连线点则删除此新点,并继续搜索直到沿此方向无点时即为一笔的终点坐标。记录此终点坐标。

6、重复 3—5 步,直到数组元素全部为 0。把各笔起终点坐标存入矢量字库中。

由于没有优化,根据以上算法转换的汉字笔划较多,如黑体字一级字库中笔划超过 132 笔的就多达一百以上。为了减少笔划(实际也是减少矢量字库的容量)采取了如下优化算法:在某点的四个搜索方向中,以最长的一笔作为有效笔,记录其起终点坐标。这样就使转换后的汉字笔划大为减少。

四、矢量汉字库的构成

我们转换的点阵字库是 40×36 点阵,其每笔坐标均在 0~39 之间。为了尽量减小矢量汉字库,采用了压缩方式存储各点坐标。我们知道,在高级语言中整型数仅占 2 个字节,其范围是 -32768~32767。而 $40 \times 40 \times 40 = 64000$ (65536 所以我们可以把三个坐标数放于一个整型数内,即 $G_0 = 1600 \times G_1 + 40 \times G_2 + G_3 - 32767$ 。另外,我们还对四种字体的一、二级字库中的所有汉字的笔划进行了统计,结果如附表。

根据统计结果,我们可以按照自己的需要确定矢量字库中一个汉字所占的字节数。比如,我们仅需要一级宋体矢量字库,则矢量字库中每记录(一个汉字)的字节数如下确定:由于没有多于 108 笔的汉字,因而选最大笔数为 108,每笔起终点坐标共 4 个数据,每三个数据放入一个整型数内,所以 $108 \times 4 \div 3 \times 2 = 288$ 这就是矢量字库每记录的字节数,整个矢量字库为 $288 \times 40 \times 94 = 1082880$ 字节,即不到 1.1MB。

五、矢量汉字库在绘图仪上的使用

矢量字库在绘图仪上使用,可以用各种方法输入汉字,程序自动找到所对应的矢量汉字记录,读出后转换为每笔的四个坐标,即可在绘图仪上写出

漂亮的汉字。该字库可用于各种绘图仪。

程序均用 BASIC 语言编写,已在 IBM-PC 及其兼容机上通过。

程序一是字库转换程序,将宋体一级字库转换为矢量字库,每个汉字的笔划不超过 108 笔。也可以按自己的需要更换点阵字库名和选用适当的最大笔划数。

程序二是在 SPL-400 绘图仪上写汉字的程序,其矢量字库的名字和记录长度应与程序一相对应,写字时如需放大或旋转,只需增加相应的绘图仪命令即可。

附表

笔划范围	(60	60-71	72-83	84-95	96-107	108-119)119
字库名称							
楷体一级字库	1201	1220	929	370	38	2	0
楷体二级字库	802	756	878	449	113	10	0
黑体一级字库	824	741	883	687	422	159	44
黑体二级字库	658	418	607	603	427	223	72
仿宋一级字库	2800	774	171	15	0	0	0
仿宋二级字库	1875	788	301	43	1	0	0
宋体一级字库	2276	879	428	147	30	0	0
宋体二级字库	1496	783	472	196	55	5	1

程序一

```

5 REM ZKSC40B 40 点阵字库转换为矢量字库
10 DERINT I=N
20 DIM Z%(39),IGI(432),IFXS(3)
30 OPRN"R",#1,"HTLIB40F",288:FIELD #1,288
   AS DIF%
40 OPEN"R",#2,"HZK40F",180:FIELD #2,180 AS
   D2%
70 M=0:CLS:LOCATE 5,10:PRINT"第 字,共 笔":
   LOCATE 7,10:PRINT"共有 字超笔划"
80 FOR N=1 TO 3760
90 GET #2,N:Z%(0)=STRING$(36,"0"):FOR I=1
   TO 39:Z%(I)=Z%(0):NEXT I
100 FOR I=1 TO 36:FOR J=2 TO 1 STEP-1:L=ASC
   (MID$(D2%,(I-1)*2+J,1))
110 FOR K=8 TO 1 STEP-1:L1=L AND 1:L=L\2
120 IF L1=1 THEN MID$(Z%(8*(J-1)+K-1),I,
   1)="1"
130 NEXT K,J,I
140 FOR I=1 TO 36:FOR J=3 TO 1 STEP-1:L=ASC
   (MID$(D2%,(I-1)*3+J+72,1))
150 FOR K=8 TO 1 STEP-1:L1=L AND 1:L=L\2
160 IF L1=1 THEN MID$(Z%(16+8*(J-1)+K-
   1),I,1)="1"
170 NEXT K,J,I:FOR I=1 TO 432:IGI(I)=0:NEXT
   I
176 GOSUB 2000:FOR I=1 TO 143
180 MID$(D1%,I*2+1,2)=MKI$(1600*IGI(3*I

```

```

+1)-32767+40*IGI(3*I+2)+IGI(3*I+3))
190 NEXT I:PUR #1,N
200 LOCATE 5,12:PRINT MID$(STR$(N)+" "2,4)
210 LOCATE 5,22:PRINT MID$(STR$(NUB)+" "2,
   3):IF NUB>108 THEN M=M+1
220 LOCATE 7,14:PRINT MID$(STR$(M)+" "2,3)
280 NEXT N
290 CLOSE
300 END
1100 REM 能否前进
1110 G%= "1"
1120 ON IDD+5 GOTO 1140,1180,1160,1280,1300,
   1200,1240,1260,1220
1140 IU=IU+1,GOTO 1300:REM X
1160 IU=IU+1:IV=IV+1,GOTO 1300:REM YX
1180 IV=IV+1,GOTO 1300:REM Y
1200 IU=IU-1,IV=IV+1,GOTO 1300:REM YS
1220 IU=IU-1,GOTO 1300:REM S
1240 IU=IU-1:IV=IV-1,GOTO 1300:REM ZS
1260 IV=IV-1,GOTO 1300:REM Z
1280 IU=IU+1:IV=IV-1:REM ZX
1300 IF (IU>39) OR (IV>35) OR (IU<0) OR (IV<0)
   THEN G%= "0" ELSE F%=MID$(Z%(IU),IV+1,1)
1320 RETURN
1500 REM 前进一步
1510 IV=IB:IU=IA:IDD=ID,GOSUB 1100:G1%=G
   %
1520 IF G1%="1" THEN IA=IU:IB=IV
1530 RETURN
1600 REM 删除点
1610 G2%="1"
1620 FOR ID1=-4 TO 4
   1630 IF (ID1=0) OR (ID1=ID) OR (ID1=-ID)
   THEN 1680
1640 IU=IC1:IV=IC2:IDD=ID1,GOSUB 1100:IF G%
   ="0" THEN 1680
1650 IU=IC1:IV=IC2:IDD=-ID1,GOSUB 1100:IF
   G%="1" THEN G2%="0":ID1=10
1680 NEXT ID1
1690 IF G2%="1" THEN MID$(Z%(IC1),IC2+1,1)
   = "0"
1700 RETURN
1800 REM 输出一笔
1805 IF NUB>107 THEN 1812
1810 IGI(1+4*NUB)=I:IGI(2+4*NUB)=J:IGI(3
   +4*NUB)=IA:IGI(4+4*NUB)=IB
1812 NUB=NUB+1
1820 RETURN
2000 REM 处理一个字
2010 NUB=0

```



```

2020 FOR I=0 TO 39;FOR J=0 TO 35
2040 IF MID$(Z$(I),J+1,1)="0" THEN 2200
2050 ID=-3;IC1=I;IC2=J;GOSUB 1600;FOR ID=0
    TO 3;IFXS(ID)=0;NEST ID
2060 FOR ID=-4 TO -1
2080 IA=I;IB=J;GOSUB 1500
2090 IF G1$="0" THEN 2122
2100 WHILE G1$="1"
2110 GOSUB 1500
2120 WEND
2122 IF ABS(I-IA)>ABS(J-IB) THEN IFXS(ID+4)
    =ABS(I-IA) ELSE IFXS(ID+4)=ABS(J-IB)
2124 NEXT ID
2126 ID=-4;FOR IFX=1 TO 3;IF IFXS(IFX)>IFXS
    (0) THEN ID=IFX-4;IFXS(0)=IFXS(IFX)
2128 NEXT IFX
2130 IF IFXS(0)=0 THEN 2180
2134 IA=I;IB=J;GOSUB 1500
2140 IF G1$="0" THEN 2180
2150 WHILE G1$="1"
2160 IC1=IA;IC2=IB;GOSUB 1600;GOSUB 1500
2170 WEND;GOSUB 1800;GOTO 2200
2200 NEST J,I
2210 RETURN
程序二
5 REM HTHZ40
10 DEFINT I-N
20 DIM N(432)
30 OPEN "R" #1;"HTLIB40S";320;FIELD #1,200 AS
    D1$,88 AS D2$
90 LPRINT "J1T1&4,4,1";CHR$(13)
100 IC=100;IR=200
110 INPUT"输入汉字";HZ$;JHZ=LEN(HZ$)/2
120 FOR IHZ=1 TO JHZ
130 IQ=ASC(MID$(HZ$,2*IHZ-1,1))-&HAO;
    IW=ASC(MID$(HZ$,2*IHZ,1))-&HAO
140 JL=94*(IQ-16)+IW
220 GET #1,JL
224 FOR I=0 TO 99;Z=CVI(MID$(D1$,1+2*I,
    2))+32767;N(I+3*I)=FIX(Z/1600);N1=Z-N
    (1+3*I)*1600;(N(2+3*I)=N1\40;N(3+3*
    I)=(N1 MOD 40);NEXT I
226 FOR I=0 TO 43;Z=CVI((MID$(D2$,1+2*I,
    2))+32767;N(301+3*I)=FIX(Z/1600);N1=Z-
    N(301+3*I)*1600;N(302+3*I)=N1\40;N(303
    +3*I)=N1 MOD 40;NEXT I
230 FOR K=0 TO 107
250 IY0=N(2+4*K);IX0=39-N(1+4*K)
260 IY1=N(4+4*K);IX1=39-N(3+4*K)
287 IX0=IX0+IC;IY0=IY0+IR
290 LPRINT"M";STR$(IY0);"STR$(IX0);CHR
    $(13)
297 IX1=IX1+IC;IY1=IY1+IR
300 LPRINT"D";STR$(IY1);"STR$(IX1);CHR
    $(13)
310 NEXT K
320 IC=IC+100/2
360 NEXT IHZ
370 LPRINT"J0";CHR$(13)
380 CLOS
390 END

```

武侯计算机网络工程公司开展微机系列合同保修业务

本公司前称西部电脑维修公司,拥有雄厚的软硬件技术力量,专门从事硬件维修多年,具有丰富的实践经验,在广大用户的要求和科委的支持下,现特开展计算机保全维修业务,具体办法如下:

一、保全维修办法

1. 由我公司维修技术人员上门办理保修手续。
2. 单位委修的机器,由我公司技术人员检查、加封并双方认可后,办理保全维修手续。
3. 凡参加保全维修的机器及设备,采用一次性缴费的办法,价格合理,服务周到,用户满意。
4. 我公司开展一年、二年两种保全维修业务,期满后在开展续保业务。

二、保全维修方式

1. 对参加保全维修的客户,我公司将提供优质服务,定期、分段对客户机器设备进行保养,并提供技术咨询和完善使用功能。
2. 成都市内上门服务,凡是送到我公司的机器,主机二天内交货,外设三天内交货,特殊情况(如停电,需特殊器件等)例外。

3. 参加保修的机器如因自然灾害或人为损坏,酌情处理。

4. 凡本公司售出的机器,免费保修两年。

开户行:工商行跳伞塔分理处 帐号:67012804 户名:成都武侯计算机网络工程公司 地址:一环路南二段11号
邮编:610013 电话:(028)——581252(昼) 580290(夜) 联系人:谢 军 况光名

FOXBASE 通用窗口式数据库文件装载器

江苏省镇江市经济信息中心 张 懿 (212001)

摘要 鉴于目前多数管理信息系统无文件装载功能,本文讨论了如何充分利用 DOS 提供的输入输出改向以及 Foxbase 的数组等功能,编写一个通用数据库文件装载程序的原理。

关键词 窗口 文件装载 数据库

软件系统的“人—机”界面设计得是否合理、美观和友好,将直接地影响到系统的效率和操作人员的兴趣。信息管理系统,尤其是那些管理着大量结构相同或不相同的数据库文件的系统(如商品供求信息处理系统),在进行查询、浏览或其它一些操作时,需要频繁地改换被操作数据库。有时被操作库并非在一个目录路径下,而是分别存放在若干个其它目录路径下。因此,库名的键入会使人感到很烦琐,虽然在一定程度上,能以只使用某些特定路径或按规则给库取名的办法解决,但这是以降低程序的灵活性为代价的。

在一些优秀的软件中,如 Pctools、Turbo 系列的 Pascal、Basica、C 等及一些排版系统中都有一个文件装载窗口,用户可→、↓、↑、←等移动光条,选中后按(Enter),则选点文件就被装入进行处理;按(ESC),则退出。使用文件装载系统有如下优点:

1. 用户一般无需键入文件名;
2. 用户不知道目录中有何文件时,也可以先浏览后选择;
3. 不致出现“File not found”之类的情况,因为窗口中列出的文件都是存在的;
4. 加上库结构合法性检查程序段,可以防止数据库结构与系统不匹配的错误出现。

目前多数管理信息系统无文件装载功能。那么是否能不借助于其它诸如:C、Pascal 之类的高级语言,仅利用 DOS 的输入输出改向功能和 Foxbase 提供的语言成分,开发一个界面友好、功能完善的窗口式数据库文件装载器(程序)呢?实践证明完全是办得到的。下面即是开发一个 Foxbase 通用窗口式数据库文件装载器的一般主要步骤:

首先,利用 DOS 的输入输出重定向功能(而非一般利用屏幕信息文件的方法,速度更快),即通过! dir\path1\...\pathn*.dbf>filename.txt 生成文件名文本文件(Foxbase 中的 dir 或 dir *.dbf 在此不适用,因其将文件名分屏显示,中间需人工干预,且速度也较慢)。

其次,创建一个只含一个字段且字段长为 12 的库,并用 Appen from filenamesdf 向库中追加文件名

记录,删除无关信息,并对“非法路径”、“文件不存在”等情况作预处理。

然后,用一个二维数组作为文件名库→窗口显示之间的缓冲。构造算法,使得对窗口的翻页、上滚、下滚等操作转化为:对数据库记录指针的正确定位,进而取记录并对显示缓冲数组刷新,同时也对屏幕刷新:所谓选中窗口某一文件,也即是选中对应数组元素(库文件名)。

另外,还需记录最近一次操作现场,使得下次运行时能自动重现上次的选择情况,我们可以使用变量文件来达到这个目的。

下面是一个具体实现的例子,该装载程序由 loadfil.prg 和 selefil.prg 两个程序块构成,为便于阅读和说明,已用分隔线对程序中某些相对独立的部分作了划分。

(一)Loadfil.prg

1. 从 11 行—34 行,定义变量并赋值,其中 M₀, N₀ 为显示缓冲数组的行列数。也即屏幕上文件名列表区的行列数。

2. 36—41 行 调入记录前次选择情况的变量文件,如其不存在,自动将路径初始化为 C 盘根目录,文件名初始化为空白。

3. 54—63 行 生成文件名文本文件。

4. 66—101 行 生成文件名库,并对各种情况作预处理。

5. 104—124 行 显示装载投影窗口及有关信息。

6. 126—136 行 计算文件名库中的记录数是否为显示缓冲数组元素个数的整数倍,不是则添加空白记录,使之成整数倍,以便于处理;lnb 为库中记录总数除以 N₀。

7. 141—151 行 对文件名库记录指针定位,取记录对显示缓冲数组刷新。

8. 153 行 运行键码处理程序。

9. 154—178 行、209—224 行对库文件被选中的情况作处理。其中 162—164 行、215—217 行保存选择情况。

10. 186—205 行 路径输入情况的处理。

(二)键码处理程序 selefil.prg

1. 4—13行 在文件名列表区显示文件名。

2. 50—52行 初始化光条。

3. 53—147行 是选择区键码处理模块,由于使用了宏代换和IF语句,使得此段结构极为清楚、高效。

当前目录路径:\cpx\data[24个库文件]

数据文件名:WJ40.dbf [按↓选择]

ZXF01.dbf ZXF.dbf WORK46.dbf WJ46.dbf
WORK47.dbf WJ47.dbf WORK48.dbf WJ48.dbf
WORK49.dbf filename.dbf WJ49.dbf WJ40.dbf

图1

运行该装载程序后,屏幕将出现如图1的立体投影窗口,该窗口分为三个区域,从顶到底分别是目录区、文件名区(同时也作提示信息区)和当前路径库文件名列表区。

当前目录路径:c:\cpx\data[57个库文件]

数据文件名:WORK72.DBF[按↓选择]

WORK60.dbf WORK61.dbf WORK62.dbf WORK63.dbf
WORK64.dbf WORK65.dbf WORK66.dbf WORK67.dbf
WORK68.dbf WORK69.dbf WORK70.dbf WORK71.dbf

图2

此时,光标定位于文件名区,该区所显示文件名是上次选中的库文件名,若想继续对该库操作,按回车即可,若此次要操作的库与上次不同,则操作者有两种选择:一是在此输入库文件名(带不带扩展名均可),若库不存在,则光标自动定位到文件名列表区,二是按↓键直接进入文件名列表区选择。此时操作者可用↑、↓、→、←、Pgdn、Pgup移动光条、滚屏或翻页前后搜索(Home、End的定义,使操作人员可以直接到首屏和末屏,这对于有大量库的情形尤感方便),选中后按<Enter>即可(如图2所示)。当光条在窗底时,按↓窗口显示内容上滚一行;在列表区的顶行时,按↑则会出现两种情况:如果当前不是第一屏,则选择区内容下滚一行,反之,则将进入文件名区(此时按↓仍可回到列表区),此时,如再按↑则光标进入目录区,该区被清为空白,输入合法的目录路径名,按回车则系统将该路径下的库文件列表,光标再次定位在文件名区,若该路径下无库文件,则显示如图3,如果路径非法,则在文件名区提示“路径错,按↑重输”

当前目录路径:\123[0个库文件]

提示:无库!-[按↑重选]

图3

在目录区和列表区按<Esc>均可退出装载。
用户实际应用该装载程序可有两种方式:

第一种:在用户需录入文件名的地方,直接将其作为用户的一个子程序来调用(此时应删除程序清单中加星号的语句),用户也可定义某个“热键”,以便随时调用该程序了解其它路径下的库文件情况。

第二种:删去清单中155—156行(加&&方式1)、209—224行,解除清单中的“*”号。把用户的实际处理程序作为一个子程序在此被调用(如果可能的话)。这对当某一路径下有大量结构相同的数据库,并需频繁更换被操作库时,尤其高效。

该窗口式数据库文件装载机(程序)界面友好、功能完善,其窗口大小可变,并能随意定位。经在“镇江市产(商)品供求信息联机管理系统”上使用,证明效果极佳,大大地提高了工作效率,并减轻了工作人员的工作强度。

由于未借助于其它诸如:C、PASCAL之类的高级语言,所以,这无疑是一种全新的成功尝试,使用该程序可以提高管理信息系统界面的档次。

Loadfil. prg

* FOXBASE 通用窗口式数据库文件装载机

* 作者:镇江市经济信息中心 张 懿

set talk off

set stat off

set scor off

set safe off

set inten off

publ row0,col0,1,c,tot-file

publ Cur-R, Cur-L, Cur-Up, Cur-Dn, End, Home, Escape, Sfile, No-file

publ path-name, last-name, exist-old

Cur-r=4

Cur-l=19

Cur-up=5

Cur-dn=24

End=6

Home=1

Escape=27

Cr=13

PgUp=18

PgDn=3

m0=3

n0=4

publ job(m0,n0)

rowb=3

row0=rowb+2

col0=14

l=1 c=1

no-file=0

enter-name=.t.

exist-old=.f.

-line 35-


```

if file('path-name.mem')
restore from path-name addi
else
init-path='c:'
init-name=repl(' ',12)
endif
* —line 42— *
okl=1
lp-no=.t.
do while okl>0
if lp-no
path-name=init-path
last-name=init-name
sfile=last-name
lp-no=.f.
endif
set colo to
clea
disk=sys(5)
st-file='filename.txt'
path-name=trim(path-name)+'\
exten='*.dbf'
if file('filename.txt')
! dir & path-name& exten>& disk& st-file
else
! dir & path-name& exten>>& disk& st-file
endif
* —line 63— *
clea
sele 10
use filename
zap
appe from filename sdf
* —————
go top
do case
case recc()=2&& invalid directory
tot-file=0
dele for recno()=2
pack
repl name with repl(' ',12)
last-name=repl(' ',12)
mag0='提示:路径错'
msg='[按↑重输]'
case recc()=4 && file not found
tot-file=0
dele for recno()<=3
pack
repl name with repl(' ',12)
last-name=repl(' ',12)
msg0='提示:无库!'
msg='[按↑重选]'
other && file exist
dele for recno()<=4
go bott
dele

```

```

pack
do while .not.eof()
t-name=substr(name,1,at(' ',name)-1)+'*.dbf'
repl name with t-name
skip
enddo
tot-file=recc()
msg0='数据文件名:'
msg='[按↓选择]'
endcase
* —line 102— *
set colo to /bg+
@rowb,col0 clea to row0+m0+m0+1,col0+n0*12+n0
set col0 to/rb+
@rowb-1,col0-1 clea to row0+m0+(m0-1)+1,col0
+(n0)*12+n0-1
set colo to w+/b
* —————
@ rowb,col0+4 say'当前目录路径:'
@ row(),col()+1 say sturr(path-name,len(path-
name),1,'')
@ row(),col()+1 say '['
do case
case tot-file<10
ll=1
case tot-file>=10.and .tot-file<100
ll=2
case tot-file>=100.and .tot-file<1000
ll=3
endcase
@ row(),col()say int(tot-file) pict repl('9',LL)
@ row(),col()say'个库文件]'
@ rowb+1,col0+4 say msg0+last-name pict repl('x',
24)
@ row(),col0+30 say msg
* —line 124— *
go top
if mod(recc(),m0*n0)<>0
tnb=int(recc()/(m0*n0))+1
i=1
do while i<=mod(recc(),m0*n0)
appe blank
i=i+1
enddo
else
tnb=int(recc()/(m0*n0))
endif
lnb=tnb*m0
* —line 137— *
lb=1 && lb={1—lnb}
ok2=1
do while ok2>0
go (lb-1)*n0+1
i=1
do while i<=m0

```

```

j=0
do while j<n0.and..not.eof()
j=j+1
job(i,j)=name
skip
enddo
i=i+1
enddo
* —line 152— *
do selefil
if exist—old
ok1=0 && 方式 1
exit && 方式 1
* if no —file=0
* path —name=trim(path—name)
* last—name=sfile
* use & path—name& sfile
* endif
* init —path=stuff(path—name,len(path—name),1,"")
* init—name=last—name
* save to path—name all like init—????
* exist—old=.f.
* set colo to w/b
* @ rowb+1 ,col0+17 say last—name
* -----
* 保存屏幕语句
* @ 24,0,clea && 此下 4 行是示范!
* @24,25 say' 数据库:'
* @ row (),col() say trim(path—name+sfile)
* @row (),col()+1 say' 被选中!'
* 恢复屏幕语句
* -----
* sele 10
* use filename
endif
* —line 179— *
enddo && ok2=0
if ok1=0
exit
endif
* -----
l=1 c=1
do while .t.
path—name=repl(' ',30)
set colo to w/b
@ rowb,col0+4 say' 输入目录路径:'
get path—name pict repl('x',23)
read
rk=mod (readkey(),256)
do case
case rk=12
ok1=0
exit
case rk=15.or.rk=16
if path—name=repl(' ',30)
loop
else
exit
endif
other
loop
endcase
enddo
enddo && ok1=0
* —line 208— *
if exist—old
if no—file=0
path—name=trim(path—name)
last—name=sfile
use & path—name & sfile
endif
init—path=stuff(path—name,len(path—name),1,"")
init—name=last—name
save to path—name all like init—???
exist—old=.f.
set colo to w/b
@ rowb+1,col0+17 say last—name
@24,25 say,' 数据库:'&& 此下 3 行示范
@ row(),col() say trim(path—name+sfile)
@ row(),col()1 +say' 被选中!'
endif
retu
* * * * *
* selefile.prg:select file with cursor *
* * * * *
set colo to w/b
i=1
do while i<=m0
j=1
do while j<=n0
@ row0+i+(i-1),col0+(j-1)*12+(j-1)say job(i,
j)
j=j+1
enddo
i=i+1
enddo
ok3=1
do while ok3>0
if enter—name
do while .t.
@ rowb+1,col0+4 say msg0;
get last—name pict repl('x',12)
read
rkey=mod(readkey(),256)
do case
case rkey=5
lb=1
exit
case rkey=4
ok2=0 ok3=0
enter—name=.t.
last—name=repl(' ',12)

```

```

retu
case rkey=15. or. rkey=16
if file('&path -name& last -name. dbf');
. or. file('& path-name & last-name')
sfile=last-name
exist-old=. t.
ok3=0
exit
endif
lb=1
exit
other
loop
endcase
enddo
enter-name=. f.
endif
if exist-old
exit
endif
* -----
set colo to w/r+
view="@ row0+1+(l-1),col0+(C-1)*12+(c-1)
say job(1,c)"
& view
clear typeahead
ok4=1
do while ok4>0
keyin=inkey(0)
* -----
set colo to w/b
& view && char=white,base=blue
set colo to w/r+
* -----
do case
case keyin=Cur-R
c=iif(c<n0,c+1,1)
& view
case keyin=Cur-L
c=iif(C>1,C-1,n0)
& view
case keyin=Cur-Dn
if l=m0
& view
lb=iif((lnb-lb)>m0,lb+1,lnb-m0+1)
enter-name=. f.
ok4=0 ok3=0
exit
endif
l=iif(l<m0,l+1,m0)
& view
case keyin=Cur-Up
if l=1
set colo to w /b
& view
if lb-1>0
lb=lb-1
ok3=0 ok4=0
enter-name=. f.
else
lb=1
ok4=0
enter-name=. t.
endif
exit
endif
l=iif(l>1,l-1,1)
& view
case keyin=PgUp
lb=iif(lb-m0>0,lb-m0,1)
l=1 c=1
ok4=0 ok3=0
exit
case keyin=PgDn
lb=iif(lb<(lnb-2*m0+1),lb+m0,lnb-m0+1)
l=1 c=1
ok4=0 ok3=0
exit
case keyin=Home
lb=1 l=1 c=1
ok3=0
exit
case keyin=End
lb=lnb-m0+1
l=1 c=1
ok3=0
exit
case keyin=cr
& view
sfile=job(l,c)
if. not. file('& path-name& sfile')
loop
endif
ok3=0
exist-old=. t.
no-file=0
retu
case keyin=Escape
ok3=0 ok2=0 ok1=0
no-file=1
exit
other
& view
loop
endcase
if keyin=cr
ok2=0
endif
enddo && ok4=0
enddo && ok3=0
retu

```

一种易于维护的 FOXBASE 通用下拉式菜单

西安交通大学工程力学系 曹玉俊 刘路放(710049)

在数据库应用系统中使用中文下拉式菜单具有使用方便,屏幕格式漂亮等优点。FOXBASE 虽然有以下下拉式菜单语句,但是由于汉化的原因,并不好用。因此通常由开发者自己编写菜单程序。但自己开发的菜单程序普遍存在着显示速度与维护方便难以兼顾的问题。为了提高显示速度,有人采用汇编语言或 C 语言编写菜单程序,但是汇编语言(或 C 语言)、FOXBASE 和汉字系统之间的兼容性和调用方式比较复杂、通用性差,修改、增删菜单项目时不太方便,而且容易出错。在使用 FOXBASE(或 DBASE)编写菜单程序时,为了提高菜单的可维护性,通常采用建立菜单数据库的方法,但其最大缺点是因为检索菜单库致使菜单显示速度太慢,影响系统性能。为了解决这一问题,笔者使用汉化的 FOXBASE 2.10 编写了一个通用下拉式菜单,其基本思想是:

1. 采用菜单数据库以方便维护;
2. 利用内存变量数组将菜单装入内存,以提高显示速度。

本文介绍的通用下拉式菜单程序利用了 FOXBASE 中的内存变量数组定义语句 DIMENSION 和存贮及恢复内存变量语句 SAVE TO 和 RESTORE FROM 实现上述设想,因此综合了菜单数据库维护方便和内存变量数组显示速度高的特点。

实现时,通过使用一个转换程序建立内存变量文件,实现菜单数据库到内存变量数组之间的转换。

菜单数据库的结构如下:

- | | |
|---------|-------------------|
| 1、菜单项目名 | C20 (以下宽度均可调) |
| 2、程序模块名 | C8 |
| 3、标志 | C5 (反映菜单项之间的调用关系) |

菜单数据库的内容示例:

菜单项目名	程序模块名	标志
数据输入		C1
核算成本		C1
销售及利润核算		C1
打印报表及凭证		C1
输入材料成本差异率	ATE	C21
输入动力消耗数据		C21
输入产品数量	AMOUNT	C21

生成成本计算表一	COST 01	C22
生成成本计算表二		C22
生成成本计算表三	CST3	C22
生成商品成本表	GOODS	C22
结转产品销售	JZSALE	C23
结转其他销售	JZ_XS	C23
成本计算表一	COST01P	C24
成本计算表二		C24
成本计算表三	CST3P	C24
商品成本表	GOODSP	C24
一车间成本计算	COST 0201	C3202
二车间成本计算	COST 0202	C3202
三车间成本计算	COST 0203	C3202
一车间	CST21P	C3402
二车间	CST24P	C3402
三车间	CST25P	C3402

标志域第二位数表示菜单级别,第三位表示对应的一级菜单位置,第四、五位表示对应的二级菜单位置。例如 C3402 表示这一组菜单是三级菜单,对应的一级菜单是标志域为 C1 的第四个菜单项目,对应的二级菜单是标志域为 C24 的第二个菜单项目。这种菜单数据库能全面反映菜单结构以及菜单项目与程序模块的对应关系。

转换程序 CRTM.PRГ 把菜单数据库 ME NU.DBF 的信息转换成内存变量,并存放在文件 ME NU.MEM 中。主控菜单程序 C.PRГ 通过恢复 MENU.MEM 中的内存变量(及内存变量数组)将菜单全部装入内存,实现菜单高速显示。这样当增删菜单项目时只需要增、删、改菜单数据库中的记录,然后运行一次 CRTM.PRГ 即可。而菜单数据库中菜单项目与程序模块的对应关系简单明了,不易出错。转换程序 CRTM.PRГ 还能自动检测菜单结构中的越界错误,提示维护者及时修改。

本菜单系统的菜单结构约定如下:

- 一级菜单项目数 (=9)
- 二级菜单项目数 (=18)
- 三级菜单项目数 (=15)

本菜单系统除了具有维护方便的特点外,还具有结构简洁、清晰,易于扩充及通用性强的特点,稍加修改还可以增加菜单级数。

程序清单略去,需要者请同作者联系。

在西文 DOS 操作系统下显示汉字文本文件

安徽大学电子工程与信息科学系 明 军 (合肥 230029)

在西文 DOS 操作系统下,不能直接显示汉字文本文件,而在汉字操作系统下,一些较大的文件由于内存不够,又不能运行,给使用者带来了不便。为此,笔者模仿西文 DOS 的 TYPE 命令,编写了一个在西文 DOS 操作系统下显示汉字文本文件的程序。源程序由 TURBO C 编写,起名 HZTYPE.C,参见源程序清单。该程序使用与西文 TYPE 命令类同,同时也允许直接进入程序后输入文件名。此外,TYPE 命令对于输入文本一行超出 80 个字符的数据,另起一行显示,造成屏幕显示紊乱,特别是一些表格,很难了解原文本的内容。本程序对一行超过 80 个字符以后的数据,采取不再显示的方法。另外程序在每显示满一屏暂停,由使用者选择“继续”还是“退出”。程序设置显示画面为蓝色底色,绿色前景,显得更为美观。程序在 286 微机上通过,使用效果良好。

下面对程序编写原理作简要说明:

1. 汉字显示必须选择图形方式,为使屏幕能提供更多的信息量,程序对屏幕设置了 VGA 的高分辨率方式,水平为 640 点,垂直为 480 点。汉字库选择了 2.13A 中的 HZK16,为 16×16 点阵。行间间隔留有 4 个像素点,故一行可显示 40 个汉字,一屏可写 24 行。

2. 屏幕写方式有两种。一种是通过标准函数直接在屏幕上绘点,程序编写简单,但显示速度较慢;另一种方式是直接向屏幕对应的视频缓冲区送数据,需要了解计算机的视频缓冲区地址,程序也较简单,而且显示速度较快。本程序选择了第二种方式。VGA 方式的视频缓冲区在内存的首地址为 A0000:0000,一个字节对应屏幕上水平方向顺序八个像素点。

3. 汉字文本文件是以内码的形式存放的,其中汉字由两个内码组成,特征是每个内码均大于 A0H,设为内码 1 和内码 2。内码与区位码的对应关系为:区码=内码 1-A0H,位码=内码 2-A0H,

区位码=区码×100+位码。

当程序判断输入字符为汉字时,可根据上式得到对应区位码。

4. 程序选择的 HZK16 汉字库的字模,在汉字库内部存放的格式见图 1,共为 32 个字节。图 2 是

汉字“啊”的点阵图。每个汉字存放的顺序与区位码编号一致,由于区位码规定每个区为 94 个汉字,且位码和区码编号均由“1”打头,所以区位码在汉字库中的首地址为:32×[(区码-1)×94+(位码-1)]。因此,显示汉字时,应按区位码在字库中的首地址顺序取 32 个字节,写入对应的视频缓冲区地址。

5. 程序对文本中的控制字符,仅保留了对换行符 0AH 和制表符 09H 的处理。其中对制表符的处理,选择了使用较多的八个空格符号 20H。对于西文字符,程序直接调用 ROM BIOS 中的字库发生表。该字库发生表包含了一类 ASC 字符(00H 至 7FH)的点阵图,首址为内存 F000:FA6E,末址为 F000:FE6D。每个字符由顺序八个字节组成。因此,字符对应的相对首地址为:8×字符代码。此外,西文 8×8 点阵在高分辨力屏幕上显示的字形较小,程序将每一行用两行写入,即一个西文字符对应 16×8 点阵。

6. 以上说明涉及的参数,读者可根据自己的需要重新选择。特别是汉字库,可调用自己硬盘原有的汉字库,不一定使用本程序中的 HZK16 汉字库,增加硬盘的开销。但各种汉字库的存放略有不同,请注意改变程序中的对应参数。

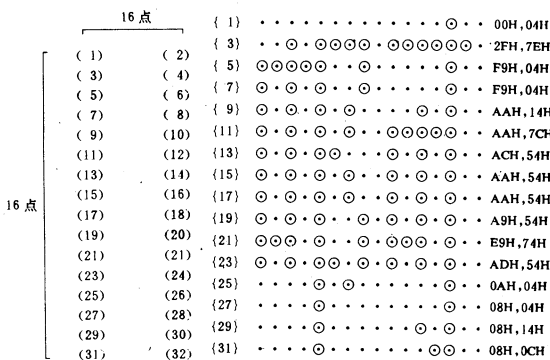


图 1 HZK16 的内部排列顺序 图 2 HZK16 中汉字“啊”的 16×16 点阵图

源程序清单:

```
#include <graphics.h>
#include <stdio.h>
unsigned char far * vga16-addr=(unsigned char far *) 0xa0000000;
unsigned char far * asc-addr=(unsigned char far *) 0xf000fa6e;
main(argc,argv)
int argc;char * argv[];
```

微机普通显示器图像处理系统概述*

甘肃省计算中心 午锁平 薛和华 刘叶梅 (730030)

摘要 利用普通微机和显示器为硬件背景(如常见的 286、386、486 等机)通过纯软件实现图像处理功能。既可在屏幕上恢复出逼真的各种图像(包括遥感图像的恢复显示);又可对各种图像进行放大、缩小,增强、分类、局部复改等处理工作;还可以在打印机上拷贝出灰度值为 16 层的黑白图像和彩色图像。

关键词 图像处理 CCT 磁带 CCD 摄像机 TM 数据 灰度值 伪彩色

一、图像数据的摄入

1. 利用遥感 CCT 磁带 TM 数据或 MSS 数据。通过中小型计算机的磁带机读取数据;然后转存在普通微机软磁盘上(硬盘也同样)。或者直接购买北京卫星地面站的遥感图像数据盘。笔者已在 VAX 机上读取了三盘密度为 1600 的 CCT 带 TM 数据,并利用此软件系统恢复出 40 余幅 512×512 个象元的天水片遥感图像**图 1(略)。

2. 通过扫描仪录入图像、图案、图样(如壁画、照片、地毯小样等)见图 2。

3. 通过摄像机录入图像数据(如 CCD 摄像仪)见图 3。

二、图像处理



波斯地毯小样

```
long int addr; unsigned int i, j, x, y, xx = 0, yy = 0, ww = 0;
unsigned char c, cc, c1 = 0, c2 = 0, ass[16][2]; * filename = "hxf-type";
FILE * fpr; * fpc; int driver = VGA; int mode = VGAHI;
fpc = fopen("hsk16", "rb");
if (fpc == 0) {printf("no cclib\n"); exit(0);}
if (argc > 2) {fpr = fopen(argv[1], "r");}
if (fpr == 0) {printf("can't open READ FILE\n"); exit(1);}
else {printf("input your filename to type:");
scanf("%s", filename); fpr = fopen(filename, "rt");}
if (fpr == 0) {printf("can't open READ FILE\n"); exit(1);}
initgraph(&driver, &mode, "bgi");
setbkcolor(1); outportb(0x3c4, 2); outportb(0x3c5, 10);
while ((fscanf(fpr, "%c", &cc)) != EOF) {
if (yy >= 480) {c = getch(); if (c == 'q') break; else {cleardevice();
outportb(0x3c4, 2); outportb(0x3c5, 10); xx = 0; yy = 0;}}
if (cc == 0x0a) {xx = 0; yy = yy + 20;}
if (xx >= 78) {continue;}
if (cc == 0x09) {xx = xx + 8;}
if (cc >= 0x20 && cc < 0x80) {addr = 8 * cc;
for (i = 0; i <= 7; i++) {x = xx; y = yy + 2 * i;
* (vgahi - addr + addr + x + y * 80) = * (asc - addr + addr + i);
* (vgahi - addr + x + (y + 1) * 80) = * (asc - addr + addr + i);
```

```
xx = xx + 1;}
if (cc >= 0xa0) {ww++; if (ww == 1) {c1 = cc;
else {c2 = cc; ww = 0; addr = 32 * ((long)94 * (c1 - 0xa0 - 1) + c2 - 0xa0 - 1);
fseek(fpc, addr, 0);
for (i = 0; i <= 15; i++) {for (j = 0; j <= 1; j++) {
ass[i][j] = fgetc(fpc); x = xx + j; y = yy + i;
* (vgahi - addr + x + y * 80) = ass[i][j];}}
xx = xx + 2;}}
}
getch(); fclose(fpc); fclose(fpr); colorsetgraph();
}
```

参考文献

- [1] 叶欣编译,《TURBO C 使用大全》,北京科海培训中心,1990.4.
- [2] 寇国华等编译,《DOS 使用大全》,电子工业出版社,1989.12.
- [3] 陈学谦等译,《IBM PC 编程指南》,电子工业出版社,1988.10.
- [4] 夏东涛编译,《MS-DOS 高水平程序设计》,电子工业出版社,1989.1.

1、预处理:对于 CCD 摄像仪和 CCT 磁带转录的数据均要经过预处理。既要剔除和删去文件的头、控制符和说明等部分字节,且要确定其图像的行、列长度和象元数。

2、图像的放大、缩小处理。放大缩小一次仅为横向、纵向各一倍,若要缩若干倍则可边缩边存盘,对缩过的再缩。

3、图像的分类和着色定义:即对图像数据灰度值 0~255 的范围进行分类(在 EGA 和 VGA 的显示卡上最大可分为 16 类,但每类区间可在 0~255 之间自定义选择,在 TVGA 卡上为灰度值 64 类,彩色为 256)。

4、图像局部替换、重存盘、加座标显示(用于地图等)、加汉字标注明及地名镶嵌汉字说明等。

5、多幅图像的拼接(横向);连接(纵向);图像的分割、取窗口等处理。

6、遥感数据图像处理(如图像增强、比值处理、合成、变换等处理)。

7、支持汉字系统下的各种处理操作和图像显示。

8、用于人事档案、公安文档、身份证等图像与文档组合显示的信息处理。

三、图像输出

1、打印灰度值(0~255 级符号或数值)或是被定义后的灰度分类值。

2、黑白或彩色打印机打印提花毛巾、地毯图案等轧版样(图略)。

3、硬拷贝灰度和彩色的花样稿(见图 2、3)。

4、屏幕照像(图略)。

四、本软件支持的软件环境

本软件支持的软件环境为 PC-DOS、MS-DOS、COMPAQ-PCDOS 等均可,均为 3.0 以上版本。汉字系统为 CCDOS、UPS、2.13 等汉字环境。软件由 turbo-PASCAL、汇编语言、C 语言编写。图像文档组合处理由 PASCAL 和 FOXBASE 编写。

五、源程序清原

由于篇幅所限略去,需要者,请同作者联系。

参考文献

[1]、《Turbo-pascal (4.0) 使用手册》,中国软件技术公司软件交易出版社,1988。

[2]、赵恒玉等编,《微型计算机图形处理基础及实用图形软件的开发》,航空工业出版社,1991。

[3]、陶伟等编,《EGA、VGA、TVGA 高级微机图形编程指南与实例》,北京希望电脑公司,1991。

[4]、黄智编著,《图象处理和识别实用程序库》,天津大学技术出版社,1989。

[5]美 R. A. schowengerdt 著,《遥感中的图像处理 and 分类技术》,科学出版社,1991。

[6]《中国遥感卫星地面站用户简讯》总字第二、三期。

* 本文是在甘肃省自然科学基金 ZR-90-50 项目支持下完成的。

** CCT 磁带 TM 数据是为甘肃省科学院地质灾害防治研究所进行的磁带数据处理的数据。



图 3 人物肖像

· 书讯 ·

《系统的互连与互通》

《系统的互连与互通》一书,详细探讨了组网技术从互连到互通的根本转变,深入介绍了当今世界的互连互通技术方法,内容涉及互连互通的基础,互连的结构模块及相关技术,互通的策略、方法及应用,国际上最新的互连互通动态等。该书对计算机、通信、网络等方面的工程技术人员和广大的计算机与通信设备的使用部门了解互连互通很有助益,也可作为高等院校师生及有关工程技术人员学习、设计的参考书,本书结构严谨,语言流畅。

全书近 30 万字。每册 22 元(含邮费)。需要此书者,请同本刊唐大利联系。

多用户帐务处理系统的设计与实现

抚顺铝厂计算机中心 齐英武(113001)

摘要 本文提出了多用户帐务处理电算化系统中数据结构的设计和控制机制的实现的一种新设计思想,它适用于各种多用户数据库管理系统。并澄清了修改控制机制中由于时序和对象的交错造成的各种混乱现象。

关键词 多用户 帐务处理 数据结构 控制机制

帐务处理工作做为会计核算工作的主体和核心部分,也是会计电算化的主要内容。因此对于大中型企事业单位的财务部门,由于总帐会计科目数量多、各类单据流量大,开发多用户帐务处理电算化系统势在必行。

多用户帐务处理系统的特点主要是:1、减少数据冗余,提高数据资源的使用效率;2、多站点多用户同时使用,提高会计业务的工作效率;3、数据统一存储管理,满足了财务数据对安全性、一致性等的特殊要求。

下面就从两方面来阐述多用户帐务处理电算化系统的总体设计中所涉及的问题。

一、数据结构的设计

1、设计思想:目前,我国及世界各国普遍采用复式借贷记帐法;即对发生的业务采用一借一贷的简单分录方式,或采用一借多贷、一贷多借的复合分录方式并据以登帐。实现帐务处理电算化后也必须遵守这个原则。由于这种记帐方法存在着平行登录、重复记帐的现象,这样衍生了两个问题:数据冗余、重复劳动;借贷方各自为政,对帐只能手工实现。

多用户系统可以象单机版本电算化软件那样建立总分类帐库。这样既满足了数据共享,又保证“数出一家”省去了对帐过程。但又产生后续问题:①多站点多用户可同时进行录入、修改、查询等操作,很难保证数据的完整性和一致性;②由于总帐会计科目数量多,多站点多用户同时操作总分类帐会产生“颈口现象”,导致系统运行效率降低。

如果采取牺牲空间换取效率的方法来解决上述问题。即单方登录、重复记帐方法,为每个总帐科目均单独设立帐簿,记帐凭证由单方登录、双方共有(登录方将凭证内容同时登录到借贷双方帐簿);帐簿分录仍由双方分别单独进行(但双方分录的明细科目各不一样)。这样既解决了平行登录的工作效率问题,又解决了对帐、结帐计算机半自动化问题。但仍没有解决由于总帐会计科目数量多,多站点多用户同时操作个别常用帐簿所出现“颈口现象”,导致系统运行效率降低,甚至对数据库交替操作造成死锁现象等问题。因此又进一步为每个帐簿设置两个外围凭证库:本方登录库和对方登录库。主要目的是

为了将凭证登录与帐簿分录在时序上区别开来,以解决“颈口”与“死锁”现象的出现。

2、实现方法:上述设置外围凭证库的设计思想的核心是帐簿与帐簿间完全隔离,仅通过进(I)、出(O)两个不同的外围凭证库来交换凭证。这样肯定会解决“死锁”问题,但对于个别常用帐簿的外围凭证库,由于多用户多站点同时进行录入、修改、查询、分录等操作仍会出现“颈口”现象。为此我们又采取下述方法来解决“颈口”问题。①另设置一临时过渡凭证库(d),将它置于外围凭证库(I,O)与帐簿之间。使得帐簿所有涉及凭证的操作(录入除外)都采用批处理方式。即首先将外围凭证库中满足操作条件的凭证拷贝到临时过渡凭证库中,进行各种处理后统一将结果再传回外围凭证库中。这样就减少了帐簿对外围凭证库的占用时间,缓和了“颈口”现象。②凭证的录入也需要采用批处理方式,使用中间变量暂时存储录入的凭证,统一向外围凭证库一次性附加及向对方凭证库转入,以便缓和“颈口”现象。

3、数据交换区的设置:统一将所有帐簿的所有数据库(各类字典库除外)存放在一个数据交换区里,便于统一管理,既保证了财务数据的安全性,又便于数据的转储和备份。

二、控制机制的实现

多用户帐务处理电算化系统的控制机制主要是指在时序上、空间上控制数据流在从记帐凭证录入、帐簿登录、明细科目分录、结帐、转帐、对帐直到科目汇总整个帐务处理全过程中的各种流向。而其中的关键问题是从记帐凭证录入到帐簿登录过程,其它后续工作在前期工作完全正确基础上实现起来将没有过多麻烦。

我们将记帐凭证录入和帐簿登录从时序上分为三个阶段:录入、复核、登帐。将操作方式分为两大类:修改、对帐。下面分别加以阐述。

1、录入控制机制:主要是人为规定。记帐凭证分为三大类:收款凭证、付款凭证、转帐凭证。因此规定凡涉及收、付凭证的业务均由生成日记帐的科目(例现金、银行存款)来录入;转帐凭证则由业务的贷方来录入。另外生成日记帐的科目之间的业务也以贷

方为主。

2、复核、登帐的控制机制:复核与登帐在时序上进行时分两种方式。①顺序型:适于业务不多,而且工作量在时间上分布比较均匀的部门:即录入方录入记帐凭证后并不马上登帐,而需等复核方将录入凭证复核确认后才可双方进行登帐,否则应由录入方立即对录入记帐凭证进行修改,直到复核方确认后才可继续进行。在这种方式中,复核方无权修改录入方录入的记帐凭证。这种方法的优点是:登帐的凭证已经双方确认,错误的概率大大减少,可节省大量对帐时间。缺点是:录入方必须等待复核方复核后才能进行下一步工作,造成时间滞后。②非顺序型:适于业务量大,工作量比较集中的部门:即录入方录入记帐凭证后立即登帐(不必等待对方复核);复核方复核后如无错误立即登帐,否则可立即修改录入方录入的记帐凭证,自认为正确后进行登帐。在这种方式中录入方和复核方均有权对录入的记帐凭证进行任意修改,无论是在双方登帐前或后。优点是:双方可根据自己要求修改记帐凭证,然后自行登帐,满足时间上的要求。缺点是:登帐的凭证未全部经过双方确认,错误的概率增加,双方修改过凭证均需进行对帐后方可进行科目汇总,大大地增加了对帐的工作量。

3、修改的控制机制:顺序型中修改的控制机制不太复杂,这里就不论述了。非顺序型中修改的控制机制十分复杂,在时序上、在对象上都有交替关系。另外,财政部有关文件规定:“经计算机登帐处理的系统会计凭证及据以登记的相应帐簿,软件只能提供留有痕迹的更正功能”。因此,修改的控制机制变得至关重要了。这里我们将修改的对象分为三大类:①科目方向:是指会计录入记帐凭证时将记帐凭证的借贷方向录错,这时只能简单地将该记帐凭证打上作废标志予以撤消(包括已登帐的业务明细)。②凭证内容:是指录入方录入记帐凭证时将凭证的结算票号和金额录入错误,此时可以由借贷双方任意修改。但在做科目汇总之前必须首先进行对帐。③明细调整:是指借贷双方进行帐簿登录时将自己内部的明细科目分录错误,这种情况由本帐簿内部自由调整,对科目平衡不产生影响。

除了修改对象分类外,修改的时序上也有不同分类。对于录入方分为:

①对方复核前:这时对记帐凭证的修改,既要使本帐簿内已登录业务明细可进行留有痕迹的修改,又要使对方既可以继续复核登帐,又要了解录入方已做过修改以便后续对帐。②对方复核后:这时对记帐凭证的修改,既要使本帐簿内已登录业务明细可进行留有痕迹的修改,又要使对方了解录入方已做

过修改以便后续对帐。

对于复核方的修改只有上面第二种情况。

综上所述修改对象与时序的不同的交错操作构成了复杂的修改控制机制。这种控制机制的构造又不能过于繁琐,以免造成理解困难和数据冗余。我们只在各总帐科目帐簿的外围凭证库中使用一个数字型标志数据域;在总帐科目帐簿库中使用一个字符型标志数据域,即可完成整个修改控制机制的标识工作。外围凭证库的标志数据域仅使用0~5这六个数字来标识。

1:录入方已录入并登帐,但复核方尚未复核登帐;

2:录入方已录入并登帐,并且复核方已复核确认并登帐;

0:录入方已录入并登帐,但复核方尚未复核登帐,而录入方又进行了修改;

3:①录入方已录入并登帐,而复核方已复核发现录入记帐凭证有误进行了修改,然后登帐;②录入方已录入并登帐,并且复核方已复核确认并登帐,后借贷双方中任意方发现有错进行修改,修改方修改后重新登帐;

4:无论何时借贷双方发现录入记帐凭证科目方向分录有误,进行科目方向修改;

5:所有标志数据域为“4”的凭证的作废标志;总帐科目帐簿库的标志数据域仅使用“*”或“ ”来标识。“ ”:该条已登帐的明细业务为正确;“*”:该条已登帐的明细业务为错帐;该简单的修改控制标志的设立,完全满足了整个系统修改和对帐功能的需要。

4、对帐的控制机制:对帐的控制机制就是当全部本期发生的记帐凭证录入、复核、登帐完毕后,利用外围凭证库的修改标志数据域的信息进行对帐操作。假设标志数据域中的值的集合为X;任取x属于X,存在y属于X。

① $y < 2$:仍有未复核凭证,不能进行对帐;

② $y = 4$:将该条凭证标志数据域的值改为“5”,将总帐科目帐簿库中所有属于该条凭证的明细业务的标志数据域的值改为“*”;

③ $y = 3$:将该条凭证的金额与对方科目外围凭证库中对应凭证的金额比较:如相等将该条凭证及对方凭证的标志数据域的值改为“2”;如不等由对应双方商议修改后进行对帐操作;

④ $x = 2$ 或 $x = 5$:全部对帐完毕,可进行下一步科目汇总;

以上两部分设计思想在《抚顺铝厂会计核算电算化系统》中已经实现,在试运行过程完全满足实际要求,受到用户好评。

FOXBASE+2.10 报表通用打印程序

石油大学经济管理系 贾士超(北京昌平 102249)

摘要 作者采用 FOXBASE+2.10 命令和功能,编写出一种适合各种报表格式的通用打印程序,并进行了准编译。

关键词 报表打印 数据库 字段 文件

FOXBASE+2.10 是继 DBASE III 之后出现的功能齐全、性能优良的微机数据库管理系统软件,它对微机内存实行动态管理,具有较强的计算能力和较快的处理速度,增加了内存变量数组和自定义函数,扩展了上拉和下拉式菜单、屏幕窗口滚动等功能,并提供了三维图形等辅助工具。这些重大改进吸引着众多的用户,随着产生了大量的数据库管理系统和应用程序。

数据库管理系统的输出多以打印报表的形式出现,然而 FOXBASE+2.10 软件本身所提供的报表输出格式不符合我国的报表格式习惯,而用户要输出的各种报表的表头不同,字段名不同,数量不一,字段类型有别,宽度各异,记录有多有少,大小不等,因此,在输出报表的数量较多的情况下,编写每个输出报表的打印程序几乎是不可能的。因此,笔者用 FOXBASE+2.10 所提供的命令和功能编写了一段程序,并进行了准编译,实现了各种报表格式的打印输出。本程序需要在 2.13H 汉字系统支持下运行,用户只需在 FOXBASE+2.10 系统中给出数据库文件名,报表即可按我国习惯的格式自动打印输出。程序清单如下:

一、主程序 fbtp.prg

* FOXBASE+2.0 报表打印输出通用程序

```
CLEAR
CLEAR ALL
SET DATE ANSI      && 设置程序运行环境参数
SET ESCAPE OFF
SET STATUS OFF
SET PROCEDURE TO sub  && 打开过程文件 scr.prg
DO scr              && 调用初始屏幕画面过程 scr.prg
answ=" "
DO WHILE NOT answ$ "Ee"      && 总循环,循环一次打印一张报表
STORE " " TO myn
DO WHILE NOT .myn$ "yYtT"    && 确认文件名及路径正确输入
STORE SPACE(8) TO mdbf
STORE " " TO myn
STORE " " TO mdrv
STORE SPACE(15) TO mph
@2,2 SAY "请输入:"
@4,2 SAY "1. 默认驱动器是(A:,B:,C:,D:),"GET mdrv
@6,2 SAY "2. 指定路径名是,"GET mph
@8,2 SAY "3. 要打印的数据库文件名(不带扩展名),"GET mdbf
```

```
READ
SET DEFAULT TO &mdrv
SET - PATH TO &mph
sdbf=TRIM(mdbf)+".dbf"
IF FILE(sdbf) . && 判断文件是否存在
SELECT 1
USE &sdbf  && 若文件存在,则在工作区打开
LIST OFF
? CHR(7)
WAIT "是要打开的数据库吗?" TO myn
CLEAR
ELSE
@12,2 SAY " "
? CHR(7)
?"指定磁盘上不存在该文件,重新输入!"
ENDIF
ENDDO
COPY TO strf STRU EXTENDED  && 产生数据库结构文件,以获取结构信息
myn=" "
mttl=SPACE(30)  && 表头标题变量
mtco=" "
munt=SPACE(8)
unyn=" "
DO WHILE NOT .myn$ "Yy"
@4,2 SAY "请输入(不选择,按回车!),"
@6,2 SAY "1. 报表表头名称,"GET mttl
@7,2 SAY "2. 报表顺序数,"GET mtco
@8,2 SAY "3. 报表数据统一单位,"GET munt
@9,2 SAY "4. 是否打印制表日期?"GET unyn PICTURE "@Y"
@11,2 SAY "输入正确吗?"GET myn PICTURE "@Y"
READ
ENDDO
sttl=L TRIM(TRIM(mttl))
stco=L TRIM(TRIM(mtco))
sunt=L TRIM(TRIM(munt))
lett=LEN(sttl)
leco=LEN(stco)
leun=LEN(sunt)
PUBLIC R,C  && 用公用变量 R,C 代替打印机坐标测试函数
R="PROW()"
C="PCOL()"
@20,2 SAY " "  && 定位屏幕光标位置
? CHR(7)
WAIT "准备好打印机后,按任意键开始打印..."
CLEAR
@12,20 SAY "正在打印...按 ESC 键暂停!"
@24,0 SAY " "  && 定位屏幕光标位置
SELECT 2
USE strf  && 选择第二工作区,打开数据库结构文件
```

```

PUBLIC rec,width    && 定义为公用变量
rec=RECCOUNT()      && 结构文件的记录数,即打印数据库的字段数
width=0             && 表格中所有字段名或记录的总宽度变量
DO vdim             && 调用过程 vdim.prg,获得数据库结构信息
width=2+2*rec+width && 表格总宽度,2+2*rec 为表格中竖线的总宽度
ttlm=(width*24-lett*36)/(2*24) && 求表头标题对中值公式)
ttlm=IIF(ttlm>0,ttl,0) && 确保正数
SET ESCAPE ON
ON ESCAPE DO stpp    && 此后只要按 ESC 键,则调用过程 stpp.prg
SET DEVICE TO PRINT  && 将下面@....SAY...中的输出内容送往打印
打印机
SET MARGIN TO 0
@ 0,0 SAY SPACE(ttlm)+"@j"1"+sttl+"@ A" && 打印表头,字型
为"j"
@ & R,&C SAY"&20`0">" && 设行距为20,字间距为0,以打印实线表
格
IF leco>0
leco=lec0+2
@ &R+1,2 SAY "表"+stco && 在表格左上角上打印表顺序
ELSE
@ & R+1,2 SAY " "
ENDIF
IF leun>0
@ & R,0 SAY SPACE(width-leun-8)+"单位,"+sunt+"&16"
ENDIF && 设定行距为16,以打印实线表格
SELECT 3
USE otab && 选择第三工作区,打开制表符数据库文件(注)
DO hebo WITH 1 && 调用过程 hebo.prg,打印标题栏第一行
GOTO 2
DO hebo WITH 0 && 调用过程 hebo.prg,打印标题栏第二行
GOTO 3
DO hebo WITH 1 && 调用过程 hebo.prg,打印标题栏第三行
GOTO 2
SELECT 1 && 选择打印数据库工作区
GO TOP
DO WHILE NOT .EOF() && 将数据库中的所有记录打印完为止
SCATTER TO Y && 将打印数据库中某个记录的数据赋予数组 Y
SELECT 3 && 制表符数据库工作区
DO hebo WITH 2 && 调用过程 hebo.prg,打印表体中的数据
IF & R>55 && 打印行数超过55行,则换页
DO chpg && 调用换页过程 chpg.prg
ENDIF
SELECT 1 && 选择打印数据库工作区
SKIP
ENDDO
SELECT 3
GOTO 4
DO hebo WITH 1 && 调用过程 hebo.prg,打印表格最后一行
IF unyn$="Yy"
@&R+1,(width-20) SAY"制表日期,"+DTCO(DATE())+"&24"
ENDIF
EJECT && 走纸换页
SET DEVICE TO SCREEN && 将下面@....SAY...中的内容送往屏幕
ON ESCAPE && 关闭 ESC 键的激活功能
SET ESCAPE OFF
CLEAR
CLEAR ALL && 清除所有内存变量,关闭所有打开的文件
ERASE strf.dbf && 删除程序本身所建立的结构文件
? CHR(7)
14,20 SAY "打印完毕!"
20,18 SAY ""
WAIT SPACE(18)+"按 E 键退出,其它键打印另一张报表..."TO ans

```

```

CLEAR
ENDDO
CLOSE PROCEDURE && 关闭过程文件
SET TALK ON
SET ESCAPE ON

```

注:otab.dbf 的记录为:

Record # OA OB OC OD

1				
2				
3				
4				

二、过程文件 sub.prg

* 屏幕画面过程 scr.prg

```

PROCEDURE scr
SET SCOREBOARD OFF
CLEAR
H="@0,0 SAY CHR(14)"
& H+"QROCI5D50,348BD540,200]" && 利用 2.13H
的特殊显示功能,美化屏幕
&H+"V8,50,348,540,200,1,50,1]" && EGA 640×
350 下的情况
&H+"[-100|160*4@IFOXBASE+]"
&H+"[-150|250*1@P报表打印]"
&H+C15D72,148B489,30]"
&H+V8,72,148,489,30,1,20,1]"
&H+"C15D88,118B461,25]"
&H+V8,88,118,461,25,1,15,1]"
&H+"C15D99,93B441,20]"
&H+C15D110,73B419,15]"
&H+"C15D121,58B397,11]"
&h+"C15D132,47B375,8]"
&H+"C15D143,39B353,6]"
&H+"C15D154,33B331,5]"
&H+"C15D165,28B309,4]"
&H+"C15D175,24B287,3]"
k=inkey(3) && 延时3秒
&H+"QR1]"
RETURN

```

* 随时中断打印过程 stpp.prg

```

PROCEDURE stpp
SET DEVICE TO SCREEN
CLEAR
L=INKEY()
? CHR(7)
@ 10,0 SAY ""
WAIT"按 Q 键退出打印,其它键继续打印..."TO choice
IF UPPER(choice)="Q"
? CHR(7)
CANCEL && 终止程序,返回圆点提示符
ELSE
CLEAR
@14,2 SAY"继续打印...,按 ESC 键暂停!"
SET DEVICE TO PRINT
RETURN
ENDIF

```

* 换页过程 chpg.prg

```

PROCEDURE chpg

```

```
GOTO 4
DO hebo WITH 1  && 打印表格底线
EJECT
SET DEVICE TO SCREEN
@ 14,20 SAY " "
? CHR(7)
WITH SPACE(20)+"准备好打印机,按任意键开始打印新的一页..."
SET DEVICE TO PRINT
@0,0 SAY " "  && 初始化打印机坐标测试函数
GOTO 1
DO hebo WITH 1  && 打印续表第一行
GOTO 2
RETURN  && 返回调用程序,继续打印
```

* 获取数据库结构信息过程 vdim. prg

```
PROCEDURE vdim
PUBLIC wl(rec),wr(rec),x(rec),y(rec),z(rec)  && 定义为用变量数组
j=1
GO TOP
DO WHILE .NOT. EOF()
lng2=FIELD-LEN
fnn2=FIELD-NAME
y(j)=TRIME(fnn2)  && 字段名
z(j)=LEN(y(i))  && 字段名宽度
mx2=MAX(lng2,z(j))  && 字段名宽度与字段宽度最大者
x(j)=ROUND(mx2/2,0)  && 中文方式下表格中各栏宽度
wdth=wdth+2*(j)  && 表格总宽度
wl(j)=INT((x(j)-z(j)/2))  && 某栏中数据对中左边空格数
wr(j)=ROUND((x(j)-Z(i)/2),0)  && 数据对中右边空格数
j=j+1
SKIP
ENDDO
RETURN
```

* 打印过程 hebo. prg

```
PROCEDURE hebo
PARAMETER u  && 参数为 u,即标题栏、表体和表底的不同打印情形
i=1
DO WHILE i<(rec+1)  && 直到一个记录的字段数据打印完毕
IF i=1
@ & R+1,0 SAY OA  && OA、OB、OC、OD 是制表符数据库中的字段变量
ENDIF  && OA 即“r”
DO CASE
CASE u=0
@ & R,& C+wl(i) SAY y(i)+SPACE(wr(i))  && 打印标题栏中的字段名
```

```
CASE u=1
@ & R,& C SAY PEPLICATE(oB,x(i))  && 打印表格中横线
CASE u=2  && 打印数据库中数据
SELECT 2  && 选用打印数据库的结构数据库 strf. dbf
GOTO i  && 将指针定位于第 i 个记录
fityp = FIELDS-TYPE  && 赋字段类型
fdec = FIELD-DEC  && 赋字段小数位数
IF fityp="N"  && 若定段为数值型
IF y(j)=0  && 若数值为 0 或 0.00
@ & R,& C,SAY SPACE(2*(i))  && 打印空格
ELSE  && 若非 0 值
y(i)=L TRIM(SRR(y(i),19,fdec))  && 数值型字段不超过 19 位
z(i)=LEN(y(i))  && 转换为字符型并测出其长度
@ & R,& C+x(i)-INT(2(i)/2) SAY Y(i)  && 数据左边空格,以使其对中
@ & R,& C SPACE(x(i)-ROUND(z(i)/2,0))  && 数据右边空格,以使其对中
ENDIF
ENDIF
IF fityp="C"  && 若为字符型
y(i)=TRIM(LTRIM(y(i)))
z(i)=LEN(y(i))
@ & R,& C+x(i)-INT(Z(i)/2) SAY y(i)
@ & R,& C SAY SPACE(x(i)-ROUND(z(i)/2,0))
ENDIF
IF fityp="L"  && 若为逻辑型
@ & R,(& C+x(i)-1) SAY y(i)  && 逻辑型字段宽度为 3
@ & R,& C SAY SPACE(x(i)-2)
ENDIF
IF fityp="D"  && 若为日期型
@ & R,& C SAY y(i)
@ & R,& C SAY SPACE(2*(i)-8)  && 日期型字段宽度为 8
ENDIF
IF fityp="M"
@ & R,& C SAY SPACE(2*(i))  && 若为备注型字段,不打印
ENDIF
SELECT 3  && 选用制表符数据库工作区
ENDCASE
IF i<rec
@ & R,& C SAY OC  && 打印表格中中间的竖划线
ELSE
@ & R,& C SAY OD  && 最后一个字段打印完后,打印最右边的制表符
DNDIF
i=i+1
ENDDO
RETURN
```

本程序短小精练,可作为 FOXBASE-2.10 中任何实用程序的一个过程来调用。勿庸置疑,该程序可打印输出任何在 DBASE 下建立的数据库报表。

IT286 DH 微机键盘故障维修一例

广西南宁铁路电务段微机工区 林勇奇(530001)

故障现象

一台 IT286 DH 微机,开机约十五分钟后键盘死锁,击任何键屏幕上均无反应,复位后可工作约十五分钟后又进入死锁状态。用户反映该机在购回之后不久即有此现象,开始时可连续工作约三小时左右,后来可连续工作时间日渐缩短。

检查与处理

此故障呈现周期性,怀疑是元件的性能不稳定。开机检查,待故障出现时,用逻辑笔检测键盘输出插座中的数据输出引脚(2、3脚),发现其输出固定为低电平,击任何键均无变化,可判断故障在键盘电路。用逻辑笔检测键盘电路,测得单片机 8035 的 PSEN、ALE 端有信号输出,DB0~DB7 信号有变化,说明单片机工作正常。检查 P1 口,发现该口输出全固定为低电平,检查电路得知单片机 8035 的 P1.3、P1.4、P1.5、P1.6 分别与 M5(74LS154)的 A、B、C、D 端相连接,查 M5 的这几个引脚亦为低电平,说明它们之间的接线无问题,查 M5 的输出选通 y0 列,说明 M5 工作正常。单片机 8035 的 P1.0、P1.1、P1.2 分别与 M3(4512)的 A、B、C 端相连接,查 A、B、C 三个引脚为低电平说明接线良好,此时 M3 的输出应选通 x0,键盘扫描的位置为(y0,x0),该位置是【↓】键,按动【↓】键检查 M3 的 O/P 输出

端有变化,M6(74LS14)的 2 脚的电平亦随着变化,信号能到达单片机 8035 的 T1(39 脚),说明译码器 M5 及数据选择器 M3 的工作正常,故障范围缩小到 M1(2764)及 M2(74LS373)两片芯片上。查 M2 的输出 1Q~8Q 信号有变化,芯片工作正常,至此 M1 是最后的怀疑对象。用无水酒精擦干净电路板,然后用放大镜仔细观察 M1 与单片机的连线,未发现断线和焊接不良,故怀疑是 M1 本身的故障,有文献介绍一些 EPROM 快速编程器,由于编程速度太快,注入电荷太少致使 EPROM 固化的数据不可靠,怀疑该故障与此有关。将 M1 焊下,用 BJM-1 型编程卡读出 EPROM 内的数据,然后再将读到内存的数据与芯片上的数据进行比较,发现在连续 20 次的读出比较中有 3 次发生异常。考虑到键盘在开机或复位后能正常工作一段时间,因此可以认为比较中数据相同次数多的数据是正常的,而 3 次比较不同的数据带有奇异数据。取正常的数据给 EPROM2764 重新固化一遍,将其焊回板上,故障排除。

原因分析

当 EPROM 内的某些单元的数据发生变化后,单片机将从中取到错误的“指令”,进行错误的操作,结果是执行了一段莫名其妙的程序,最后落入一段死循环“程序”,从而使键盘发生“死锁”。

珠算水平竞赛统标命题不再难

(本刊讯) 贵州省毕节地区财贸学校电脑室(邮编:551700)的莫卫平经长期潜心研究,现开发出一种在计算机上实现珠算水平竞赛试卷的自动生题软件。所生成的试卷严格遵照中国珠算协会标准,一次输出成形,包括实现手写体、字码大小、间距、行距自动调整等功能,完全符合国家标准,出题速度快,试题套数多(可达 6 万套),从而缓解了珠算竞赛统标命题出题难和新卷供不应求的矛盾。该软件已通过贵州省珠协技术鉴定。需此软件者,请同莫卫平联系。

好消息

(本刊讯) 本刊新接到国家科委中国科技信息研究所通知,《计算机应用研究》继续被列为中国科技论文统计源用期刊。1991 年中国科技论文统计源用刊为 1230 种,1992 年为 1217 种,舍去了少数刊登科技论文太少的期刊,增加了质量较高、已试办多年而最近改为正式出版的学术类或科技类期刊,或办刊时间不长但反映高科技研究与发展状况的期刊及国际合办在中国出版的中文期刊。

小结硬盘不能自举的故障分析、排除及保护措施

山西省临汾行署计委经济信息中心 汪文娟(041000)

众所周知,硬盘以其存贮容量大,运行速度快,操作简便等优点而深受广大用户的欢迎。而一旦硬盘出现了故障,将会给用户造成很大的损失。在此,特向大家介绍几种有关硬盘不能自举的故障分析、排除及保护措施。

一、故障现象及分析

硬盘自举时,屏幕通常显示如下几种信息,归结起来,均属于不能自举的故障现象。

1. Disk boot failure
2. Error loading operating system
3. Missing operating system
4. Invalid partition table
5. Non-system disk or disk error
6. 系统转入 ROM BASIC 而锁机
7. 系统不能识别硬盘符,屏幕显示:

Invalid drive specification

上述故障现象有以下几种原因:

①磁盘面擦伤。尽管硬盘的盘片和磁头都是密闭的,空气通过高效过滤器才能进入密闭室内,但进入密闭室的空气也不是绝对无尘埃的。还有磁盘在未接通电源时,磁头和盘片是直接接触的,当只有在工作时才会产生浮力,但磁头和盘片间有一个很小的间隙,因而不得当的搬运和装卸,强烈振动都可能会使磁盘的磁面损伤,出现定位不正确和读/写错误。

②使用环境差。机器的使用环境差,卫生条件不好;有的计算机室内还有静电复印机;有腐蚀性气体;湿温度偏高偏大等,都可能会使尘埃(特别是复印机的碳粉)进入密闭室,粘在磁头或盘片上,造成读写错误。

③掉电和人为误操作。当系统出现上述故障现象时,多为硬盘上的0道信息被破坏。磁道0是由两个引导模块分配表和目录区及三个系统文件所组成。磁道0介质损坏,系统便不能通过该驱动器自举。

二、故障排除方法

故障的排除方法很多,应根据故障现象认真分析,对症下药。通常可试行下列方法排除。

1. 首先用正确的 COMMAND.COM、IBMBIO.COM、IBMDOS.COM 文件替换错误的或版本不匹

配的文件,观察能否恢复正常。

2. 若不能恢复正常,则运行 FORMAT C:/S 程序观察能否恢复正常。

3. 若运行失败(也就是不能恢复正常),可以通过运行低级格式化 LOWFORM.EXE 程序,重新对硬盘进行初始化,这样可以把硬盘中的坏扇区识别出来,记录在 DOS 的可读写的记录块中,在系统读写硬盘时会自动跳过这些坏扇区。然后调用 FDISK 程序,在调用 DOS 分区时,打“N”并在光标提示符“—”下,设置开始标号为“1”,结束标号为“304”,总柱数为“304”,最后用 FORMAT C:/S 进行格式化,观察看能否恢复正常。

4. 若上述三种方法均不成功,可通过调用 FDISK 程序激活 DOS 分区,使含有 DOS 的分区状态为“A”,即在提示符下打“2”(改变激活分区),再打 DOS 分区的序号(1,2 或 3)(故障现象 6 可实行这种方法排除)。

5. 若上述四种方法都失败,那么只能从硬件结构上调整,避开 0 磁道。

三、保护措施

①关机前运行 SHIPDISK 或 PARK 锁头程序,使硬盘磁头离开数据区,再切断主机和驱动器电源。

②硬盘运行格式化 FORMAT C:/S 后,删除 COMMAND.COM,然后用一个或多个没用的大程序拷贝到硬盘上,使它足以占满 0 磁道,并将该文件置为可读,而不能删,以防止删除后让出空间被后来文件存入时占用,最后向磁盘拷贝各种程序和 COMMAND.COM。

(注:上述方法适用于 IBM PC/XT 低版本操作系统机型效果最佳)。

欢迎订阅《西南旅游》双月刊

邮发代号:62—87

国外代号:BM656

定 价:每期 1.60 元

全年 6 期:9.60 元

通讯地址:成都盐道街 3 号

邮政编码:610012

带字库打印机的打印针的故障检测

湖南财经学院信息系 马良渝 (长沙 410079)

打印针是用特殊材质制成的高耐磨零件,但是在报表打印中,其运动次数常高达每日数百万次。尽管其额定寿命为二亿次,然而一个中型企业的财务系统的繁多的划线报表却往往在 6~8 个月中即可将中间的几根针磨损。在国内流行的高速的带字库的打印机中,这种情况更加严重。

用肉眼一般难以分清究竟是哪一根针磨损了,为此有必要用软件技术协助诊断。

这类打印机均装有 24 根针,按从上到下的次序分成三组,每组八根针,对应一个字节的 8 个二进制位。为了便于用高级语言处理,可将每根针的二进制码转换成十进制。设上、中、下三组分别为 P、Q、R。其示意图如图一所示。

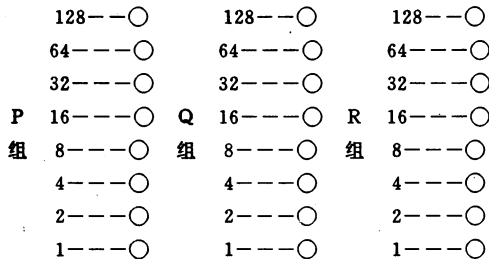


图 1

运用大家熟悉的 BASIC 语言可写出打印针的故障检测程序如下:

```

10 'Failure diagnosis of Printer's pins
30 CLS
35 PRINT "    TEST PIN OF PRINTER MENU"
40 PRINT "    1. AR2463"
45 PRINT "    2. AR3240"
50 PRINT "    3. LQ1600K"
55 PRINT "    4. OKI8320C"
60 PRINT "    5. QUIT TEST MENU"
70 INPUT "    PLEASE SELECT MENU NUMBER
   ",N
100 I=1
110 FOR M=1 TO 3
120 IF N=1 THEN A=1 ELSE A=128
130 FOR K=1 TO 8
135 ON N GOSUB 200,300,300,400,195
140 IF M=1 THEN P=A,Q=0,R=0
145 IF M=2 THEN P=0,Q=A,R=0
150 IF M=3 THEN P=0,Q=0,R=A
155 FOR J=1 TO 60

```

```

160 LPRINT CHR$(P),CHR$(Q),CHR$(R);
165 NEXT J
170 LPRINT
175 I=I+1
180 IF N=1 THEN A=A*2 ELSE A=A/2
185 NEXT K
190 NEXT M
191 LPRINT
192 LPRINT"图二"
195 END
200 LPRINT I,CHR$(27);"J0060";
220 RETURN
300 LPRINT I,CHR$(27);"*";CHR$(38);CHR
   $(60);CHR$(0);
310 RETURN
400 LPRINT I,CHR$(27);"%1";CHR$(60);CHR
   $(0);
410 RETURN

```

菜单中列出了国内常用的带字库的打印机,运行中只要选择对应的号码即可打印出图二的结果。如果所有的针完好,在每一个针号之后应该有一条清晰的黑线。如果某号码后面的线条断续不清,则表明该针磨损严重,或针通道有堵塞现象,应更换针或清洗打印头。如果某号码后面无线条出现,则是磨损严重或已断针。增加菜单选项和不同的打印机的点形象命令,可以将该程序用于任一种打印机。在不带字库的打印机上运行时,注意不要启动汉字打印驱动程序。因为有的驱动程序会屏蔽掉程序中的某些控制参数(如 CHR\$(0))而使诊断失败。

图中 2 号后无线条,表明 2 号针故障严重经检查确实该针已断。

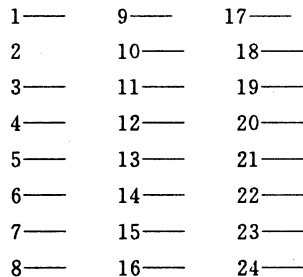


图 2

本文作者任信息系副主任、副教授,从事系统开发及硬件维护、教学及科研。

ROM BIOS 系统参数在 IBM PC 机维修中的应用二例

华中理工大学自动控制工程系自控教研室 甘早斌(武汉 430074)

所谓 ROM BIOS 是直接和硬件设备打交道的管理程序,它驻留在系统板上的只读存储器 ROM 中,主要完成以下几个功能:(1)对硬件的自诊断测试;(2)进行系统配置分析;(3)对基本 I/O 设备初始化;(4)引导 DOS 系统或 ROM BASIC;(5)提供基本的中断服务,等等。一旦 PC 机启动成功后,ROM BIOS 将加电自诊断测试的结果、硬件设备的配置情况(如:软盘、硬盘、显示器、打印机等的各种状态信息)和操作结果存放在内存绝对地址为 400H—4FFH 的范围之内,通常称这一区域为 ROM BIOS 系统参数区(也有的则叫 BIOS 数据区(BIOS DATA AREA)),用户可以从该区读取各种有用的信息来进行计算机病毒的检测、计算机的故障分析等等。

下面结合两个例子把作者的一点点经验介绍给读者,以供参考。ROM BIOS 数据区的具体内容请参见毛明著的《IBM PC 及长城(0520)系列微机实用软件技术》一书中的附录 B 或者 STEVEN ARM-BRUST 和 TED FORGERON 著的《PC/AT USER'S REFERENCE MANUAL FOR IBM PERSONAL COMPUTERS》一书(该书有中译本)。

一、病毒的检测

有一台 COMPAQ 386 微型机,用硬盘启动后,不论是按 CTRL _ P 还是 COPY FILENAME TO PRN 打印机都无任何动作,计算机并发出“嘟! 嘟! ……”的响声,起初以为是打印机出了什么毛病,结果换了一个性能完好的打印机后问题仍然没有解决。后来,调入 DEBUG,显示 ROM BIOS 参数区的内容如下:

```
A)DEBUG
-D40,0000
0040:0000 00 03 00 02 00 00 00 00-00 03 00 00 00 00 00 00
0040:0010 61 44 BF 80 02 00 00 00-00 00 30 00 30 00 10 19
0040:0020 64 20 34 05 30 0B 3A 27-30 0B 30 0B 30 0B 30 0B
0040:0030.....
```

由 BIOS 数据区的参数可知,400H—407H 存放的是 RS-232 适配器的基地址 COM1:3F8H, COM2:2F8H,408H—409H 存放的是 PRN 口地址 378H,而上面显示的则是 0300H、0300H、0300H,这样就可以断定,该硬盘的引导扇区感染了病毒,该病毒修改了 DOS 通信区的 COM1 串口,COM2 串口

和 PRN 口的地址,因此,凡是与 COM1、COM2、PRN 等有关的操作均不正常。后来找了一台同 DOS 版本同机型的无病毒的微机,首先利用中断 13 将干净硬盘 0 面 0 磁道 1 扇区的硬盘主引导记录读到内存的缓冲区,然后将缓冲区中的内容写到 A 盘的 0 面 15 磁道 1 扇区,其次再用干净的系统盘启动带病毒的机器,利用中断 13 将 A 盘的 0 面 15 磁道 1 扇区的硬盘主引导记录读入内存缓冲区中,然后将其写入被感染病毒的硬盘 0 面 0 磁道 1 扇区中[程序略],重新启动计算机,打印机恢复正常了。

二、故障分析

另有一台 PC/AT 微型机,用 A 盘启动时,指示灯亮,但不能读盘;用硬盘启动后,对 A 盘也不能进行读写操作,这显然是软盘驱动器部分的毛病,但毛病究竟出在哪里呢?起初把主机外壳取下来,检查了几遍,始终找不出原因所在,后来又调用 DEBUG,显示 ROM BIOS 参数区的内容如下:

```
A)DEBUG
-D40,0000
0040:0000 F8 03 00 00 00 00 00 00-78 03 00 00 00 00 00
0040:0010 41 52 BF 80 02 00 00 00-00 00 3C 00 3C 00 30 0B
0040:0020 3A 27 30 0B 30 0B 0D 1C-10 19 64 20 34 05 30 0B
0040:0030 3A 27 30 0B 30 0B 30 0B-30 0B 0D 1C 34 05 00 00
0040:0040 4E 44 C0 00 00 00 00 00-00 03 50 00 00 10 00 00
0040:0050 00 12.....
```

由 BIOS 系统参数可知 0441 处为软盘操作后的出错标志,44H 表示寻道失败和所需扇区未找到的错误。由此可知,问题就出在与磁头读写动作有关的部分,经检查,无索引脉冲,最后证实是发光二极管和光敏三极管错位了,不能产生索引脉冲,无法找到标志磁道的起始位置,因此,不能进行读写操作。经调整后,A 盘读写,启动均正常了。

从上面的二个例子来看,较好地利用 ROM BIOS 系统参数,可使我们的工作事半功倍,收到很好的效果,上面是作为例子来说明,只用到了绝对地址为 0400—0409H、0441H 中的数据,事实上,我们从 0449—0466H 单元中可以了解到显示器当前状态、显示器控制芯片起始口地址等的各种信息,从 0413H 单元中可以知道系统的实际内存容量大小;从 0474H 单元可以得知硬盘的操作状态信息等等,这些均可为我们计算机故障分析与维护中带来很大的方便。

计算机软件的发展趋势和潜在的突破

武汉化工学院计算中心 王庆春(430074)

摘要 软件是计算机技术中的一个重要领域。本文主要论述了当前计算机软件发展的几个主要趋势,即软件市场的商品化、软件的标准化、数据库、网络系统及人工智能等。同时,又介绍了软件的几种潜在的突破,即并行程序设计、形式化技术及非过程语言程序设计等。

一、前言

软件是计算机技术中的一个重要领域,同时它深刻的影响着计算机硬件的发展。因此,软件在计算机的应用中起着十分重要的作用。但是随着个人计算机的大量出现及软件市场的激增,软件工业发生根本性的变更。即由过去的大型计算机厂家及计算机公司来开发软件,逐步变为小型的专业性公司或个人来开发软件。同时,这种变革又推动软件的发展,从而产生大量较好的软件和软件的开发工具及环境,并建立了各种基于网络和通讯技术的综合性应用系统。另外,超级计算机软件、智能化软件以及更高级的软件开发方法和工具等相继问世,为软件工业的发展以及应用展现出了更加美好的前景。

二、软件的主要发展趋势

当前计算机软件的最主要发展就是软件的商品化。随着个人计算机的普及和应用范围的不断扩大,随之而来的软件就成为商品,走上了流通市场。但过去软件常常是以少量的并同硬件组装在一起的方式出售,目的是想吸引客户购买支持这些软件运行的硬件系统。而今天,这种情况正在发生变化。其一是个人计算机的广泛应用已使得软件进行了大量的传播,目前一个流行的软件可能有上百万个复制品,这与计算机用户的显著增加有直接关系,同时又有很多特定的应用(如电子数据表等),又促进了计算机更为广泛的应用。这种连锁发展趋势的一种结果,促进了数据库的发展,这样就使软件逐步变成商品。另一方面是对于用户界面的不断改善,使其出现了一些功能日益增强的图象处理和文字处理等,为了适应计算机的更广泛应用,就促使这些软件成为商品。其二,是有成千上万个小的公司或个人正在开发和出售软件,并从中赢利。其中他们中的大多数并不从事任何软件的开发工作,仅依赖于出售软件的收入,这无疑就加速了软件成为商品。

随着软件市场的商品化,就要求软件朝着统一化和标准化方面发展。因此在许多层次上就出现了标准接口。过去是不具备这种一致性的,而今天就有

很多统一的标准。例如,文件格式、图形程序库(Macintosh quick Draw 和 X 窗口系统)、操作系统(如 PC-DOS、UNIX)和 OS/2 等)、页面格式化语言和网络协议等。另外,高级程序设计语言 FORTRAN 的使用已实现了一个有效的标准化,因此,可以相当容易地将这类软件移植到各种机器上。这种标准化的发展对软件的进一步发展产生了深刻的影响。

软件朝着商品化和标准化方向发展的综合影响将是发展软件部件。过去软件典型地分布在大型的,整体的一个完整的系统里,而今天通常是以小的部件为一体。例如,电子数据表程序、字处理软件、数据库管理程序、编译程序、网络通讯程序以及许多别的软件都是以部件的形式出售。另外一些子部件,如输入/输出驱动程序、屏幕管理程序、文件存取方法及格式转换程序(如 LOTUS 1-2-3 格式换成 DIF 格式)等,也是单独发展的。大力发展软件部件将是计算机软件发展的一个主要趋势。

近 10 年来数据库技术已逐渐发展成熟,它已成为计算机和许多其它科学技术领域的一个重要组成部分。早期的数据库对用户的要求要具有较高水平的专门技术,特别是程序员,他们必须懂得数据管理的概念,了解有关的应用环境,并能对应用问题进行程序设计。而近年来许多成功的数据库管理系统(DBMS)相继问世,既有适合于大、中、小、微型计算机中某些特定的专用系统,也有适合于所有类型的通用系统。例如,SQL/DS、DB2、INGRES、ORACLE 等,它们的共同特点是使用简单,出错能够恢复,并具有用户友好的接口和一系列优良的性能,就使得数据库技术的应用更为普遍。目前存在着一种数据库技术与人工智能相结合的发展趋势。作为二者相结合的产物,已产生了一批专家数据库、知识数据库和演绎数据库等。这类系统既能处理大量的数据,又具有推理、递归的智能化能力。随着数据库应用领域的不断扩展,又出现了多介质信息管理问题,这种信息系统要处理的信息除了传统的以格式化数据表示

的信息之外,还包括多种存储在不同介质上,以图象、图形、正文、声音等非格式化数据表示的信息。因此,设计多介质信息管理的 DBMS 已成为数据库领域中一个新的重要研究方向。除此之外,面向对象的数据库模型和实现技术,新型数据库应用系统的设计方法和工具等也是引人注目的前沿课题。

个人计算机和工作站的出现,促进了计算机网络的发展。若没有计算机网络,每个用户只能孤立地使用他自己所占有的机器。与此相反,在计算机网络环境中,本机用户可以共享其它的系统资源,或把本机系统的功能和资源供给网中的其它用户使用,所有这一切都是在网络软件的支持下实现的。网络软件包括通讯软件、网络协议软件、网络应用软件、网络服务管理系统以及用于特殊网络站点的软件等。

除了向组合式通用软件的趋势发展之外,开发特殊的软件依然是一种很强烈的要求。例如,工厂的自动化系统、航空交通系统等。这些系统或多或少的是以实时的方式出现,且由计算机系统所控制的过程,所发生的速率远比人工干预要快得多。近年来,计算机行业和应用部门正在积极研究计算机集成制造(CIM)系统。此系统是在 80 年代后期,随着工业机器人、柔性制造技术的兴起,为适应机械、电子行业小批量、多品种、短周期的特点。将计算机辅助设计(CAD)、计算机辅助工程(CAE)、计算机辅助制造(CAM)、计算机辅助检验(CAI)、管理信息系统(MIS)以及计算机辅助决策支持系统(DSS)等集成在一起的计算机综合制造系统。

人工智能是一门由心理学、哲学、语言学、电子学和计算机等多学科交叉而成的一门科学。人工智能的任务是研究使机器能够执行人类智慧所需要的任务。欲使机器具有智能能力,将要进行适当的程序设计,并为其提供相应的数据库。人工智能的进展取决于若干要素,如适应于编写人工智能的程序设计语言,用的最多的是 LISP 语言、prolog 逻辑程序设计语言以及面向对象的程序设计语言 Smalltalk 等。优良的计算机和先进的计算技术对于人工智能的研究十分重要。基于知识系统(即专家系统)是人工智能最活跃的应用研究领域。专家系统通常包括知识库、推理求解系统以及人机接口等。对于许多应用,专家系统必须与数据库和其它程序连接起来,但在开发中获得知识的任务可能比编制专家系统程序的任务更为困难和复杂,它涉及到要与具有专家知识的人进行交互作用。人工智能还与某些具有学习能力、处理感觉信息(如听觉和视觉)能力、控制其它类型的机械(如机器人等)以及能够执行其它的认识功能的计算机系统的发展有一定的联系。总之,人工

智能的发展是一门多学科综合发展的产物。

三、潜在的突破

计算机软件进一步发展,可能在以下几个领域,即并行程序设计、形式化技术、非过程语言程序设计等方面获得突破。

高性能计算机的应用是增强并行性,即在同一时间内可以使用几个功能单元或进行并行操作,以提高某些问题的求解速度。目前尽管人们仍在设法提高单处理机的速度,但是如果能够并行地去解决一个问题。其性能和速度将会得到更为显著的改善。获得具有高度并行性能的关键在于能够编制并行程序。大多数的现行软件是为单处理机的任务而编制的,若将一个程序进行适当的分割,使得不同的模块可以在不同的处理机上并行的执行,无疑这将是一个大的突破。为使并行处理变为现实,则必须发展各种程序设计的新技术,例如并行程序设计语言与并行编译技术。同时这些新技术势必要与新的开发环境紧密相连,以便能够快速而便利地生成高效率的并行程序。并行程序的突破将会在总体上对计算机科学和技术产生深远的影响,将会导致具有更高的处理能力的机器被普遍使用,并行处理机的处理能力将是非并行处理机处理能力的 10~100 倍。

另一个潜在的突破是将形式化方法用于软件开发,形式化方法或技术将涉及到给程序附加精确的含义。最严格的方法是程序验证法,即证明一段代码与实现这段代码规范说明之间功能的等价性,或证明规范说明与一组需求之间功能的等价性。形式化技术能够运用到各种研制开发项目中,例如关系运算、面向对象设计、良好的说明标准和接口以及 Ada 和 PASCAL 等强类型的语言。由于形式化技术能减轻软件开发中的测试阶段和维护阶段的工作量,所以它具有提高生产率的巨大潜力。若在软件开发中使形式化技术有所突破,将会大大提高软件的自动化程度和可靠性,降低软件测试和维护的成本。

近年来软件的发展是不断的提供新型的用户接口和程序设计工具。例如电子数据表程序等,它表达了管理和显示计算机中信息的一种崭新的方法。电子数据表还提供了一种新型的程序设计语言,用于描述电子表格中各个登记项之间的联系,使得在一个登记项发生改变时,能够自动修改其它有关的登记项。这种程序设计形式与传统的程序设计形式有很大区别,它是非过程语言程序设计的形式。随着计算机的发展和广泛的应用,非过程语言程序设计形式将会有很大突破。

参考文献

- [1]张海藩编著,《软件工程导论》,清华大学出

联机不打印故障分析

沈阳冶金机械专科学校 刘景明 (110044)

一、故障现象

现有长城 286 主机, 3070 打印机和稳压电源设备一套。打印机自检正常, 屏幕打印正常, 主机工作正常, 但是用 ctrl+p 命令打不出文件和目录的内容。

二、故障分析

1、在脱机状态下, 检查打印机是否自检, 如能自检说明打印机无故障, 否则检查维修打印机。

2、检查打印电缆, 最好换一条经试验无故障的电缆, 故障仍存在, 说明电缆线无问题。

3、在软硬盘上分别启动 DOS, 用 DIR 命令列目录, 能列出目录主机基本没问题。

4、检查打印卡, 换一块好的打印卡仍不联机, 则往下进行。

5、用病毒软件检测内存、软盘和硬盘是否有病毒, 因为病毒可能破坏打印口地址, 如无病毒, 往下进行。

6、检查稳压电源输出电压是否过低(约 175V), 打印机压纸卷筒钮是否过紧。

如果前 5 步无问题, 那么故障是由故障 6 引起的。这个故障一般是不易发现的, 由于供给打印机的电压较低, 驱动电流小, 而打印机卷筒钮较紧, 要使压纸卷筒卷动需要一定的转矩, 因此稳压器供给打印机的驱动电流不能驱动压纸卷筒卷动。屏幕打印正常是因为打印头往返 4 次才能打印一行, 走纸较慢, 克服压纸卷筒的转矩也相对较小。

三、解决办法

在上述条件下, 把一台同等型号但卷筒较松(正常)的打印机接到主机上, 则联机打印正常; 而把卷筒较紧的打印机接到电压正常的主机上, 则联机打印也正常。经过比较分析, 肯定是由于电压过低或卷筒钮过紧造成的。

解决的办法是:

1、提高稳压电源的输出电压, 达到约 220V。

2、调整压纸卷筒钮螺丝, 不要过紧。

通过以上办法解决后果然联机打印了。此种方法对其它型号的打印机和主机也能适用。

适用于一般 PC 机的交互式图形处理系统

(本刊讯) 中国国防科技大学一系(长沙市, 邮码: 410073)苏志军采用 BASICA 语言开发出一种适用于一般 PC 机的交互式图形处理软件, 可用于绘制尺寸准确的工程图, 提供了三种 A5 号图纸边框和标题栏, 并可采用两种方式标注尺寸, 软件具有图形打印、存盘、调用功能, 绘图方便, 准确, 符合工程人员习惯。颇具特色的是软件采用声音提示用户操作, 程序中极少用 Input 之类易于出同响应错误的语句而是代之以 Inkey \$ 变量输入, 从而可防止破坏屏幕画面。此外, 软件中设置了屏幕边界限制光标自动弹回功能, 设置了容错陷阱和多端出口保证程序运行的可靠性, 是一个适合一般计算机技术人员现状的应用软件。需此程序清单者请同苏志军本人联系。

苏志军致函本刊转告读者: 原将其开发的此种绘图软件免费奉献给本刊广大读者。长沙市内的读者可亲往苏志军处拷贝软件, 长沙市外的读者可将软盘成本费和邮资寄给苏志军本人。具体事宜请同苏志军联系。

版社。

[2]张载鸿编,《局部网络操作系统, DOS, 高级技术分析》, 国防工业出版社。

[3]李锦峰编著,《微机数据库、汉字 dBASE 一

■教程》, 陕西省电子学会。

[4]杨祥金、蔡庆生编著,《人工智能》, 科学技术文献出版社重庆分社。

人工智能与专家系统

吉林省计算机技术研究所 张鸿鸣 (长春 130012)

摘要 本文重点概述人工智能与专家系统,专家系统的应用及人工智能在数据库管理系统和CAD系统中的应用等。

关键词 人工智能 专家系统 知识库 推理机 图论理论 语义模型 推理网络

人工智能研究的一个重要目标,是提高计算机应用的灵巧性,使计算机能够做需要人的智能才能完成的工作。专家系统是人工智能领域的一个重要分支,是一个具有大量专门知识的程序系统,它应用人工智能技术,根据一个或多个人类专家提供的特殊领域知识,模拟人类专家的推理来解决那些需要人类专家解决的复杂问题。专家系统的出现和它的迅速发展使人工智能的理论和技术的渗透到了社会的许多领域。它与其它人工智能领域的重要差别之一是它使用了不精确推理,这使人们通常所用的不精确和不完备的知识得到了比较恰当地描述,它反映了客观世界中因果关系的不确定性。

本文重点概述人工智能与专家系统、专家系统的应用、人工智能在数据库管理系统和CAD系统中的应用等。

一、人工智能与专家系统

当前,在解决复杂的现实世界问题中,人工智能的理论和方法,主要是以专家系统的形式得到实际应用。专家系统可以说就是有智能因素的计算机程序系统,这种系统要使用大量人类专家水平的专业知识,解决比较困难的各种各样的实际问题。因此,人工智能的应用和专家系统的研制现在已经密不可分,而应用了人工智能技术的大量专家系统的出现,正在把计算机应用的范围更广泛地扩大到社会生活的各个方面。

1、人工智能 人工智能是计算机科学的一个分支,是用计算机程序系统来完成那些需要人类智能的任务。例如编写计算机程序,演算数学题,进行常识性推理,自然语言理解,甚至驾驶汽车,都需要智能。在过去的二十几年中,已经建立了若干能够完成这样一些任务的计算机系统。特别是有些计算机系统,能够诊断疾病,筹划复杂的有机化合物的综合,求解符号形式的微分方程,分析电子电路,对地质现象进行推断解释等。

人工智能领域诞生不足四十年,但它已经取得了很大的成绩。它的理论和技术正日趋成熟,应用领

域也在不断扩大,显示出了它的巨大生命力。在专家系统、自然语言处理、定理证明、机器人学和自动程序设计等方面都得到了应用。

2、专家系统 在人工智能领域中,专家系统是可以具体的科学、工程和医学等专业领域帮助解决复杂的现实世界问题的计算机程序系统。这些系统最大的特点在于它们使用了大批人类专家探明的领域知识。在实践中已经表明,对在一般的领域解决问题,这些系统是很有用的,专家系统的研究可导致人工智能应用,产生巨大经济和社会效益。目前,专家系统不仅限于解决科学问题,而且已经开始用于工业和企业界,并已渗透到社会的许多领域。

一个专家系统的组成,通常将可以分为两部分:知识库和推理机。知识库要以便于使用的方式存储为完成任务所需要的各种知识集合,包括常识的,书本的和经验的。推理机是一个控制机构,根据待处理的对象(如:患者,一组地图,一幅波形图)来决定用哪些知识。并运用知识进行推理,解决用户对系统提出的特定问题。推理有正向推理、反向推理和正反向混合推理。正向推理:由原始数据到结论;反向推理:先提出假设,然后去找支持假设的证据;正反向混合推理:先根据原始数据,通过正向推理帮助提出假设,再用反向推理进一步寻找支持假设的证据,反复这个过程。在这三种推理中,都有精确和不精确推理之分。精确推理:把领域知识表示成必然的因果关系和逻辑关系,推理的结论或是肯定的,或是否定的。也可以把可能性大于某个定值认为是肯定的。不精确推理:对于不肯定的证据给予某种“权”,对于多个证据要进行“权”组合。系统的附加部分还有数据库(证据库),用它来存储该领域初始证据的信息和推理过程中得到的各种中间信息。解释部分:是一组程序,负责对推理给出必要的解释,为用户了解推理过程,为维护系统提供方便,使用户容易接受。专家系统的一个重要特点是知识库与推理机分离,并且这两部分保持一定的独立性,一旦控制策略定了,系统的性能就随着知识库的丰富而增加,从而使系统

易于修改和扩充,这对不断提高系统的性能,是非常有益处的。

建立和维护知识库的主要问题是知识表示方式和知识获取能力。知识表示是指要在计算机的存储机构中表示知识,使能被计算机程序系统使用。需要表示的知识,不仅要有所完成任务领域中的大量事实和规律,而且要有类怎样解决问题的思想方法方面的知识。也就是说,系统中不仅需要表达出严格精确已经条理化的知识,而且需要表达出具有多义性或不确定性的人类“经验性”的知识。知识的形式化和如何运用是人工智能研究的重要课题。已经对许多种表示方式进行了研究和实验,例如象谓词演算、过程嵌入、语义网络、生成规则、框架和脚本等等。

知识获取部分为修改原有的知识,扩充新知识提供手段。知识获取已经被认为是构造实际专家系统的关键性的“瓶颈”问题。知识获取过程可以看作是一个专业知识从知识源到计算机程序的转移过程。知识源主要是人类专家,也可以是书本或其它资料。知识获取过程包括在知识库创建时识别出必要知识并将其形式化。建成的知识库经常会发现有错误或不完整,所以知识获取过程还包括对知识库的修改和扩充。当前,知识获取任务要求人类领域专家和计算机科学工作者密切合作并付出巨大的劳动。对如何减轻人类工作者在这一任务中的过重负担已经进行了不少研究,已有一些专门做知识获取工作的程序系统出现。

专家系统的推理机即控制机构的具体实现也有很多方式。通常,在专家系统中除了要实现利用明确规律进行的确定性推理之外,还要实现根据不充分数据或不完整知识进行的似合推理。事实上,人类在实际生活中经常根据近似正确的事实或规律进行似合推理而得出有用的结论。为了模拟人类的这种推理,许多专家系统采用了假设和规则引入,这称之为确定性因数或可信度的相应权数,从而把人类那种不精确的“预感性”的推理量化。这种作法使得许多专家系统确实具有了相当于人类专家水平的问题求解能力。在专家系统的交互过程中,认为哪些知识不合理,甚至错了,或者需要加进新知识,都要利用这个部分。开始可先建造一个简单的初级系统,在调试或运行过程中逐步修改旧的知识加进新的知识,使系统性能不断提高。

结合了人工智能技术的许多专家系统,为了提高性能,当然也可以采用任何其它领域的技术。有些专家系统解决问题主要是采用了其它技术,其中人工智能方法只是起到使系统更加灵活实用的作用。

但不管什么情况,人工智能技术应用的效果都是非常明显的。现在,能够体现出人工智能在科学技术中应用的实际专家系统已经很多。例如:

1) 诊断型,如医疗诊断,遗传咨询等;

2) 解释型,如地质资料推断解释的;

3) 预测型,如经济预测,战略部署等。这三种类型专家系统的共同特点是利用客观世界中那些不确定的因果关系和不精确推理来得出近乎合理的结论。

六十年代的中期,随着在数学、化学等科学实践工作中应用人工智能技术,专家系统开始出现。其后十几年中,由于人工智能研究比较侧重于把计算机应用扩大到更多领域,所以各种各样的专家系统在许多不同领域大量出现。近期以来,专家系统的研制特别活跃。人们已经要求专家系统不仅要能够以专家水平来解决实际问题,而且要能够做咨询服务,也就是要有对所解决问题的解释和论证能力,从而才能够为人类用户提供令人信服的建议或劝告。

二、专家系统的应用

专家系统领域自六十年代中期诞生以来,显示了它的巨大生命力,在许多科学技术领域中得到了广泛地应用。

(1) 知识获取系统 TEIRESIAS, 可用于帮助修改和扩充像 MYCIN 系统那样基于规则的知识库。这个系统本质上是一个知识库的编辑系统,也可以看作是一个以建议采纳方式工作的人工智能学习系统。系统使领域专家能够以非常自然的方式发现知识库中的问题并且能做出改正,而无需有计算机人员中间介入。系统的工作方式基本上是监控基于规则专家系统的实际运行,发现问题后可以回溯。这一能力使系统还可以做专家系统的解释和论证方面的工作。

(2) 启发式 DENDRAL 系统是首批专家系统之一。这个系统在化学分析领域中大显身手,可以根据未知有机物分子的质谱数据来推断分子结构。这个系统经过后来不断地发展和扩充,在完成复杂的分析质谱数据任务的一些方面已经达到或超过人类专家的水平。该系统大体可分为规则、生成和测试三个步骤。规划步骤根据分子的原子组成和质谱数据,推断出有机物分子必然要包含和不可能包含的亚结构,为生成步骤生成可能分子结构提供必须满足的约束条件。生成步骤根据图论理论,在满足约束条件的前提下,生成所有可能的作为待选的分子结构。测试步骤运用了表达有机物分子在质谱处理过程中发生碎裂规则,找出待选分子应该形成的质谱数据,与实际质谱数据比较,从而确定出最可能的分子结构。

本启发式系统及其扩展的实际使用,已经在有机化学分析领域中产生了深远地影响。

(3)生成程序 CONGEN 系统,可以生成有环分子结构。早期的启发式系统 DENDRAL 的结构生成部分还只能生成无环的分子结构,因此,被 CONGEN 系统所取代。这个生成程序还可以同化学工作者交互工作,允许指定任意来源所确定的关于结构的信息,为化学工作者灵活运用提供有多种方法。在实际应用中已经表明,在帮助确定有机物分子结构方面,这个系统完全是一个强有力的独立程序,并且已经受到化学工作者的喜爱。

(4)Meta-DENDRAL 系统。启发式 DENDRAL 系统测试部分需要使用质谱产生过程中分子碎裂的规则。形成这些规则,由于多种因素是很困难的工作,此系统被设计用来自动形成那些规则。因此,本系统是一个知识获取系统,并且也可以看作是一个以从实例中学习方式工作的人工智能学习系统。系统要求的输入是一组有机物分子中每个分子的已知结构和相应的质谱数据,同时还要有一个限制规则生成的领域的语义模型。在领域的语义模型中概括有关于质谱产生过程中分子碎裂的一些理论知识,从而为规则形成程序提供了一个引导。采用了统计和语义上约束的结合,是这个学习系统与只寻找数据重合而不评价重合意义的那些单纯的统计学习程序的不同之处。领域的语义模型还可以由化学工作者修改。这个系统形成规则的学习过程大致可以分为解释概括、规则生成及规则归并整理等三个阶段。解释概括阶段对每个分子按领域语义模型的约束找出可解释质谱数据的分子碎裂发生过程,从而形成单个分子的具体规则。规则生成阶段要依据前一步骤的结果形成一般的规则,它的工作是通过“生长”一颗碎裂规则的树,从一般的树开始,然后逐步填加上所要求的各种特性。规则归并整理阶段要对形成的规则提高水平,并且在以后的测试中还有新的实例输入,又可以适当地修改已有规则使其也能适用于新的情况。该系统在工作中已经成功地重新发现了若干类分子已知的质谱碎裂规则,尤其重要的是它还发现了对某些结构族的新规则。同时这个系统探讨的学习策略,也已经推广到其它的一些应用中去。

(5)专家系统 CRYSLIS,是化学分析方面在蛋白质结晶学领域中工作,可以较好地完成对电子云密度图的解释任务。这个系统采用了人工智能研究中探讨的“黑板”体系结构,把不同水平的假设和各种可利用的知识来源都组织起来。系统还采用事件驱动的控制结构,也就是保持一个当前的表,根据

这个事件表来决定程序的流向。

在有机化学的合成方面,特别有代表性的专家系统应当有 LHASA、SECS 和 SYNCHEM。这几个系统寻找较好的化学合成路线都是通过对“与或”目标树进行的启发式搜索。树的根节点是要合成的目的分子,其下一节点是可以通过化学反应生成它的分子,如此继续,终端节点是相当于原料的分子,连接各节点的分支是化学反应。组合几种分子形成化合物表达为“与”节点,生成化合物可供选择的方式表为“或”节点。于是可以看出影响这类系统找到较好合成路线的关键是表达化学反应知识库的组织和规模。这里的三个系统都具有大规模的化学反应知识库。LHASA 和 SECS 可以同化学工作者人机会话式工作,为化学工作者使用方便提供了一种简单而又灵活的专用语言,这使得在完成寻找合成路线任务方面与系统配合工作,确实可以达到比化学工作者单独工作要更好的效果。SYNCHEM 在搜索合成路线过程中考虑到反应效率、难度、花费等多方面因素,从而进一步探查了人工智能启发式搜索技术的使用和能力。可以预料对这类系统化学反应知识的进一步改进,必定能更大提高系统的性能,并有可能产生导致对化学反应分类的改进。

(6)大型数学问题求解系统 MACSYMA。它综合了关于符号积分方面早期人工智能工作和其它符号处理方面的工作,经过不断地进化发展已经成为具有较强数学符号处理能力的实用系统。这个系统规模很大,可以实行微分、积分、解方程组、级数展开、矩阵运算、向量代数及数量级分析等许多方面的数学操作。虽然这个系统的性质已经极大地数学化了,但是,在处理不适合于算法求解的数学问题方面,以及在对用户使用系统进行辅助教育方面,人工智能技术都提供了许多功能。

(7)在矿藏勘探方面工作的专家咨询系统 PROSPECTOR。它能够模拟有经验地质专家的推理,向实际勘探工作者提出高水平的有用建议。从理论上说,这个系统的主要功能是用特定情况的数据去匹配描述了不同矿床情况的模型,实际的实现比较复杂。不同矿床的地质模型需要由地质专家提供,对此提供有专门的编辑程序。系统把关于矿床模型的地质知识表示为一个推理网络,从用户处得到的证据要引起推理网络中各断言概率性值的传播。系统还要利用一般地质知识处理许多其它问题,在保证与用户进行方便的自然语言对话方面,系统利用了一个自然语言接口工具性程序 LIFER。这个系统预报矿床位置等实用能力已经被实际勘探所证实,现在这类专家系统已经广泛应用于矿床定位及石油

PC 系列微机总线与任天堂系列游戏机卡槽总线的接口

王让定 倪志新 冯百明

西北师范大学计算机开发应用室 (兰州 730070)

摘要 本文介绍了 PC 系列微机总线与任天堂系列卡槽总线的接口硬件及仿真软件的设计思想。该接口为游戏节目软件的开发及节目卡的制作提供了便利的开发环境。

关键词 接口 游戏节目 卡槽总线

一、引言

目前,任天堂系列家用电脑游戏机已广泛应用于千家万户。对家用电脑在我国的普及,对提高儿童的智力开发及全社会对高新技术产业的认识,起到了积极的推动作用。但随之而来的问题是许多家长担心小孩成天沉溺于游戏娱乐,耽误了学习,影响了青少年的健康成长。因此,许多科研机构和产业实体从这个现实出发,研制适合于我国国情的集游戏娱乐与学习为一体的家用电脑游戏机及教学节目。

勘探等方面。国内也有石油测井解释系统,在石油领域中发挥巨大的作用。

(8)医疗专家系统。肺功能诊断系统 PUFF、遗传学试验系统 MDLGEN、精神病诊疗系统 HEADMED 以及国内的各种医疗专家系统等等。

总之,以上列举了一些专家系统的应用实例,目的希望能对研究人工智能和专家系统起着抛砖引玉的作用。

三、人工智能在数据管理系统中的应用

这里应该重点强调,人工智能在数据库管理系统中有着广泛的应用。数据库是实现复杂数据处理的最新技术,在数据库管理中引入了人工智能技术,实现了有智能因素的信息存储和检索,是有极其重要意义的研究课题。人工智能技术应用于数据库管理系统大体有三个方面。首先,是应用人工智能关于自然语言理解的研究成果,改进数据与用户之间的接口。这方面已经有一些数据实现了用自然语言存取,有些可以处理自然语言中经常发生的省略、代替或拼写出错等问题,有些还能在对询问“理解”的基础上回答问题,即提供出比形式上正确的内容更多的回答。其次,在提高数据系统效率方面,可以考虑在接受用户询问后做询问优化使易于处理等等。最后,在扩展数据库系统能力方面,可以考虑对数据库保存信息进行推理,以便可以提供出对数据库中并没有明显存贮答案那样问题的回答等等。

任天堂系列家用电脑游戏机的游戏节目基本上全从国外进口,我国没有这方面的开发机构。要想开发游戏节目,需建立一套开发环境,利用普及型 PC 系列兼容机剖析现有游戏节目,编制适合我国家庭教育的教学软件,并将两者融为一体,制成节目卡,提供给广大少年儿童。

为进行上述工作,笔者开发了 PC 系列微机与任天堂系列游戏机游戏卡的连接接口,该接口插入 PC 微机总线插槽,并为游戏卡提供仿真的游戏机卡

四、人工智能在 CAD 系统中的应用

当前,人工智能 CAD 的研究,正在成为热门课题。尤其是在电子 CAD 领域中,人工智能技术的应用显得更为活跃。大规模集成电路 VLSI 设计自动化的智能方法已经出现,VLSI CAD 专家系统就是其中的代表。这些方法认为,人的知识是能掌握并可存储到一个专家系统的知识库中,知识形式的试探能处理 VLSI 设计问题的真正复杂性,以调整与修改设计行动,减少试探次数。又如数字系统式电路越来越复杂,故障诊断和测试已不在是传统算法所能解决得了的,必须探索新的智能方法。目前,主要侧重研究用专家系统方法的试验性测试产生软件,以及基于知识的内部自测试设计工具。另外,印刷电路板的布线算法,由于得不到令人满意的算法和花费太多的人机交互时间,人们开始采用人工智能技术来解决这一难题。因此,迫使人工智能的 PCB 布线方法的研究提到议事日程上来了。

总之,人工智能在科学技术中的应用是多方面的,体现出人工智能应用的各类专家系统正在许多领域出现。虽然从当前来看人工智能的理论和方法还都不是很成熟,但是作为计算应用研究的一个前沿,人工智能应用研究最近的进展相当迅速。完全可以确信,人工智能技术的广泛应用,必将在世界科学技术发展中起到越来越大的作用。

最小覆盖算法及正确性证明

中南财经大学信息系 金银秋(武昌 430064)

摘要 本文中的最小覆盖问题是指用一个圆覆盖平面上的若干个点。本文给出了求近似最小覆盖圆的最大距离算法。估计了它与最小覆盖圆的误差上界为 $\frac{1}{2}\sqrt{4-2\sqrt{3}}m$ (m 表示最大距离), 并运用谓词归约方法证明了算法的正确性。

关键词 算法 最小覆盖 程序正确性证明

一、最小覆盖问题

在模糊信息处理中, 需要确定 n 个点隶属于某圆的隶属程度。通常用最小覆盖圆来代替该圆。在计算机作图中, 对于给定的 n 个点, 需要用一个圆来覆盖它们, 标为同一种色彩。这也是最小覆盖问题, 最小覆盖问题在军事、卫星通讯中都有广泛的用途。

最小覆盖问题的定义: 给定平面上 n 个点 $A_1(x_1, y_1), A_2(x_2, y_2), \dots, A_n(x_n, y_n)$, 确定一个圆, 使得该圆是覆盖 n 个点的最小圆。为了简化求最小覆盖圆的计算过程, 但又不失实用性, 本文给出了用近似最小覆盖圆代替最小覆盖圆的算法, 并估计出它们之间的误差。

二、算法

1. 算法思想

采用区间逼近的方法, 把最小覆盖圆限制在二个较接近的同心圆中, 然后用较大的覆盖圆代替最小覆盖圆。为叙述方便, 我们用 r_A 表示最小覆盖圆, r_F 表示内圆, r_L 表示近似小的覆盖圆。

2. 算法步骤

(1) 求出 N 个点中两两之间距离最大的两点 A, B 。

(2) 以 A, B 两点连线的中点为 r_L, r_F 的圆心 C , 并计算圆心坐标和 r_F 圆的半径。

(3) 以 C 为圆心, 求出各点与 C 的距离中的最大距离, 并把该距离定为 r_L , 其相应的点为 P 。

3. 估计 $r_L - r_F$ 的上界

用 m 表示上述算法中距离最大点 A, B 两点的

距离, L 表示 P 与 A, B 距离中的最大距离。

命题 1: 由上述算法确定的 r_L 圆是覆盖圆

证明: 由算法第三步知, r_L 是各点与 C 的距离中的最大者。

因此, r_L 圆是覆盖圆。

命题 2: 根据上述算法所得到 $|r_L - r_F|$ 的上界为:

$$|r_L - r_F| \leq \frac{1}{2}\sqrt{4-2\sqrt{3}} \cdot m$$

证明: 因为 m 是两两点之间的最大距离, 所以有: $m \geq l$ 而 $m = 2r_F$

$$\text{得: } -2r_F \leq -L$$

$$\text{又: } 2r_L - 2r_F \leq 2r_L - L$$

$$\text{得: } r_L - r_F \leq \frac{2r_L - L}{2} = r_L - \frac{1}{2}L \quad (1)$$

当 m, L 一定时, 只有当 P 点在 \overline{AB} 的垂直平分线上时, 使 r_L 最大(如图 1)。

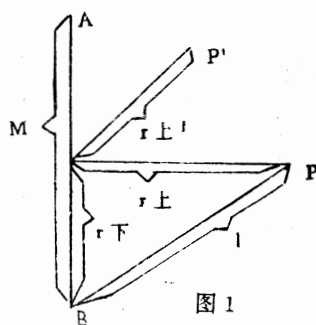


图 1

是写体号, 选择所要读写的存储映射体, 后序动作是

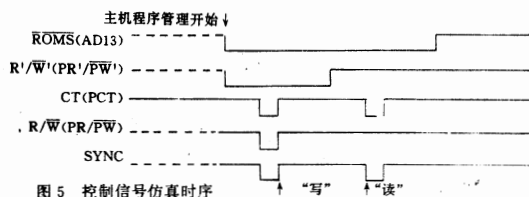


图 5 控制信号仿真时序

送读写地址, 并开始读写数据。

三、结语

任天堂系列游戏节目的开发是一项复杂的工作, 既需要有软、硬件开发环境, 又应具有丰富的软件知识及很强的编程能力。笔者在此方面进行了一点工作, 利用此接口开发了较简单的学习节目, 并加载到一些游戏节目之中, 收到了较好的效果并有一定的实用价值。

若P点不在垂直平分线上,总有 $r_L' < r_L$ 。

因此,我们考虑P点在垂直平分线上时:

$$|r_L - r_F| \leq \sqrt{L^2 - r_F^2} - \frac{1}{2}L \quad (2)$$

将 $r_F = \frac{1}{2}m$ 代入(2)左右端,并变形得:

$$(r_L - r_F)^2 \leq \frac{5}{4}L^2 - \frac{1}{4}m^2 - L \cdot \sqrt{L^2 - \frac{1}{4}m^2}$$

因为 $L \leq m$,用 m 代 L 得:

$$(r_L - r_F)^2 \leq \frac{2 - \sqrt{3}}{2}m^2$$

$$\text{所以: } |r_L - r_F| \leq \frac{1}{2}\sqrt{4 - 2\sqrt{3}}m \quad \text{证毕}$$

三、程序

1. 数据结构

A: ARRAY[1...N, 1...2] OF Real, 该数组用来表示n个结点的坐标。

C₁, C₂: Real, 表示圆心坐标。

r₁, r₂: Real, 表示r_F, r_L两圆的半径。

P, q: Integer, 指示距离最长的两点。

L, M: Real, 分别表示p, q两点的距离和I, J两点的距离。

I, J: Integer, 用来指示任意两点。

2. 程序(用PASCAL语言)

输入断言: A = ((x₁, y₁), (x₂, y₂), ..., (x_n, y_n))

$\wedge n > 0$

输出断言: (C₁, C₂) = r_L 圆心坐标 \wedge r = r_L 圆半径

程序: p := 1; q := 1; L := 0;

FOR I := 1 TO N-1 DO

begin

FOR J := I+1 TO N DO

begin

M₁ := SQRT(SQRT(A[I, 1] - A[J, 1]) + SQRT(A[I, 2] - A[J, 2]))

IF M > L THEN

begin L := M; P := I

q := J end

end

end

C₁ := (A[p, 1] + A[q, 1])/2;

C₂ := (A[p, 2] + A[q, 2])/2;

r₁ := L/2; r := r₁;

FOR I := 1 TO N DO

begin

L := SQRT(SQRT(A[I, 1] - C₁) + SQRT(A[I, 2] - C₂));

IF r < L THEN r := L;

end

四、程序正确性证明和复杂性分析

上述程序分别由求圆心坐标和半径两部份构成。第一部份是求两两点之间的最大距离L和相应

的两点p, q。第二部份利用求集合中最大值方法求半径。本文只对第一部份的正确性进行说明。

1. 生成断言

输入断言: 同前

输出断言需要描述n个点两两点之间距离的最大者。根据程序的特点, 我们定义如下一族集合。

令 $D_0 = \{1, 1\}$, $\overline{1, 1}$ 表示第1点的距离

$$D_K = \{K, K+1, K, K+2, \dots, K, N\} = \bigcup_{j=K+1}^N \{K, j\}$$

($N \geq K \geq 1$), $\overline{I, J}$ 表示I, J两点之间的距离。

$$D_N = \Phi$$

$$\text{输出断言: } L = \overline{p, q} = \max_{K=0}^N \bigcup D_K$$

2. 寻找循环不变式

采用从外向内的方法构造如下不变式

(1) 外循环不变式:

$$L = \overline{p, q} = \max_{K=0}^{I-1} \bigcup D_K$$

(2) 内循环不变式:

$$L = \overline{p, q} = \max_{K=1}^{J-1} \{ \bigcup_{K=1}^{J-1} \{ \overline{I, K} \} \bigcup_{K=0}^{I-1} D_K \}$$

3. 根据程序正确性证明法则, 程序的正确性证明归结为一组命题和一个待证程序的证明。

(1) 命题组:

$$L = \overline{p, q} = \max_{K=0}^{I-1} \bigcup D_K \wedge I = N \rightarrow L = \overline{p, q} = \max_{K=0}^N \bigcup D_K \quad (1)$$

$$\text{输入断言} \rightarrow 0 = \overline{1, 1} = \max_{K=0}^0 \bigcup D_K \quad (2)$$

$$L = \overline{p, q} = \max_{K=0}^{I-1} \bigcup D_K \rightarrow L = \overline{p, q} = \max_{K=1}^{I-1} \{ \bigcup_{K=1}^{I-1} \{ \overline{I, K} \} \bigcup_{K=0}^{I-1} D_K \} \} \quad (3)$$

$$L = \overline{p, q} = \max_{K=0}^{J-1} \{ \bigcup_{K=1}^{J-1} \{ \overline{I, K} \} \bigcup_{K=0}^{I-1} D_K \} \} \wedge J = N+1 \rightarrow L = \overline{p, q} = \max_{K=0}^N \bigcup D_K \quad (4)$$

以上4个命题中, (1)、(4)是反映当循环终止时, 应满足的命题, (1)式反映了当外循环终止时(即 $I=N$ 时)能推出的命题。在证明中注意到: 集合 $D_N = \Phi$; $\bigcup_{K=0}^{N-1} D_K = \bigcup_{K=0}^N D_K$ 。(4)式反映了当内循环终止时(即 $J=N+1$ 时)能推出p, q两点是距离集 $\bigcup_{K=0}^I D_K$ 中的距离最大者。利用 $\bigcup_{K=0}^I D_K = (\bigcup_{K=0}^{I-1} D_K) \cup D_I$, $D_I = \bigcup_{K=I}^N \{ \overline{I, K} \}$ 和 $\overline{I, I} = 0$, 得知命题(4)成立。(2)、(3)是反映进入循环前应满足的命题, 它们的正确性是很明显的。

(2) 一个特征程序:

$$\text{输入断言: } L = \overline{p, q} = \max_{K=1}^{J-1} \{ \bigcup_{K=1}^{J-1} \{ \overline{I, K} \} \bigcup_{K=0}^{I-1} D_K \}$$

$$\text{输出断言: } L = \overline{p, q} = \max_{K=1}^{J-1} \{ \bigcup_{K=1}^{J-1} \{ \overline{I, K} \} \bigcup_{K=0}^{I-1} D_K \}$$

程序: M := $\overline{I, J}$;

IF M > L THEN begin L := M; P := I; q := J end

根据程序正确性证明规则, 该程序正确性证明归结为如下命题的证明:

爱华 AH1488D 单色显示器行扭及无显示故障的排除

林业部白城林业学校计算机室 赵明生(吉林白城 137000)

一、行扭曲故障

1. 故障现象: 屏幕上的文字沿行的方向扭曲, 出现类似“S”形光栅(见附图)。

2. 分析与检查: 根据故障现象分析, 本机采用桥式整流方式, 假如桥式整流中有一臂二极管正向电阻变大, 使波纹接近 50Hz, 光栅就可能出现“S”扭曲, 一般还要伴随着扫描线稀疏的变化。其原因是纹波电压作用到垂直扫描系统后, 使光栅上一部分扫描线稀疏, 另一部分稠密。另外, 滤波电容容量变小或损耗增大, 经该电容滤波后的直流电压作用到水平扫描系统中, 也会产生此类故障(此时波纹频率为 100Hz, 每屏扭曲次数应为前述的 2 倍)。本机经替换滤波电容后故障仍然存在, 在检查整流二极管时, 发现确实有一个二极管 D501 (IN5401) 正向电阻接近 70KΩ, 经更换该二极管显示恢复正常。

二、无显示故障

1. 故障现象: 计算机启动后无光标和字符显示, 显示器指示灯不亮, 并烧保险丝。

2. 分析与检查: 拆开显示器检查发现, 交流保险丝已熔断, 更换新的保险丝后开机又迅速熔断。经分析, 烧保险的原因, 一方面是由于负载过重造成的; 另一方面可能是交流回路(变压器初级回路)的阻抗太小造成的。为了确定故障所在位置, 先将电源变压器次级插头拔下, 换一个新的保险丝, 开机仍烧保险。说明交流回路一定存在问题, 拆下电源变压器初级接线端, 用 DT-830 型数字表测量变压器初级直流电阻为 13.5Ω, 可以肯定阻值太小, 再测量正常的显示器的变压器初级电阻为 72.8Ω。这说明电源变压器初级线圈有短路现象, 使阻抗减小, 导致初级电流增大而烧保险。因为电源变压器初级线圈在开始时可能有较少一部分发生短路, 初级电流照正常有所增加, 变压器本身热量随之

增加, 逐步使漆包线的绝缘层老化, 使短路匝数越来越多, 最终使初级电流超过保险丝的容量而熔断, 使显示器无法供电而造成不显示的故障。

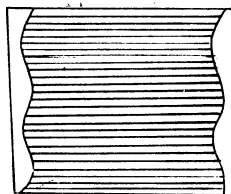
由于在市场上很难买到原配变压器, 所以只好买一代用品(代用品要考虑初、次级的电压及变压器的功率, 初级电压为 220V, 次级电压为 18V, 功率为 25~30W, 还要考虑体积和安装位置)。更换后开机试之, 基本正常; 但显示器又出现行扭(类似上例故障)的现象, 据了解本机在出现故障前无此现象, 所以多数是因为电源变压器没有屏蔽, 使该变压器泄漏 50Hz 电磁干扰造成的。在通电状态下观察, 用手移动变压器使之逐步远离显示器主电路板, 此现象逐步消失。说明扭曲就是由于变压器屏蔽不良引起的。解决的办法: 把原变压器的屏蔽罩拆下, 套在新变压器上, 并将开口处向机壳方向(向外), 再开机试之故障消除。

综上所述:

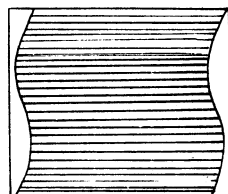
1. 该机造成无显示、烧保险故障的原因是由于电源变压器初级短路造成的。

2. “S”形扭曲一般有以下三种可能:

- ①滤波电容容量减少或损耗增大;
- ②桥式整流中有一臂二极管正向电阻变大;
- ③电源变压器屏蔽不良。



纹波频率为 100 赫



纹波频率为 50 赫

$$L = \overline{p, q} = \max \left\{ \left(\bigcup_{K=1}^{J-1} \{ \overline{I, K} \} \right) \cup \left(\bigcup_{K=0}^{I-1} D_K \right) \rightarrow \langle \overline{I, J} \rangle > L \rightarrow \overline{I, J} = \right.$$

$$\left. \overline{I, J} = \max \left\{ \left(\bigcup_{K=1}^J \{ \overline{I, K} \} \right) \cup \left(\bigcup_{K=0}^{I-1} D_K \right) \right\} \quad (6)$$

$$L = \overline{p, q} = \max \left\{ \left(\bigcup_{K=1}^{J-1} \{ \overline{I, K} \} \right) \cup \left(\bigcup_{K=0}^{I-1} D_K \right) \rightarrow \langle \overline{I, J} \rangle \leq L \rightarrow L = \right.$$

$$\left. \overline{p, q} = \max \left\{ \left(\bigcup_{K=1}^J \{ \overline{I, K} \} \right) \cup \left(\bigcup_{K=0}^{I-1} D_K \right) \right\} \quad (7)$$

(6)式说明了如下事实, 当 I, J 两点距离 $\overline{I, J}$ 大于 L 时, $\overline{I, J}$ 是所给集合中的最大距离。在证明过程中注

意到 $\bigcup_{K=1}^J \{ \overline{I, K} \} = \bigcup_{K=1}^{J-1} \{ \overline{I, K} \} \cup \{ \overline{I, J} \}$, 和 $\overline{I, J} > L$ 的前提下, (7)式说明了当 I, J 两点距离 $\overline{I, J}$ 比 L 小时, $L = \overline{p, q}$ 仍是最大距离。其证明过程与(6)式相同。

从程序知, 求圆心坐标共进行了 $\frac{1}{2} N(N-1)$ 次比较, 求近似最小覆盖圆半径共进行了 N 次比较, 所以总的比较次数为 $\frac{1}{2} N(N-1) + N$, 故算法复杂性为 $O(N^2)$ 。

陶瓷生产线计算机实时控制系统

梧州市计算机技术服务中心 唐 敏(543002)

梧州市工业技术设计研究所 唐 明(543000)

摘要 本文介绍了一个陶瓷生产线微机实时控制系统。系统采用 SM286 微机代替常规的仪表监控系统,实现了对多点炉温、多台电机转速及施釉过程的自动控制。

关键词 陶瓷生产线 计算机控制 多变量系统

一、概述

本装置所控制的陶瓷生产线的工艺流程为:原材料处理—冲压成型—施釉—烧成。工艺要求流水线上的五台滑差电机按一定比例协调运行,且瓷砖的传送速度要随辊道窑的温度变化。本系统使用 SM286 计算机对陶瓷生产线上的电机转速、炉温及施釉过程实行自动控制,可分别调整各分部电机的控制特性,使各电机间按一定的比率及偏差运行,在此基础上可按工艺要求手动或自动实现全线同时升速或减速。由于使用计算机对多个温度变量及传送速度实行近似的解耦控制,使得对温度设定值的跟踪性能优于由模拟调节器组成的多个独立单环控制系统。本系统从 91 年 6 月正式投入运行,温控误差 $\leq \pm 2.5^\circ\text{C}$,速度误差 $< 5\%$,产品质量提高,节油率达 3.5%。

二、燃油辊道窑动态模型分析及系统解耦

燃油辊道窑是一个多变量系统,其主要控制目标是炉膛温度及产品传送速度。炉温与风量、喷油量、烟囱开度及产品流量有关,其不可测的干扰多,很难建立精确的动态数学模型。而炉窑系统一般都是时间常数大,对参数反应迟钝。因此,在建立动态模型时,不妨忽略小时间常数的影响,从宏观规律上建立一个简单的数学模型,以便于调节器参数的计算和整定。另外,由于系统的时间常数大,当采样时间足够短时,断续控制方式趋近于连续控制方式,因而可以将模拟量的计算方法用于数字控制系统中。电烘炉的开环传递函数为

$$W_d(s) = \frac{K_d}{T_1 s + 1} e^{-T_2 s} \quad (1)$$

利用劳劳级数展开化简可得

$$W_d(s) = \frac{K_d}{T_1 s + 1} e^{-T_2 s} \approx \frac{K_d}{(T_1 s + 1)(T_2 s + 1)} \quad (2)$$

$T_1 \gg T_2$

基于窑炉本身的特点,为使系统调节过程平稳,尽量减少超调量,要求系统的过渡过程接近于单调的衰减过程,决定选用滞后校正器—PI 型调节器,将系统校正成典 I 型系统,PI 调节器的传递函数是

$$W_i = K_D \frac{T_1 s + 1}{T_1 s} \quad (3)$$

校正后系统的开环传递函数为

$$W'(s) = W_i(s) W_d(s) = \frac{K_P K_D}{T_1} \frac{1}{s(T_2 s + 1)} \quad (4)$$

炉温及速度控制系统的动态结构图如图 1 所示

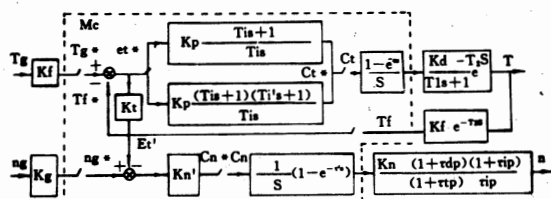


图 1 炉温及速度控制系统的动态结构

陶瓷烘炉是一个复杂的多变量系统,各温区温度的变化,对其它温区的温度亦有影响,即有耦合现象,若不实行解耦,则炉温的波动幅度难以减少。设 T_1 — T_3 为炉内三个温区的温度, N_1 — N_3 为相应温区油阀的开度,通过阶跃响应试验得到耦合对象的静态特性矩阵为^{[2]P99}

$$K = \begin{bmatrix} K_{11} & K_{12} & K_{13} \\ K_{21} & K_{22} & K_{23} \\ K_{31} & K_{32} & K_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0.125 & 1 & 0 \\ 0 & 0.125 & 1 \end{bmatrix} \quad (5)$$

式中 K_{ij} 表示被控制变量 T_i 与操作变量 N_j 的藕合关系。当各个控制通道与耦合支路的传递函数都有比较相近的动态特性时,或者这些通道的动态部份的等效时间常数均较小时,可以不考虑动态部分的影响,用静态解耦法即可获得相当好的效果,利用式(5)可得解耦矩阵^{[2]P97}

$$D = K^{-1} K_A = \frac{1}{|K|} K^* K_A = \begin{bmatrix} 1 & 0 & 0 \\ -0.125 & 1 & 0 \\ 0 & -0.125 & 1 \end{bmatrix} \quad (6)$$

其中 K^{-1} 为 K 的逆矩阵, K^* 为 K 的伴随矩阵, K_A 为 K 的对角矩阵。计算机实现解耦控制不需要增加任何硬件,只需要在调节模型的输出程序 N 后插入一个解耦网络子程序 D ,即得最终输出 N'

$$\begin{bmatrix} N_1' \\ N_2' \\ N_3' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -0.125 & 1 & 0 \\ 0 & -0.125 & 1 \end{bmatrix} \begin{bmatrix} N_1 \\ N_2 \\ N_3 \end{bmatrix} \quad (7)$$

炉温控制静态解耦系统方框如图2所示。

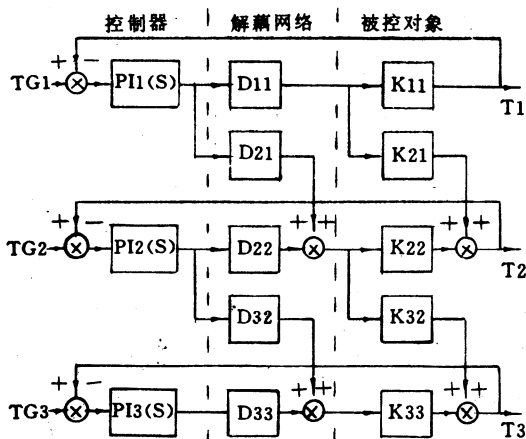


图2 炉温控制静态解耦系统方框

三、系统硬件分析

本系统由稳压电源、温度控制回路、速度控制回路、自动施釉控制回路、微型计算机及相应的接口电路等组成,系统硬件框图如图3所示。

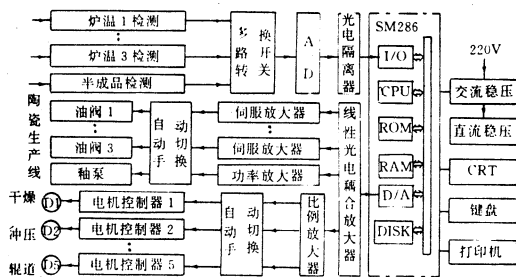


图3 系统硬件框图

辊道窑的各点温度值由热电偶检测并经温度变送器放大,经A/D转换后送至SM286计算机内进行数字滤波、比较、PID运算、解耦运算等,所得的油阀开度值经D/A转换后输出至伺服放大器,通过角行程执行器控制油阀的开度,使炉温自动跟踪给定值,一旦温度超限即在CRT上发出声光报警并显示相关数据。各油阀开度经D/A转换后反馈给计算机,实现闭环控制。

计算机根据速度给定值及实际温度与温度给定值的偏差自动调整产品传送速度,以实现烧成温度与时间的最佳配合。采用了反射式红外传感器检测瓷砖,不受环境光线的干扰,并可将瓷砖与辊道区别

开来。检测到的信号送入计算机经延时后控制施釉泵及施釉阀,实现自动施釉。

四、系统软件分析

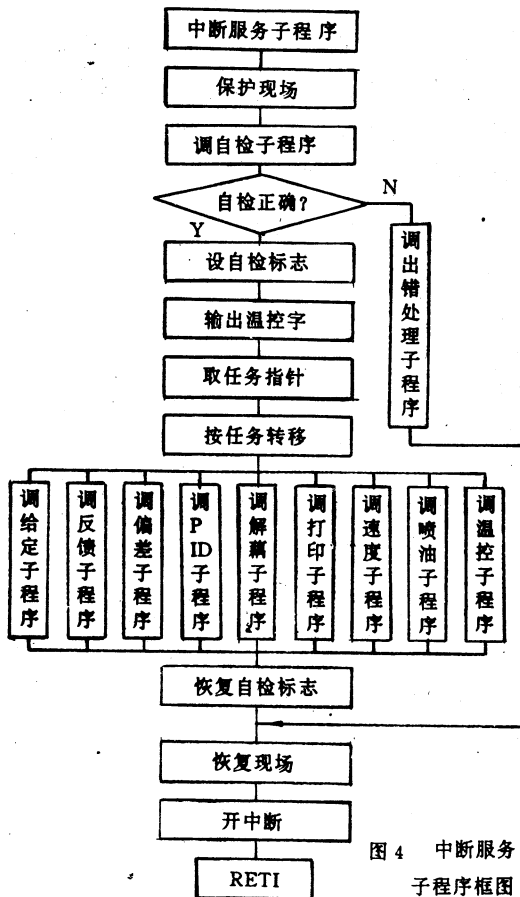


图4 中断服务子程序框图

系统软件采用Turbo-Basic与汇编语言混合编程,其中实时控制部分采用汇编语言编程,管理部分采用Basic语言,整个系统软件由控制主程序、中断服务子程序、管理主程序和子程序库组成。以286微机为核心的控制系统用户界面良好,全下拉式菜单设计易于普通工人操作,屏幕上可显示陶瓷生产线工艺流程图、各工艺参数直方图及变化趋势图,所有控制参数均可在线修改,生产数据自动存盘并可在线打印。系统采用积分分离的PID算法,当偏差大于某一值时使积分系数为零,且调节器的输出具有上下限限幅,既可减少超调量,又可达到积分校正的预期效果。中断服务子程序框图如图4所示。

参考文献

- [1]刘植桢,郭木沙,何克忠:《计算机控制》,第1版,北京,清华大学出版社,1981.
- [2]王永初:《解耦控制系统》,第1版,成都,四川科技出版社,1985.

工作站间调用:一种局部网应用开发方法

哈尔滨船舶工程学院计算机科学系 王慧强(150001)

摘要 资源共享与站间通讯是计算机联网的主要目的。根据局部网特点,借助于虚拟卷思想,本文提出了把上述两种应用融为一体的新的应用方式—工作站间调用。首先给出了局部网虚拟卷思想,然后详细地阐述了站间调用的设计实现方法和安全性考虑。

关键词 局部网 虚拟卷 站间调用。

一、引言

资源共享与工作站间通讯是计算机联网的主要目的。本文根据微机局部网的特点,借助于虚拟卷概念,提出了把上述两种应用结合为一体的新的应用方式—工作站间调用。

所谓站间调用是指:任给一个局部网上的工作站 U_i ,应能根据需求和给定的访问权限,利用另一个工作站 U_j 的软件访问 U_j 上的数据,因此,技术上讲,站间调用是软件共享与数据共享的结合;应用上讲是为工作站间的信息交流提供一种手段,从而,可以认为是一种通讯方式。本文首先给出了局部网虚拟卷概念,在此基础上,通过对局部网主要应用环境—办公室信息处理的分析,详细阐述站间调用的设计实现方式和安全性考虑。

二、虚拟卷概念[1]

微机局部网对网络硬盘(文件服务器)的管理往往是通过一个专门程序(如PLAN网的NET.EXE)来实现的,概括地讲,这个专门程序主要的功能有两个:

1、虚拟卷维护功能:用于虚拟卷的建立,删除,命名和保护等;

2、虚拟卷的访问功能:用把虚拟卷映象到用户指定的逻辑驱动器上,并可授予不同的访问权限。

虚拟卷是微机局部网的一个重要概念,是借助于网络硬盘进行软件与数据资源共享的桥梁,也是实现网络通讯服务如电子邮件的基础,虚拟卷相当于一个PC磁盘驱动器,所以又叫虚拟软盘,逻辑驱动器等等。

虚拟卷可以通过特定的网络服务器命令对共享大硬盘的划分而建立起来,使得其它任何工作站能象使用本地驱动器那样使用它们,从而也使得任一工作站的磁盘驱动器个数由原来的A,B,C三个增加到A—Z若干个不等(如PLAN—5000可至Z,共26个)。

以PLAN—5000网为例,虚拟卷思想可形象地表示成如图1。

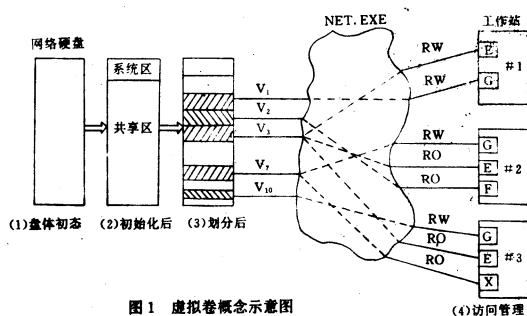


图1 虚拟卷概念示意图

由图1,可以看出:

1. 同一卷在不同站上可映象为同一驱动器号上,如 V_3 ,这是共享卷的典型用法;
2. 同一卷在不同站上可映象为不同的驱动器号上,如图中的 V_1 ;
3. 不同卷在不同站上可映象为同一个驱动器号,如 V_1, V_7, V_{10} ;
4. 同一卷或不同卷在映象时可授予不同工作站不同的访问权限。

但无论何种情况,对虚拟卷的操作,最后都落实到工作站所对应的卷上。

三、设计与实现方法

政府机关的办公室之间存在着错综复杂的上下级及同级之间的业务关系。但经仔细分析你就可以发现,这种联系的实质是信息联系,更进一步地分析,实际就是某些信息项(数据项)之间的联系。这里表现为信息项的交叉使用,职能的转接,互相补充,协调和信息共享等。因此,为实现站间交叉调用功能,第一步必须做到心中有数,进行用户调查和信息分析。为此,要使用系统分析中的信息交叉图和工作站资源需求调查表[2]等,在此基础上进行分析整

理。这里,我们提出了用横向联系矩阵与四级互访权限矩阵的方法来描述和确定工作站间的信息联系及其联系类型,从而把站间调用的实施转化为上述矩阵的实施。

一般地,设一个系统有 N 个用户, U_1, U_2, \dots, U_n , 对任意的 U_i , 有 N_i 个数据文件 $Fi_1, Fi_2, \dots, Fi_{Ni}$, 对任意的 Fi_l , 有 M_l 个数据项 $Fi_{l1}, Fi_{l2}, \dots, Fi_{lm_l}$, 则考查任意的 U_i, U_j 是否有联系, 只要考查 U_j 对任意一个 Fi_{lk} 及 U_i 对任一 Fi_{lp} 是否使用即可, 为此, 可建立一矩阵来描述这种联系, 如图 2 所示, 我们称之为横向联系矩阵。

		U_1	U_2	\dots	U_i	\dots	U_n
U_1	\dots						
U_2	\dots						
\vdots							
U_i	Fi_1	\dots					
\vdots	Fi_l	0	1		1		0
\vdots	Fi_k	0	0		1		1
\vdots	Fi_{m_l}	1	0		1		1
U_{n-1}	Fi_{ni}	\dots					
U_n							

图 2 横向联系矩阵

这样,我们可在每个交叉的格上,对应两种不同的情况标上两种不同的值:为 1 时有联系;为 0 时无联系。但问题并没有完全解决,上述矩阵虽较精确地表达了站间联系,却不能刻画联系的性质和类型,即对信息访问的权利和形式,即是只读(RO),还是可读可写(RW),还是无任何联系(-)等等。为此,还须建立互访权限矩阵。而且依据信息组织级别及设计方便,应把访问权限分四级:工作站级,卷级,文件级和数据项级。这里,我们给出本系统的站级互访权限矩阵,如图 3 所示。

	U_1	U_2	\dots	U_n
U_1	Rw	Rw	\dots	Ro
U_2	-	Rw	\dots	Ro
\dots				
U_n	-	Ro	\dots	Rw

图 3 互访权限矩阵

有了横向联系矩阵和互访权限矩阵,就可以考虑站间调用的实施了,亦即用程序来描述实现上述两矩阵。站间调用的实施分两级实现,第一级是网络系统级,第二级是工作站程序员级。第一级实现主要解决把相互调用的数据及支持程序以可行和方便的形式提供应用程序员,具体地讲就是虚拟卷的安排,方法如下:

1. 每个参加站间调用的站都设有一个存放调用数据和程序的虚拟卷,设为 D_x , 简称调用卷。

2. 为做到系统管理过程对用户的透明性,所有虚拟卷到逻辑驱动器的映射都由网上的工作站自举程序自动完成,无须用户干预。

3. 对所有的站来讲,属于本站的 D_x 映射到本站时,用某一个彼此相同的逻辑驱动器名,而映射到它站时,又固定使用另外一个驱动器名。例如:设某局部网有四个站,分别设调用卷为 D_1, D_2, D_3, D_4 , 则映射情况可如图 4 所示。

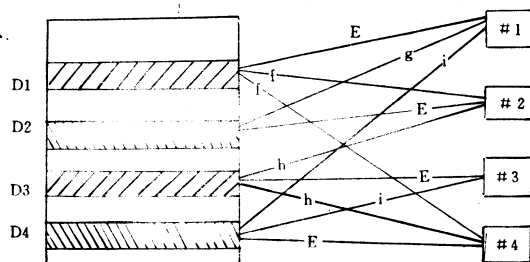
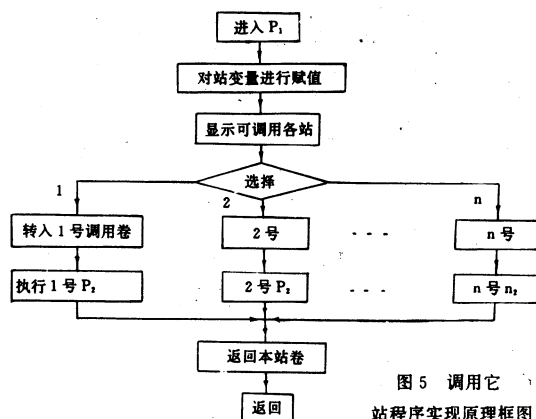


图 4 调用卷映射方法



这样,就可以统一的方式和方便灵活的途径来使用调用卷。因此,在这种情况下,对任意的工作站来讲,对自己的数据处理,就是对逻辑盘(E)进行处理;而对某个另外工作站如 #4 进行处理,就是对 i 号逻辑驱动器进行处理。

第二级实现是工作站程序员级。在第一级实现的基础上,给出有关说明及约定,由程序员们共同实现之,基本方法是:

1. 每站提供一个调用它站的程序,存于本站的 D_x 中。主要功能是,根据图 2 说明的可以调用的站建立相应的站间调用支持程序,并且根据互访权限

软件重用的模板方法

北京航空航天大学软件工程研究所 郭江 (100083)

摘要 本文描述了软件重用的模板方法。这种方法主要是将程序设计过程分为两个独立部分即算法说明和实现说明。在表达算法时并不必须确定算法所涉及的数据类型和表示。通过推迟对相关数据的选择,软件模板即算法说明就可以在应用中重用。也就是将同样的算法用到不同的数据类型之上。

关键词 模板 重用 强制语言

一、概述

传统上,强制语言通常不允许程序员不确定数

据的类型和表示就去实现算法。例如,对普通元素集进行排序的程序需要预先说明要排序数据的类型和

矩阵,针对不同的站,启动不同的支持程序。

2. 每站提供一个支持它站站间调用的程序,其主要功能是检查调用的合法性;对不同的站提供不同的服务,保证正确返回调用者主菜单。

如设上述两程序分别取名为 P1 和 P2,则其实现原理框图如图 5 和图 6 所示。

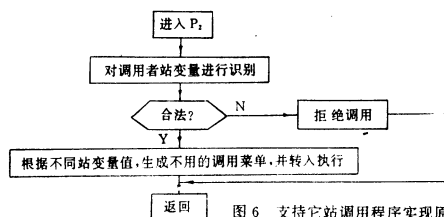


图6 支持它站调用程序实现原理框图

四、安全性考虑

最后,着重谈谈本系统数据安全性考虑,从以上论述可以看出,引入站间调用的确为我们带来应用上的极大方便,但却同时给我们带来了一个严重的问题,就是信息的安全,主要是如何防止信息泄露问题。

首先我们考查一下办公信息的开放级别与类型,然后说明我们所采用的相应对策。通过调查分析,机关办公信息的开放形式有以下几种类型:

1. 公开数据:对所有人员开放;
2. 部分公开数据:只对一部分工作人员开放,如本机构或本机构部分人员开放;
3. 不公开数据:只供某个人或某几个指定的人使用。

显然,类型1没有保密安全问题。因此,这里主

要就集中在类型2,3两类数据的安全问题。而且对于类型2来讲,还可进一步划分为对谁公开,公开多少问题。如果把上述信息形式落实到网络的虚拟卷上,则就变成如下情况:

公共卷:(1)共享卷

小组卷:(2)此卷可供哪些站访问,访问方式?

(3)此卷中哪些文件可供哪些站访问,访问方式?

(4)此卷的任一文件的任一数据项可供哪些站访问?访问方式?

私人卷:(5)工作站独占卷

对上述五种情况,采用的相应安全对策如表1所示。

表1,五种情况的安全对策

情况(1)	建卷授权公共共享读写
情况(2)	建卷授权公共,小组只读,私人读写,并设小组,私人两级口令
情况(3)	由应用程序中通过站变量与选单生成法控制
情况(4)	
情况(5)	建卷授权独占读写与分级口令

总之,本系统使用的安全对策是确认、授权与选单生成三种方法相结合。

五、结束语

上述设计思想被成功地应用于某大型信息系统的研制,该系统已通过省级鉴定,投入运行,并受到好评。

参考文献

- [1]李增智,《局部计算机网络》,陕西科技出版社,1986.
- [2]王慧强,“局域网资源共享设计”,《应用科技》,待发表,1992.

表示。很明显,预先说明数据的类型和表示可能适合于一个应用而不适合于另一个应用。

本文描述了一种有助于软件重用的程序设计方法。在这个方法中,程序设计过程分解为两个独立的部分:算法说明和实现说明。在表达算法时数据的类型和表示并不确定,而只是表达算法的最本质特性。通过推迟对相关数据的选择,称作软件模板的算法说明可以在应用中进行重用,即使用同样的算法而使用不同的数据类型。例如,在一个排序模板中,要排序的元素类型并不确定,以便在一个应用中可用模板来对整数排序而在另一个应用中用来对字符串排序。

模板定义在抽象数据类型的基础之上,而这些数据类型在强制语言中则是由程序员来实现的,但程序员并不一定是模板设计者。一个实现者产生一个数据类型可重用的实现库,如集合、序列、树等一被普遍用在公式化算法之中的类型。而一个模板设计者使用这些数据类型为标准算法产生一个模板库。在使用一个模板时,程序员从库中选择适当的数据类型实现。如果有若干可以选择使用的实现方式,那么可以根据应用的特定需求如程序的性能或功能来选择一种。这样,模板就能被自动转换成为一个实例,即转换成一个裁剪好的程序。通过将算法模板说明的数据类型的约束不同的实现方式之上,可以用几种不同的方式来实现算法,模板方法如图1。

为了描述的方便,我们定义一个称为 Sort 的模板,它将一个序列中的元素进行排序。Sort 将起输入序列中的每个元素插入到一个二叉树中并保持二叉树的性质不变。即,对任何带有左子树 t_1 和右子树 t_2 的节点 n 来说, t_1 中所包含的值“小于” n 中的值,而 n 中的值又“小于”或“等于” t_2 中所包含的值。模板 Sort 使用了两个称作 Seq(序列)和 Bintree(二叉树)的抽象数据类型。

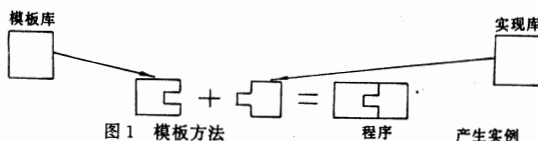


图1 模板方法

类型 Seq 的值或者是空序列 nil,或者是一个值和另一个序列用 Cons(连接)构造出的序列。一个 Bintree 值被定义作 niltree,一个叶子节点是由一个带有 leaf 的值构造出来的,而一个内节点则是由一个值和两个带有 node 的子树形成的。Sort 的定义在要排序的元素类型中具有一般性,如图2。其中操作

符“<”是重载的,即它可以应用到各种类型的操作数上,而对每种类型它的定义都不相同。用这种方法,Sort 隐含地参数化在一个序关系之中。

```
sort(nil) = niltree
Sort(cons(x,y)) = instrt(x,sort(y))
insert(n,niltree) = leaf(n)
insert(n,leaf(m)) =
  if n < m then node(leaf(n),m,niltree)
  else node(niltree,m,leaf(n))
insert(n,node(l,m,r)) =
  if (n < m) then node(insert(n,l),m,r)
  else node(l,m,insert(n,r))
```

图2. 模板 Sort

为了使用模板 Sort,程序员要确定一个产生实例指令。在程序中,该指令将对数据类型 seq 和 Bintree 的实现方式进行转换。由于 Sort 具有一般性,因而该指令必须确定要排序的元素类型。例如,假设我们希望用 Sort 来产生一个 C 函数的实例,用来对字符串数组进行排序,而我们的库中包含了称作 Carray 和 Ctree 的 Seq 和 Bintree 的 C 实现。为了能在一个更广泛的上下文中使用 Carray,必须在数组大小方面对它进行参数化。如果我们选择数组大小为 128,那么用图3中的指令就能用 Sort 来产生出想要的函数体实例。

```
Assign(y,Sortx)Where
x:Seq(String)/Carray(128)
y:Bintree(string)/Ctree
Endwhere
```

图3. 用 C 语言排序

图3中的指令表明用模板 Sort 来产生实例,那么就须从 x 获得输入,并将其结果约束到 y 上。变量 x 和 y 的类型由 where 子句中实现符号“/”来给出,其中 x 标记了一个由 Carray 实现的串的序列,而 y 标记了一个由 Ctree 实现的串的 Bintree。也可以用模板 Sort 来产生一个 Pascal 过程实例,用它来对一个整数数组进行排序,如果我们的库中包含了 Seq 和 Bintree 的 Pascal 实现 Pascalarray 和 Pascaltree,那么用图4中的指令就可以用 Sort 来产生想要的过程体实例。

```
Assign(y,Sortx)Where
x:Seq(Int)/Pascalarray(128)
y:Bintree(Int)/Pascaltree
Endwhere
```

图4. 用 Pascal 排序

二、模板方法

模板方法允许程序员不预先说明数据类型和表示就能表达算法。这里主要是因为模板是定义在重

载的抽象数据类型的值之上。一个抽象数据类型是一个由操作符和载体(操作符可以操作的值的集合)的有限集组成的代数。一个抽象数据类型具有的优势是载体的准确表达对类型的具体应用是隐藏的,即对那些调用其操作符的程序来说是不可见的。

在传统上,程序设计语言对抽象数据类型的实现,主要是允许程序员为隐含表示的数据类型的实现定义接口。抽象类型的应用必须用这些接口来表达,而这些接口是用于表明抽象的操作符是应解释作语句还是解释作表达式。因此,在一个应用之中,并不总能将抽象操作符作为数学意义上的函数来对待,这是模板方法与传统抽象数据类型使用的不同之处。抽象操作符总是被解释作函数。使用构造和递归的方式来定义新函数的实现是由低层的抽象操作符自动推导出来的。

2.1 定义模板 一个算法说明应当拥有几个特性。为了得到计算复杂性及其执行性能,算法在描述怎样计算时必须是构造性的。而且,算法说明不应受副作用的影响,因为如果计算受影响,那么就会构成计算模型的特例。最后,它应能进行转换和形式解释。为了达到这些特性,模板可用应用的标志来定义。

模板是一个类型的一阶谓词函数定义的集合。一阶谓词函数并不以函数为输入输出。而且,在模板中所使用的操作符除了条件(if-then-else)之外均被严格解释,即操作符的操作数在应用时都是作全面评估的。构成模板的每个函数都有一个能从函数的形式及其子表达式的类型上推导出来的类型。例如,模板 Sort, 便有类型 Seq(alpha) → Bintree(alpha)。其中 alpha 是一个全称量词类型变量,因此,Sort 可以认为是一个无限单个分类排序函数的一个有限表示。例如,这个类包含了对整数、串等排序的函数。

2.2 实现抽象数据类型 实现者的工作是为一个抽象数据类型的值寻找一种数据表示,然后用它来实现有效的类型运算符并确保其正确性。数据类型可用在任何强制语言中,如用 C 或 Pascal 实现,它们都有特定的说明语句可以用来定义类型。用模板产生实例的过程就是用实现各种抽象数据类型操作符的强制语言代码段来构成程序的过程。抽象数据类型的值是用实现语言中的具体的值来表示的。当用 R 来作为抽象数据类型 T 的数据表示时,可以使用如下的说明形式: R[]:T[Tc]。其中 Tc 是实现语言的一个具体的类型,符号[和]限定了将由实例产生器作为宏定义对待的文本。在这种方法中,实例产生器保持了特定实现语言的独立性。例如,考虑说

明:

```
Repint[],Int[integer]
```

Repint 就是一个数据表示。其中抽象数据类型 Int 是由 Pascal 的具体类型 integer 所表示的。如果一个说明中所确定的具体类型不是在实现语言中预定义的。那么它的定义必须在一个块中也由[和]来限定,如图 5。

```
Ctree(),Bintree(alpha)
[struct $_tnode{alphaval;
struct $_tnode * lsucc,* rsucc;
};]
[struct. $_tnode *]
```

图 5. Ctree 的说明

如果没有能力去表达重载的类型,实现库就会随着单个类型的实现而非常迅速地增长。例如一个库中可能包含了 Bintree(int)、Bintree(string)等的实现。对于这种情况可以通过使用重载类型如 Bintree(alpha)来防止。其中 alpha 是一个类型变量。

重载类型的表示不同于通过具体类型说明使用而产生的单个类型表示,这种单个类型的表示就是由模板实例产生者所创实例。如果一个具体的类型是使用一个类型变量来定义的,那么该类型就变成了一个类型说明。例如,在图 5 中说明的 Ctree 表示就是重载的,而且是具体的类型说明 \$ _tnode 转换来的,是使用类型变量 alpha 来定义的。

由于 Ctree 是重载的,因此它的实例必须由实例产生者来产生。但[和]之间的文本不需解释,这样实现者必须使用特殊符号中来标志唯一产生的名定以便创建 Ctree 的实例。

抽象数据类型的操作符是用说明性数据表示以及实现语言的表达式或语句来实现的。操作符的实现是由形式 Pattern=[m]的说明来确定的,其中 m 是实现语言的代码段,而且 Pattern 将对 m 是表达式还是语句的符号关系特性进行转换。Pattern 可以包含那些决定在 m 中允许的数据表示的说明变量。例如,在 Pascal 的抽象类型 Int 的实现之中,可以假设 x 和 y 是由 x,y,Repint 来说明的变量。那么一个等词操作符可以由下面的说明来实现:

```
eq x y =[x=y]
```

Pattern 左边表示 eq 是由一个表示式实现的。象数据表示一样,右边实现的代码段是作为 x 和 y 上的一个参数化的宏来对待。

一个操作符也可以由一个或多个语句来实现,采用集合的形式,就象将一个值赋给一个变量一样。在这种情况下,关键字 assign 用在了 Pattern 之中。例如,假设 t3 是一个由 t3:Ctree 在 Bintree 的 C 实

局域自适应卡尔曼滤波器的图像复原

军械工程学院三系 贾英江 (河北石家庄 050003)

摘要 本文提出了一种新的图像复原算法,通过局域自适应卡尔曼滤波将被附加白噪声劣化的二维图像复原。首先根据图像局域统计特性用区域分裂与合并算法将图像分割为一系列不连续的局域簇,然后对边界像素进行邻域均匀随机填充计算,之后进行独立的交向矩形窗扫描,并对扫描信号进行自回归建模及滤波,分割后的处理过程可并行进行并考虑了人的视觉特性。

关键词 图像分割 邻域均匀随机填充 独立交向矩形窗扫描 自回建模 局域自适应卡尔曼滤波 图像复原

一、引言

近年来卡尔曼滤波器的图像复原研究已经取得

了很大的进展,但仍有弊端,归为以下几点:

a. 图像整体稳定性假设一般不成立,其第一、二

现中说明的变量,那么,node的实现可以由下面的程序段来说明。

```
assign(t3,node t1 nv t2)=[
    t3=(Struct $ tnode * )malloc(sizeof(struct
    $ tnode));
    t3->lsucc=t1;
    t3->rsucc=t2;
    assign(t3->val,nv,alpha)
]
```

该代码段的最后语句是nv到t3->val的一个抽象赋值。这种赋值用给不知道类型的变量赋值的方式提供了实现方式。这样表示赋值类型的类型变量可作为第三个参数而给出。当alpha被约束到一个实例的类型上时,这个实现就会由执行它的代码段来代替。该代码段是由实现者确定的赋值实现产生的。

处理实现说明涉及到为每个操作符的实现推导出一个类型、报告任何类型出错的pattern、并且确保如果右边有一个类型变量,那么必是Pattern左边某个变量的类型参数。

2.3 用模板产生实例 可以用与程序相关的产生实例的指令来用模板产生一个实例,该指令包含了模板的名字及多个变量,它们的说明建立了该实例的上下文。变量是由类型表达式说明的,而这些表达式又为抽象类型产生了所需的实例。产生实例的指令的例子如图3和图4,都是以Sort为模板的。注意没有确定x和y说明的抽象类型Int和String的实现。这是因为它们是用标准表示的基本类型,而标准表示是由省缺值所确定的。如果指令中的变量类型可以从模板中推导出来,那么这些变量就是需要

作说明。

三、软件模板系统

已经开发了几种工具来支持模板范例中的程序设计。它们包括产生模板实例的实例生成器以及处理实现说明的工具。

模板类型检查器已经用模板重构出来了。在这种情况下,也许不用模板而直接对类型检查器进行编码所花努力更小些,但是如果将可用于一般目的的数据类型的实现(如序列、树、符号表等)库建立起来,那么这个库就可以用在许多应用之中。

四、结论

使用软件模板的方法,将程序设计过程分成两个部分:算法和说明的实现。一个模板由程序员选择合适的抽象数据类型的实现就能产生所需的应用程序。这种方法的明显优势是有效、具有良好的文档。那些Knuth为查询所描述的算法,都可以编码来作为数据类型的实现,而且几乎很少修改即可重用。

参考文献

- [1]Henderson, P. Functional Programming Application and Implementation. Prentice-Hall, 1980.
- [2]Knuth, D. E. Sorting and Searching, volume 3. Addison-Wesley, 1973.
- [3]Thomas, M. Implementing algebraically specified abstract data types in an imperative programming language. In Leatare Notes in Computer Science, vol. 250, pp197-211, Springer Verlag, 1987.
- [4]Volpano, D. M. Software templates. ph. D. Thesis, Department of Computer Science and Engineering, Oregon Graduate Center, Beaverton, Oregon, 1986.
- [5]Volpano, D. M. STS Software Templates System. MCC technical report STP-257-87, 1987.

阶统计量通常均有突变。因此对整幅图像施以同样处理效果并不理想,所以滤波器需适应图像局部特性。

b. 图像状态空间建模通常以其自相关函数为指数型且可分离为前提,而此并非恒为真。为此必须提出新的图像模型。

二、算法

2.1 噪声图像分割 通常图像各部分的空间特性会有很大差别,因此整幅图像甚至其扫描条带并不平稳,同时卡尔曼滤波器对灰度突变不敏感,故平稳假设常使卡尔曼滤波器处理质量下降,如边界模糊、对比度下降等。这里我们将图像视为一系列平稳过程的组合,为此首先提出根据局域均值和局域屏蔽值的图像分割方法。

因噪声均值为零,故可从噪声图像求得原始图像的局域均值:

$$F_{m\alpha}(i,j) = \frac{1}{9} \sum_{K=-1}^1 \sum_{r=-1}^1 g(i+K, j+r) \quad (1)$$

其中:

$F_{m\alpha}(i,j)$ 为像素 (i,j) 的局域均值,

$g(i,j)$ 为噪声图像像素 (i,j) 灰度。

同簇区域局域均值近似相等,但其自相关函数却不同,原因是局域间空间特性的差异。因此为实现图像的合理分割还需考虑其二阶统计量,二阶统计量表征图像空间特性。为此引入屏蔽函数作为空间特性的测度:

$$F_{m\alpha}(i,j) = \sum_{u=i-w}^{i+w} \sum_{v=j-h}^{j+h} D \| (i,j) - (u,v) \| \{ |S_{uv}^x| + |S_{uv}^y| \} \quad (2)$$

其中: $\| (i,j) - (u,v) \|$ 为像素 (i,j) 与 (u,v) 的欧氏空间距离, S_{uv}^x 、 S_{uv}^y 分别为 (u,v) 处水平、垂直方向的灰度梯度,

D 为控制指数衰减的常数,

w, h 为控制点 (i,j) 邻域大小的常数。

若再考虑对角方向则(2)式变为:

$$F_{m\alpha}(i,j) = \sum_{u=i-s}^{i+s} \sum_{v=j-s}^{j+s} e^{-\| (i,j) - (u,v) \|} \left\{ \frac{1}{8} \sum_{m=0}^7 |S_{uv}^{m\pi/4}| \right\} \quad (3)$$

其中: $S_{uv}^{m\pi/4}$ ($m=0,1,\dots,7$) 为八个方向 ($m\pi/4$) 的灰度梯度,

$S = \text{Max}[K, W/z]$ 控制点 (u,v) 邻域大小, k 为扫描图像信号建模的自回归级数, w 为扫描带条宽度,这两个参数后述。

局域屏蔽值由噪声图像直接求得,故它对噪声敏感,表征了图像空间特性,因此可以用它实现局域分裂。局域均值对噪声不敏感(噪声均值为零),故它是原始图像局域均值的准确估计值。同时局域均值高度聚类,同簇区域的局域均值近似相等,因此可给

定局域簇局域均值中心间距阈值,用局域均值搜寻局域簇(局域合并)。这样经过局域屏蔽值的局域分裂后再经局域均值的簇寻处理可实现图像的较为理想的分割。上述分裂与合并算法实为根据图像局域特性将其分簇,各局域簇在空间上聚类为连续区域。图像经过分割后会产生许多小孤点严重影响视觉效果。这些小孤点极可能为分割过程中噪声所致,去除这些小孤点可采用以灰度梯度倒数为权值的邻域平均。但需注意邻域大小以及达到理想的平均效果并保持细节信息。

经过分割以后,各簇局域均值及局域屏蔽值在其簇中心聚类,而自相关函数取决于局域屏蔽值,所以各簇可近似视为广义平稳过程。分割算法主要考虑图像第一、二阶统计量,未计高阶统计量,因高阶统计量的差别人眼不能察觉。

2.2 邻域均匀随机填充 噪声图像分割后局域簇间形成边界,为此需将边界像素通过其邻域计算即进行邻域均匀随机填充操作,以便进行独立的交向矩形窗图像扫描。

如图1所示,设 $G(i,j)$ 为 $-V_c \times H_c =$ 维离

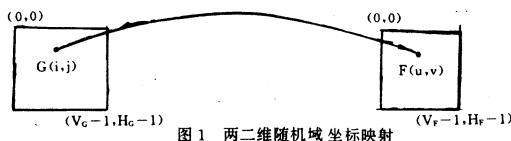


图1 两二维随机域坐标映射

散平稳随机域,其均值为零。经坐标映射得到 $F(u,v)$, $F(u,v)$ 为 $-V_f \times H_f =$ 维随机域。若映射为均匀随机映射则:

$$\text{Prob}[F(u,v) = G(i,j)] = \frac{1}{V_c \times H_c} \quad (4)$$

若 $V_c \times H_c > V_f \times H_f$, 则 $F(u,v)$ 可视为 $G(i,j)$ 样点集 $G(i,j)$ 的重新排列;若 $V_c \times H_c < V_f \times H_f$, 则可将 $F(u,v)$ 分为一些子集 $F_i(u,v)$, 各子集可视为 $G(i,j)$ 样点集的重新排列。

由于 $G(i,j)$ 均值为零且各点独立地进行均匀随机映射,故:

$$E[F(u_1, v_1) F(u_2, v_2)] = E[F(u_1, v_1)] E[F(u_2, v_2)] = 0$$

$$E[F^2(u,v)] = E[G^2(i,j)] \quad (5)$$

因此仅当 $G(i,j)$ 均值为零,则不论 $G(i,j)$ 是否为白, $F(u,v)$ 均近似为白。

图像分割后局域簇间存在边界,边界上的像素需由同簇区域的邻域操作来计算。即根据前述的均匀随机映射原理进行邻域的均匀随机填充,它也是

独立交向矩形窗处理的基础。每簇区域的边界上的窗口内既有本簇像素又有非本簇像素。前者构成源域,后者构成空白域。在窗口内的边界两侧从源域向空白域进行均匀随机映射。这样填充区域大致保持了源域的均值与方差,这一点对由分割的噪声图像及填充区域确定图像模型及其求解至关重要。

2.3 独立交向矩形窗处理 图像分割及填充处理以后,需进行交向矩形窗扫描以为后续的扫描信号建模及滤波作准备。

如图2、3所示,图像用 $M \times M$ 矩形窗进行交替换向扫描,窗内以宽度为 W 的条带扫描。

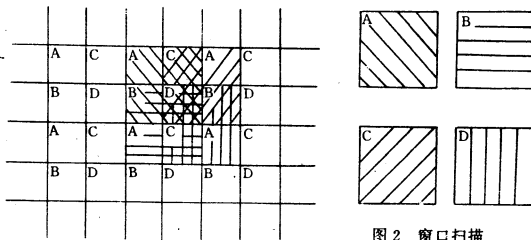


图2 窗口扫描

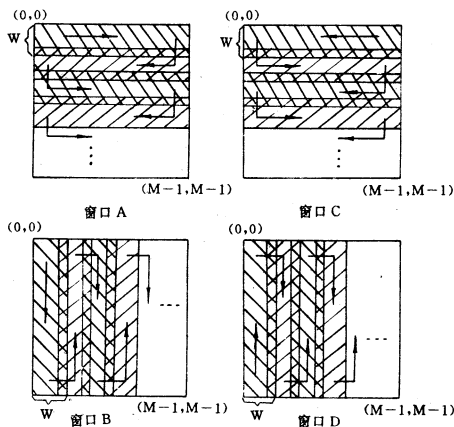


图3 交向矩形窗及其内条带扫描

与通常的光栅扫描不同,每一条带与下一个扫描方向相反的条带连接起来,条带以其间中部像素交叠。这种扫描方式保持了数据的连续性并使像素估计准确性得到提高。窗口按图2所示的方式进行交叠,即窗口在垂直或水平方向一次移动其一半宽度的距离同时改变扫描方向(旋转 90°)。每个像素被四个窗口覆盖,其值分别从四个不同方向(0° 90° 180° 270°)扫描计算,这就是所谓的“交向”。

由于分割过程引入了阈值,故各局域簇每部分的局域均值及局域屏蔽值仍有不同。用矩形窗扫描图像,则平稳要求仅限于窗口内,过程范围缩小,平稳性便更有保证,此乃局域适应之基础。同时窗口均

独立地建模、填充,这样使滤波器具有高度并行结构。

2.4 图像建模 卡尔曼滤波器的核心为其状态空间模型。下面给出由条带扫描得到的 W 维向量信号的自回归模型,并为噪声图像建模。

噪声图像经矩形窗内条带扫描可得一 W 维向量信号 $y(n)$,设 $z(n)$ 为对应于原始图像的 W 维向量信号,并为稳定过程,视其为一 K 次自回归处理的样本:

$$\begin{aligned} z(n) &= \sum_{m=1}^k A_m z(n-m) + u(n) \\ y(n) &= z(n) + v(n) \end{aligned} \quad (6)$$

其中:

$A_i (i=1, 2, \dots, k)$ 为 $W \times W$ 常数矩阵,内含自回归参数,

$u(n)$ 为 $W \times 1$ 的白噪声向量,用于自回归处理,又称系统动态噪声,

$v(n)$ 为 $W \times 1$ 劣化图像的白噪声向量,又称观测噪声,

为使用卡尔曼滤波器公式,需将(6)式改为状态空间形式,令:

$$x(n) = [z(n) \ z(n-1) \ \dots \ z(n-k+1)]'$$

则(6)式变为:

$$\begin{aligned} x(n) &= Fx(n-1) + Gu(n) \\ y(n) &= Hx(n) + V(n) \end{aligned} \quad (7)$$

其中:

$x(n)$ 为 $KW \times 1$ 向量,称之为系统状态向量,

$y(n)$ 为 W 维观测向量,

$G = [I \ 0 \ \dots \ 0]'$ 称为控制矩阵,

$$F = \begin{bmatrix} A_1 & A_2 & \dots & A_k \\ I & 0 & \dots & 0 \\ 0 & I & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & I \end{bmatrix} \quad \text{称为状态转移矩阵,}$$

$H = [I \ 0 \ \dots \ 0]$ 称为观测矩阵,

I 为 $W \times W$ 单位矩阵。

因 $u(n)$ 平稳,故(7)式变为:

$$\begin{aligned} x(n+1) &= Fx(n) + Gu(n) \\ y(n) &= Hx(n) + v(n) \end{aligned} \quad (8)$$

F 由噪声图像确定,可通过下列方程组求得:

$$\begin{aligned} Y &= XT \\ Y &= [R(1) \ R(2) \ \dots \ R(K)] \\ X &= [A_1 \ A_2 \ \dots \ A_k] \\ R(t) &= E[z(n) \ z'(n-t)] \\ T &= \begin{bmatrix} R(0) & R(1) & R(2) & \dots & R(k-1) \\ R'(1) & R(0) & R(1) & \dots & R(k-2) \\ \dots & \dots & \dots & \dots & \dots \\ R'(k-1) & R'(k-2) & R'(k-3) & \dots & R(0) \end{bmatrix} \end{aligned} \quad (9)$$

MHS 中 P7 协议的设计与实现

北京邮电学院 191 信箱 邱大钧(100088)

摘要 CCITT 制定的 MHS 报文处理系统 X.400 系列建议 88 年版新增了 MS 信报存贮部件,扩大了 MHS 的应用范围,促进了微机在 MHS 中的应用。本文着重论述了在微机上实现远程用户代理的关键——P7 协议的主要内容、设计与实现。对实现中遇到的一些技术问题及其解决办法也进行了讨论。

关键词 MHS RUA P7 协议

现代的许多高科技领域都是由若干新技术相互

渗透、交叉、影响形成的。现代通信技术与计算机技

(9)为分块陶布利兹矩阵方程,其解有完善的递归算法。简要说明如下:首先由噪声图像信号 $y(n)$ 求得 $R(t)$ ($t=0,1,2,\dots,k$),由于平稳过程的遍历性, $R(t)$ 可根据时间平均值计算。这样就确定了 Y 和 T ,便可求得 X ,进而确定 F 。

2.5 局域自适应卡尔曼滤波器公式 从噪声图像计算出全部参数后,可得卡尔曼滤波器公式:

$$P_{(n+1|n)} = P_{(n|n)}F' + GQG'$$

$$K_{(n+1)} = P_{(n+1|n)}H' [HP_{(n+1|n)}H' + R]^{-1}$$

$$P_{(n+1|n+1)} = [I - K_{(n+1)}H]P_{(n+1|n)}$$

$$\hat{X}_{(n+1|n)} = F \hat{X}_{(n|n)} + K_{(n+1)}[y(n+1) - HF \hat{X}_{(n|n)}] \quad (10)$$

初始条件为 $x(0|0)=0, P(0|0)=p$

其中: $P_{(n+1|n)}$ 为一步预测均方误差阵,

$P_{(n+1|n+1)}$ 为滤波均方误差阵,

$K_{(n+1)}$ 为滤波增益矩阵,

R 为观测噪声方差阵

Q 为动态噪声方差阵。

求解(10)的前三式得状态滤波公式即(10)中的第四式,用它对 W 维向量信号 $z(n)$ 滤波,像素按下式计算:

$$\hat{X}_{(n+1)} = H \hat{X}_{(n+1|n+1)}$$

P, Q, K, F 均取决于局域簇及矩形窗内图特性,因此滤波器适应每一个矩形窗的局域特性,同时各局域簇可并行处理,簇内窗口可并行建模及滤波。

三、算法框图

局域自适应卡尔曼滤波器算法可归结如图4所示。

参考文献

[1]Aboutalib, A. O., Murphy, M. S., Silverman, L. M., "Digital Restoration of Images Degraded by General Motion Blur", IEEE Transactions, Automatic Control, vol. AC-22, No. 3, June, 1977.

[2]Suresh, B. R., Shenoi, B. A., "New Results in Two-Dimensional Kalman Filtering with Application to Image Restoration", IEEE Trans., Circuits, Systems, vol. CAS-28, No. 4, April, 1981.

[3]Tou, J. T., Gonzalez, R. C., Pattern Recognition Principles, Addison-Wesley, Reading, MA, 1974.

[4]Netravali, A. N., Prasada, B., "Adaptive Quantization of Picture Signals Based on Spatial Masking", Proc. IEEE, vol. 65, No. 4, April, 1977.

[5]Chen, P. C., Pavlidis, T., "Segmentation of Texture Using a Co-Occurrence Matrix and a Split-and-Merge Algorithm", Computer Graphic and Image Processing, vol. 10, 1979.

[6]S. Yokoi et al., "An Analysis of Topological feature at digitized binary pictures using local features", CGIP, vol. 4, pp63-73, March 1975.

[7]许殿元、丁树柏,《遥感图像信息处理》,宇航出版社 1990.

[8]阮秋琦,《数字图像处理基础》,中国铁道出版社, 1988.

[9]上海师范大学数学系概率统计教研组,《回归分析及其试验设计》,上海教育出版社,1978.

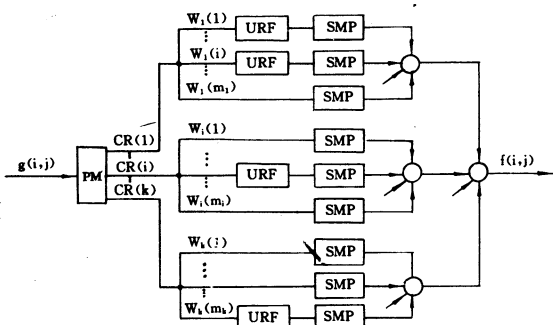


图4 局域自适应卡尔曼滤波器框图

术的发展与结合而形成的以计算机网为基础的数据通信技术,对于沟通丰富的国际、国内信息资源,克服语言及时空障碍,起着越来越重要的作用。MHS报文处理系统就是一种新型的建立在计算机网上的、以计算机间数据通信为基础的信息处理系统。它是传统邮政系统的电子模拟,融合了计算机技术、通信技术、信息处理技术,从而发展为集邮件、电报、传真、语音等业务于一身的新业务,极大推动了现代通信与办公自动化的发展。

一、基于 88 年 X.400 系列建议的 MHS 系统

CCITT 制定 MHS 标准,目的是为了提供国际上共同遵守的准则,使得不同国家的各个专用网、公用网,通过国际分组网能够互连,从而实现世界范围的信报交换网。此系列标准是在模拟传统邮政系统的基础上,吸取了电报、传真及计算机信息处理技术等精华,利用存贮转发的通信机制,在 OSI/RM 中实现的一个具有分布式结构的新一代电子邮件系统。MHS 位于 OSI/RM 应用层,具有如下特点:

- 1) 以 OSI/RM 为基础,符合国际标准
- 2) 能处理多种信息类型,如文本、语音、图像、传真等,
- 3) 利用存贮转发通信机制,克服了时空局限,
- 4) 具有更高的性能价格比,
- 5) 更好的安全机制。

88 年 MHS 功能模型具有 4 类功能部件: MTA、UA、AU、MS。其中,MS 是 88 年标准新增的部件。UA 可分二类:与 MTA 位于同一主机的称同驻 UA;与 MTA 位于不同主机的称独立 UA。MS 是 MTA 与独立 UA 间的中间媒介,只与一个 UA 对应,它可对 MTS 投递给用户及用户提交给 MTS 的信报进行存贮管理;用户通过 MS,可得到检索服务、自动警示服务等业务。MS 的引入,是 MHS 发展的一个突破,使独立 UA 在微机上实现成为可能。

88 年版 MHS 系统的总体协议结构含 4 类协议:P1、P2、P3、P7。其相互关系如图 1 示。

二、在微机上实现具有智能的远程独立 UA

随着微电子技术的迅速发展,微机性能价格比不断提高,已在各行业广泛应用,并走入家庭。如何提高微机的使用效率成为当务之急。在微机上实现具有智能的远程独立 UA,通过电话网将这些微机联网,在目前情况下,可谓经济实惠,可大大提高微机利用率,促进信息交换与数据资源共享。X.400 系列建议为实现这样的系统提供了国际标准。

经过对 X.400 系列建议的分析,决定采用如下

的系统模型(图 2):

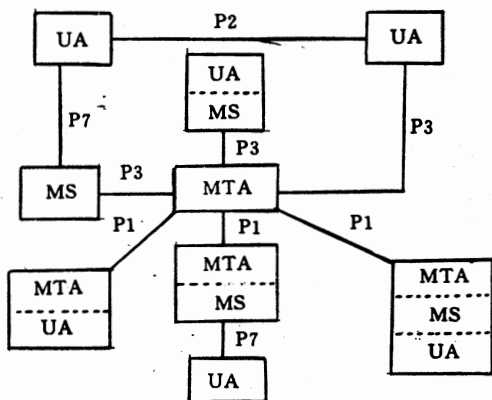


图 1 协议及功能模型图

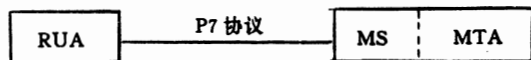


图 2 RUA 与 MS 系统连接模型

由此系统模型可知:1) MS 与 MTA 位于同一主机上,由于主机是连续运转的,因此,无论何时,UA 提交信报给 MS 或 MTA 投递信报给 MS,都不致操作失败;2) 主机为容错计算机,大大降低了丢失 MS 中信报的可能性,且故障出现的概率极小;3) MS 分担了 UA 的大部分存贮任务,并替代 UA 与 MTA 接口,RUA 仅保留少量有价值的已阅信报,且与 MS 接口简单,这使 RUA 实现的软、硬件要求降低,使其在微机上实现成为可能。

由图 2 知 RUA 与 MS 的接口通过 P7 协议实现,P7 协议是实现 RUA 的关键。P7 协议的模型如图 3 所示。

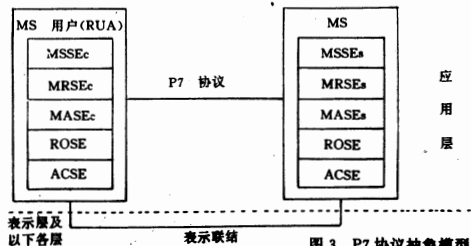


图 3 P7 协议抽象模型

根据 OSI/RM,应用层的对等应用进程用对等应用实体表示。每个应用实体包含一个用户元素和一组应用服务元素。应用进程间的信息交换由应用实体、应用协议和表示层服务完成。图 3 中,用户元

素分别为 MS 和 RUA, 应用服务元素为 MSSE、MRSE、MASE、ROSE、ACSE。其中, MSSE、MRSE、MASE 是 MHS 系统中定义的应用服务元素, 而 ACSE 和 ROSE 是 OSI/RM 应用层的公共应用服务元素。

对 MS 抽象服务的访问由 MSSE、MRSE、MASE 支持。ACSE 和 ROSE 完成应用实体间的联接建立与释放和远程安全通信服务。

P7 协议提供的服务如下:

1. MS 与 RUA 间应用联接的建立与释放

- MS-bind,
- MS-unbind,

2. 远程安全通信服务

a. 信报间接提交服务元素 MSSE

- 普通信报提交,
- 探询信报提交
- 取消延时投递
- 提交控制

b. 信报检索服务元素 MRSE

- 汇总,
- 列表,
- 提取,
- 删除,
- 向 MS 登记,
- 自动警示,

c. 信报管理服务元素 MASE

- MTS 登记,
- 改变证实。

三、P7 协议的设计

在 P7 协议的设计中, 采用如下设计规则:

1、依据 OSI/RM, 建立对应层次结构, 适应未来的发展;

2、用软件工程思想作指导, 采用结构化的程序设计方法;

3、设计目标是易于使用, 费用低廉;

4、实现国际标准定义服务的全集。

根据以上设计原则, 设计 P7 协议的层次结构为:

1、低三层, 可有以下几种选择:

- 使用电话网、电报网;
- 使用分组交换网、局域网;
- ISDN;

鉴于我国现在的分组交换网、局域网, ISDN 还不普及, 设备及传输费用较高; 另一方面, 电话网分布较广, 使用费用低。若使微机通过 RS232 和 MODEM 接入电话网, 那不仅通信设备便宜, 通信费用也低。故目前采用此方案。今后的发展, 可用 X. 25 卡接入分组网。

2、会话层及传输层: 由于 RUA 直接接入电话网, RUA 与 MS 是端到端的关系, 此二层的功能与

下层的重复, 故可不考虑。

3、表示层: 由于应用层 APDU 数据结构复杂, 为了防止数据结构定义的二义性, 使通信双方能互相识别对方发来的 APDU, 故采用 CCITT 的 X. 208 及 X. 209 定义的 ASN. 1 表示法及传送语法。

4、应用层: 完成上述服务元素定义的服务及相关功能。

四、P7 协议的实现

根据上节讨论, 可得图 4 的软件总体模块图。

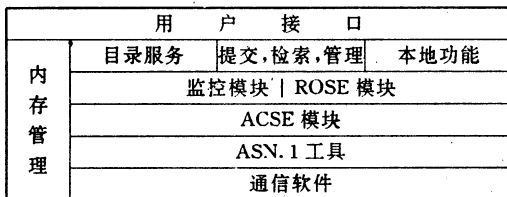


图 4. 软件总体模块图

用户接口是一集成窗口环境, 通过使用下拉菜单选项, 用户可完成 X. 400 系统的远程操作(提交、检索、管理、目录服务)和本地功能(本地文件夹管理, 信报编辑等)。

内存管理模块对机器内存进行综合管理, 有效、合理地进行内存分配与释放, 保证系统正常运行, 避免了各模块“各自为政”时可能发生的内存紧张与冲突。

目录服务提供 CCITT X. 500 系列建议定义的服务。目录服务使 MHS 有以下功能: 用户友好命名、分发表功能、证实能力(安全服务)、目录查询能力。

提交、检索、管理模块完成 MSSE、MRSE、MASE 应用服务元素的功能。根据用户要求, 对应的应用服务元素形成相应 APDU, 交 ROSE 处理。

本地功能涉及信报管理、信报编辑等。

监控模块处理自动警示, 并辅助完成接收操作。由于系统处于 DOS 单任务操作系统, 为实现前后台任务的切换, 完成给定的任务, 对 DOS 的一些中断例程进行了修改。

ASN. 1 (Abstract Syntax Notation One) 编解码工具用来对应用层的 APDU 进行编解码, 形成的二进制数据流交给通信软件处理, 此模块可根据用户的要求采用加载方式或中断方式工作。

通信软件实现机间具有流控、纠错功能的异步双工通信。它采用硬中断收、软中断发的方式, 对由编解码模块收到的数据块进行分帧处理, 发送与接收采用动态内存管理。

本系统编程用 C 语言和汇编语言混和编程, 这—是宜于移植, 二是在保证编程速度的前提下, 使代码效率高。

本系统实现环境基于 XT 机, 采用 DOS3. 3 操作系统。这是由于 XT 机性能较强, 价格便宜, 而 DOS 系统在国内使用范围很广, 这样使本系统易于

CGA 显示器低分辨率屏幕图形设置技术

吉林农垦特产专科学校 姚运生 (132109)

摘要 本文介绍了 IBM PC AT 机 CGA 显示器低分辨率图形的设置方法, 程序用 Turbo C 语言编写而成, 能够较好地绘出图形, 并显示 16 种颜色的汉字。文中给出了相应的函数, 读者可方便地加载到自己的程序中, 它拓宽了 CGA 的图形显示功能, 相信它一定会使绝大多数 CGA 用户大开眼界。

关键词 低分辨率 视频方式 CGA 显示器 写像素 字模数据

IBM PC 彩色图形适配器 (CGA) 图形方式下具有高、中、低三种分辨率方式, 高分辨率 (640×200) 具有 16 个调色板, 每个调色板只能显示黑白两色; 中分辨率 (320×200) 有 2 个调色板, 每个调色板可显示 4 种颜色, 其屏幕及调色板设置可用 Turbo C 如下语句实现:

```
palette(pnum) /* 调色板 */
mode(mode __code) /* 屏幕方式 */
int pnum;      int mode __code;
{
    {
        __BH=1;      __AL=mode __code;
        __BL=pnum;    __AH=0;
        __AH=11      geninterrupt(0x10);
        gepinterrupt(0x10);
    }
}
```

低分辨率 (160×100) 调色板确定边框颜色 (共 16 色), 可显示 16 种颜色图形, 调色板设置同上。目前所有的 IBM BIOS 只设置了中、高分辨率模式, 而没有低分辨率设置, 而且很少有资料介绍低分辨率

推广使用。

在实现过程中, 首先要解决的是内存问题, 由于 ASN.1 工具软件及菜单界面软件等规模较大, 640K 内存十分紧张。故而在实现中采用覆盖技术, 动态使用内存。

另一问题是在单任务操作系统中实现多任务。这要对操作系统本身作修改, 适应前后台操作。实现中, 要特别注意避免代码重入问题。

速度也是用户关心的问题。为提高速度, 又要考虑有限的内存, 故只在内存中建立了少量的常用数据索引链, 另有一些索引以文件形式放在磁盘上。这样内外存协调工作, 有效地提高了速度。

五、结束语

本文从实际出发, 论述了远程用户代理中 P7

的设置方法, 因此, CGA 用户很难获得低分辨率的设计途径。笔者经过大量的摸索, 终于找出了低分辨率方式设置的参数, 比较满意地设计出了低分辨率图形方式, 能够较好地绘出图形、显示汉字, 该程序用 Turbo C 语言编成, 在 IBM PC/AT 机上实现, 显示器 CGA, 汉字系统 2.13H。现将基本方法作一简要介绍:

一、设置低分辨率方式 IBM BIOS 设置视频方式, 可通过 INT 10H 简单地实现, 但 BIOS 中没有支持低分辨率图形的部分, 因此必须对视频控制器编程, 函数 GRAINIT() 中将 12 个 6845 数据寄存器装载适当的参数建立图形方式, 它们通过一个简单的端口地址存取, 先将地址 3D4H 装入一个索引值, 然后在地址 3D5H 装入数据寄存器端口, 装载完毕后用函数 CLS() 给图形页清零, 把从 B800H 开始的 16KB 字节图形存储器都置 0X00DE, 在端口 3D8H 上写入 0X09 启动视频显示进入图形方式。

```
static unsigned char GraphTable[] =
```

协议的设计与实现方法, 对远程用户代理的总体结构也做了介绍。由此实现的系统完成了 MHS 定义的用户代理的功能, 且设有接口, 易于与其它系统互连。使用远程用户代理系统可最大限度地发挥 PC 机的功能。

参考文献

1. CCITT, Blue Book, Recommendation X. 400 Series, 1988.
2. CCITT, Blue Book, Recommendation X. 200 Series, 1988.
3. Maling and Shafei, "The First Step to 1988 X. 400 Systems", ICC'91, Beijing.
4. C. F. Wilkinson, "X. 400 electronic mail", Electronics & Commun., Vol 3. No. 6, 1991.

```

0x71,0x50,0x59,0x0a
0x7c,0x06,0x64,0x70
0x02,0x01,0x20,0x00
0x00,0x00,0x38,0x28
);
int GraInit()
{
    register int i;
    outportb(0x03d9,0x02);
    outportb(0x03d8,0x01);
    for(i=0;i<128;i++){
        outportb(0x03d4,i);
        outportb(0x03d5,GraphTable[i]);
    }
    CLS();
    outportb(0x03d8,0x09);
    outportb(0x03d9,0x00);
    return;
}
CLS()
{
    int i;
    unsigned char far * ptr=(char far *) 0xb8000000;
    for (i=0;i<16000;i+=2){
        *(ptr+i)=0xde;
        *(ptr+i+1)=0x00;
    }
}

```

二、写像素 低分辨率方式下,CGA 的显示 RAM 地址每个字节表示 2 个像素,即四个二进制位表示一个点,其值在 0—16 之间,因此有 16 种颜色。像素按照在屏幕上自左向右顺序排列,第 0 像素占 4—7 位,第 1 像素占 1—3 位,所有像素存于 B800H 以下,每一像素行占 80 字节,每字节两点,这样一行 160 点,一屏共 100 行。屏幕点位置通过内存地址来确定,这里采用了直接向显示 RAM 送信息的方法,在写点时,为保证不影响其它点,则要用位屏蔽的方法,下面是低分辨率画点函数:

```

void point(x,y,color __code)
int x,y,color __code;
{
    int index,t,bit __position;
    char xor;
    unsigned int far * ptr=(int far *) 0xb8000000;
    if(x<0||x>100||y<0||y>160) return;
    xor=color __code& 128;
    color __code=color __code& 127;
    bit __position=y%2;
    color __code<<=4*(1-bit __position)+8;
    index=x*80+(y>>1);
    t=(ptr+index)&0xffde;
    if(!xor) *(ptr+index)=t|color __code;
    else *(ptr+index)=t^color __code;
}

```

利用该函数就可以在低分辨率方式下绘出各种

颜色的图形,其具体作法可参阅有关 C 语言图形绘制的方法,如直线、矩形、圆等,此处从略。

三、显示汉字 汉字显示字库是 16 点阵的,每个汉字的字模数据为 16×16 点阵,计 256 个二进制,共 32 个字节。由于在高分辨方式下,屏幕显示的 RAM 地址每个字节的一个二进制位就表示一个点,因此,高分辨方式下汉字显示只要将字模数据直接写到屏幕上相应位置即可。而低分辨率则不然,它每个字节表示 2 个像素,每个像素四个二进制位,因此,汉字输出要经过特殊加工处理。首先从汉字库中读取给定汉字的字模数据,有关汉字的字模数据许多期刊上早有介绍,这里为了说明问题,也将该函数 Get16(Ch,dot)列出。

```

get16(ch,dot)
char ch[];
int dot[];
{
    FILE * fp,* fopen();
    int i,result;
    int c1,c2,c;
    long l;
    if((fp=fopen("hzk16","rb"))==NULL){
        printf("open clib file ERROR! \n");
        exit(0);
    }
    c1=ch[0];
    c2=ch[1];
    c1=c1+96;
    c2=c2+96;
    c=c1*100+c2;
    l=c/100;
    l=c-190-6*(1-16);
    l=32*(1-l)
    result=fseek(fp,l,SEEK_SET);
    for(i=0;i<32;i++){
        fscanf(fp,"%c",&dot[i]);
        dot[i]=dot[i]&0x00ff;
    }
    fclose(fp);
}

```

读取汉字字模数据之后将其送到屏幕。这里需要对汉字的点阵及显示颜色进行加工处理,要进行一系列的字节位的操作。将原字模数据的每一字节变成四个,每个二进制位变成四位,即原来的每两个二进制位变成一个字节,所以这样输出的汉字在屏幕上原字宽度的四倍,字体颜色的确定,只要将颜色数值写到每个点去就可以了。该过程通过函数 VIEW()完成,函数中参数 X、Y 用于确定字体在屏幕上的位置,分别表示字型的左上角所在的行和列。color 表示字体颜色,其数值在 0—16 之间。

VIEW(int ch[],int x,int y,int color)

计算机绘制和打印电路图的新方法

江汉石油学院电子工程系 金波 (湖北江陵 434102)

摘要 本文介绍一种只要用普通 PC 机和 24 针打印机就能绘制和打印出高质量电路图的
新方法。根据这一方法研制出的汉字系统可以非常方便地打印教材、试卷、论文等,对计算机
辅助教学也有重要意义。

目前,微机绘图,特别是交互式绘图系统得到广泛的应用。在电气、电子技术领域,人们接触最多的是各种各样的电路图。在高等院校,电类课程的教材、试卷、电方面的论文以及计算机辅助教学领域都涉及到电路图。因此,电路图的制作、显示、修改以及打印输出等自动化程度的提高就显得特别重要。

虽然用各种高级语言编制交互式绘图系统可以达到绘制电路图的目的,但本文介绍的方法具有更多的优越性和实用性。

一、电路图形绘制和打印的一般方法

电路图形的绘制一般采用两种办法:

一种是采用专门的绘图系统绘制图形。如 Auto CAD 绘图系统、Tango 绘图系统等,另一种是用各种高级语言编制的交互式绘图程序进行键盘绘图。这种程序具有绘制、修改、存图、取图等各种绘图功能。通常多用 BASIC 语言编写。

这两种方法都有突出的缺点:第一种方法的绘图系统比较大、复杂、不易掌握。而且对硬件配置要求比较高,要配置专用的绘图仪,否则,输出的图形不理想。同时还有一个汉化问题,即在图形中写汉字也比较困难。所以一般不用。通常有得最多的是第二种方法,即采用键盘绘图程序。虽然这种方法简单实用,不必对每幅图编程,大大加快了绘图速度。但也存在两个严重的问题。

(1)图形的存储需要占用较多的存储空间,虽然采用了不少压缩数据的方法,但所占空间仍然可观。

(2)图形的输出采用屏幕拷贝,拷贝出的图形质量差。此外,图形的拷贝位置也无法控制。虽然可以采用分辨率高的显示器来提高图形质量,这要增加投资。同时,随着显示器分辨率的提高,图形的存储空间也随之增大。

二、字模点阵拼接绘图法

```
{
int index,i,j,k;
unsigned char bit,bit1,bit2,bit3;
unsigned char far * ptr=(char far *)
0xb8000000;
color=color&15;
for(j=0;j<16;j++)
for(i=0;i<2;i++)
{
index=i*8+j*160+x*160+y*2+1;
bit=ch[i+j*2];
for(k=0;k<7;k+=2){
bit1=bit<<k;
bit1>>=7;
bit1*=color;
bit2=bit<<k+1;
bit2>>=7;
bit2*=color;
```

```
bit3=(bit1<<4)|bit2;
*(ptr+index+k)=bit3;
}
}
```

CGA 低分辨率方式打破了中分辨方式下只有 4 种颜色的限制,它虽然远不如目前被人们广泛使用的 EGA、VGA 等令人满意,但对于一个已具有 CGA 图形适配器的计算机用户来说,获得低分辨的设置方法定会别开生面的。然而必须向读者声明,此篇纯属本人实践所得,并因本人水平有限,设置方法很可能存在不足之处,欢迎广大读者指点。

参考文献

[1][美]Herbert Schild 著、郭兴社等编译,《C 语言大全》,1991 年 6 月第一版、电子工业出版社。

[2]东阳生等,《Turbo C2.0 高级系统程序设计》,1991 年 10 月,北京希望电脑公司、北京市新闻出版局。

[3]王喜喜等,“谈点阵汉字的字模数据”,《微机机与应用》,1992 年 6 期第 40-44 页。

众所周知,汉字是由字模点阵构成的。字模点阵越高,打印出的字越漂亮。本方法就是仿照生成图形汉字的方法,事先用造字程序造好所需要的各种图形符号的字模点阵,绘图时将这些图形字模拼接起来,就可绘成各种各样的电路图、具体方法如下:

第一步:首先确定字模方案。根据图形单元的大小,它在整幅电路图的比例以及加上文字说明后的美观等,选择由若干个字模拼接组成。然后,再在坐标纸上精心绘制出所需图形部分。这就做成了一张字模点阵图。

第二步:使用专用的造字程序,将点阵图输入到汉字库中。汉字库中的空区和一些很少使用的区都可供专用图形符号使用。由于屏幕显示采用的是 16×16 点阵字模,打印时一般采用 24×24 点阵字模,所以,同样的图形要造两次。对造好了的图形要打印出来看一看,如果不满意再用造字程序对其修改,然后再打印。这样反复进行直至对图形满意为止。

第三步:将专用的图形和符号按规律组成图形库。如何将这些图形符号显示到屏幕上?一般采用“区位码”输入法。但这种方法非常麻烦。输入一个图形符号要查一次它所在字库中的区位码。由于有成百上千个图形符号,要记忆是不能的,这样就影响了输入图形的速度。笔者采用的是“选择法”,即将图形符号有规律地显示在屏幕的最下一行(提示行),按“>”或“<”键前后翻页,可显示出各种图形符号。键入图形符号前的数字,就可将该图形符号显示在屏幕上。这样输入图形符号的速度特别快。

第四步:应用各种文字编辑软件,如 wordstar 等。将要输入的文字、电路图键入。输入图形时与输入汉字一样。然后以文本文件形式存盘保留。

第五步:打印输出电路图。打印电路图时,首先要取消打印机的字间距和行间距,然后按一般打印文本文件的方法打印即可。

三、特点

1. 图形的存储和汉字文章一样,一个电路图一般也只需要几百个字节,大大节省了存储空间。

2. 由于图形和汉字一样,可以任意排版。图形的定位也不成问题。由于可用各种文字编辑软件,图形的修改也十分方便。由于采用“选择法”输入专用电路图形或符号,输入图形的速度也特别快,几乎与敲汉字的速度差不多。

3. 图形的大小不受屏幕面积的局限,可以任意加宽加长,只要打印机能打得下。同时图形的大小也可以通过改变字型来放大或缩小。

4. 打印输出的图形质量高。笔者采用优化的图形方案,使打印输出的图形点阵比较密,且美观实

用。见图形举例。

5. 这种方法虽然造点阵图形比较麻烦,但这是一次性的劳动,有了专用图形库,输入图形就简单了。

6. 用这种方法不但可以绘制电路图,还可以打印各种数学符号和数学式子。特别是一些专用符号,如 \bar{U} 、 \bar{i} 、 \bar{u} 、 \bar{i} 、上标、下标等。并且显示与打印的电路

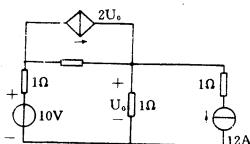


图1 直流电路

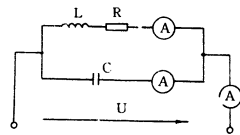


图2 交流电路

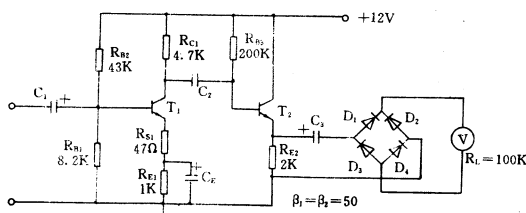


图3 放大器和整流电路

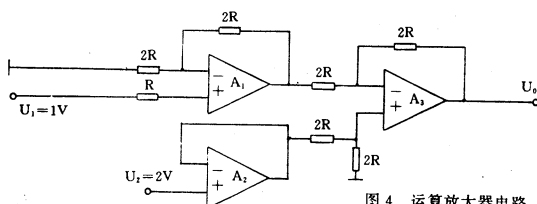


图4 运算放大器电路

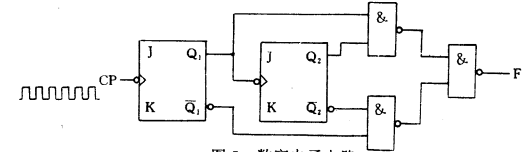


图5 数字电子电路

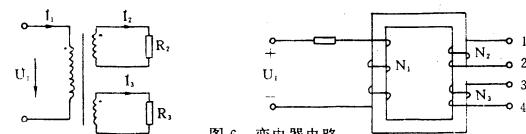


图6 变压器电路

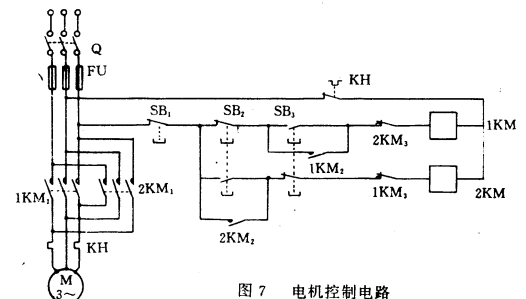


图7 电机控制电路

TVGA256 色图象截取技术

武汉水利电力学院教务处教学研究室 季军杰 (430072)

TVGA 是 Trident 微系统公司推出的一种通用的低成本、全功能、单芯片超级 VGA 图形适配器系统,它与标准 VGA、EGA、CGA、MDA 和 Hercules 卡全兼容。以 TVGA 为基础的适配器也是目前国内中、高档微机最为流行的一种 VGA 适配器。其中 TVGA8900(C)卡以 256 色支持 320×200 、 640×400 、 640×480 、 800×600 、 1024×768 五种图形模式,见表一。绝大多数演示软件都是在 $640 \times 480 \times 256$ 色方式下开发的,其图象制作得非常逼真、精美、细致,如同照片一般,很有保存、再利用的价值。然而,这些图象只能在其自身软件环境支持下才能调出显示,且图象文件格式不统一,很难被其它应用程序所采用。能否找到一种有效办法,把屏幕图象截取下来,以统一格式存放,脱离原环境也能再利用呢?笔者为此进行了一些探索,证明完全可行。下面就以 $640 \times 480 \times 256$ 色——模式 5DH 为例说明其实现原理。程序用 8088 汇编语言编写实现。

在 256 色方式下,VGA 存储图象信息时采用的是压缩象素法,即每个屏幕象素点信息占用显示缓冲区一个字节,且象点坐标与其存放位置两者之间呈线性关系。TVGA 系统采用分页显示机制,每个显示页 64K 字节长,页面段基址均为 A000H。这样,在模式 5DH 下,满屏信息要 $640 \times 480 = 300K$ 字节,需 $[300/64] = 5$ 个显示页,其中前 4 页(页号 0-3)为满页,最后一页(页号 4)只有 44K ($300 - 4 \times 64 = 44$)字节。活动显示页通过页选择寄存器控制存取。每选择一页,就把该页图象信息写入文件。此外,图

象色彩信息采用存取 DAC 寄存器组获得。DAC(数据模拟转换器)把二进制颜色信息转换成模拟电平,用来驱动显示器,DAC 中的颜色查找表用来以调色板中选出 256 色,显示在屏幕上。这些颜色数据放在文件最后。至此,一幅画面的完整信息就全部截取下来了。

上述过程的具体实现见截屏程序(程序 1),通过汇编、链接、转换为 COM 文件运行后,每按一次拷屏打印键,便把当前屏显截取下来。总共可存 $26 \times 26 = 676$ 幅图象。

至于截取图象的重现程序(程序 2),几乎是程序 1 所示过程的逆过程,因而程序流程也是对称的。汇编、链接、转换成 COM 文件运行后便可把所截图象一幅幅重新展现在你面前,令人赏心悦目。

虽然上述程序是按模式 5DH 编写的,但在其他 256 色模式下的操作原理完全一样,有兴趣的读者只需稍加改动两个程序,便可存贮/恢复所有 256 色模式图象,这样就更具通用性。

两程序均附详尽注释,以便读者理解,且已上机通过,使用效果极佳。

表 1 TVGA 256 色显示模式

模式号	分辨率	VRAM 基地址	图象大小
13H	320×200	A0060H	64000 字节
5CH	640×400	A0000H	64K 字节 $\times 3 = 59392$ 字节
5DH	640×480	A0000H	64K 字节 $\times 4 = 45056$ 字节
5EH	800×600	A0000H	64K 字节 $\times 7 = 21248$ 字节
62H	1024×768	A0000H	64K 字节 $\times 12$

图、表达式是一致的,这对用计算机辅助教学打下了基础。

7. 只要一般的 PC 机(能显示汉字)和普通 24 针打印机,而不需要增加任何新设备,就可完成电路图的输入和打印输出。适用于所有低档、高档微机。对显示器也无严格要求,只要能显示出汉字即可。所以这种方法是极其实用的。

8. 本方法并不适用于绘制一般的图形,只适合于绘制一些有规律的图形,如各种电路图就是相当有规律的,还可以绘制其它一些有规律的图形,如程序框图、系统结构图、硬件接线图等。

四. 打印输出的电路图举例

1. 由理想电路元件构成的电路图:

图(1)直流电路,图(2)交流电路。

2. 由各种电子元器件构成的电路图:图(3)放大器和整流电路,图(4)运算放大器电路,(5)数字电子电路。

3. 由变压器、电机构成的电路图:图(6)变压器电路,图(7)电机控制电路。

五. 结束语

如上所述,用本方法绘制和打印电路图是非常实用的。笔者已用这一方法构成了一套汉字系统。并用它完成了“电路分析”、“电工学”两个试题库的研制和几本教学参考书的编印,效果良好。

需要此软件者,请同作者联系。

程序 1

```

code segment
org 100h
assume cs:code,ds:code,es:code
start proc
    mov ax,cs
    mov ds,ax
    mov dx,offset int5
    mov ax,2505h;改写拷屏中断
    int 21h
    mov dx,(p1+15)/16+10h
    mov ax,3100h;驻留退出
    int 21h
start endp
int5 proc far
    jmp stt
dac db 300h dup (?);颜色数据
fn db 'AA',0;文件的 ASCII 码 Z 串
page1 proc;选显示页(AH=页号)
    mov dx,3c4h;定序器索引寄存器端口地址
    mov al,0eh;页选择寄存器索引号
    out dx,al
    inc dx;定序器数据寄存器端口地址
    mov al, ah
    xor al,02;页号位 1 取反
    out dx,al
    ret
page1 endp
stt: sti
    push ax
    push bx
    push cx
    push dx
    push ds
    push bp
    push es
    mov ax,cs
    mov ds,ax
    mov es,ax
    mov ah 0fh;判断显示方式
    int 10h
    cmp al, 5dh;640 * 480 * 256 色方式?
    je n1
    jmp el
n1: lea dx,fn
    xor cx,cx
    mov ah,3ch;创建文件
    int 21h
    push ax;文件句柄
    mov bx,ax
    mov cx,32 * 1024;每次写盘 32K
    push ds
    mov ax,0a000h;视频段基址
    mov ds,ax

```

```

    mov ah,0;
a:  push ax;
    call page1;
    xor dx,dx;
    mov ah,40h;
    int 21h
    mov dx,cx;
    mov ah,40h;
    int 21h;
    pop ax;
    inc ah;
    cmp ah,4;
    je b
    jmp a
b:  call page1
    mov cx,45056;第 4 页图象写盘
    xor dx,dx
    mov ah,40h
    int 21h
    xor bx,bx
    mov cx,256
    mov dx,3dah;
    in al,dx;
    test al,08h;
    jz 12;
11: in al,dx;
    test al,08h
    jne 11;
12: in al,dx;
    test al,08h;
    je 12;
    pop ds
    mov dx,offset dac
    mov ax,1017h;读 DAC 寄存器组
    int 10h
    pop bx
    mov cx,300h
    mov ah,40h;写入颜色数据
    int 21h
    mov ah 3eh;关闭文件
    int 21h
    cmp byte ptr fn, 'Z';
    jz,s1;
    inc byte ptr fn;
    jmp e;
s1: inc byte ptr fn[1];
    mov byte ptr fn, 'A';
e:  mov ax,0e07h;响铃,以示成功
    int 10h
el: mov al,20h
    out 20h,al
    pop es
    pop bp
    pop ds

```

4 整页(0-3)图象写盘

测试输入状态寄存器 1 之位 3
等待下一个垂直回扫期开始

建立下一个文件

```

    pop dx
    pop cx
    pop bx
    pop ax
    iret
int5 endp
p1 equ $-start
code ends
    end start
程序 2
code segment
    org 100h
    assume cs:code,ds:code,es:code
show proc
    jmp stt
dac db 300h dup(?)
fn db 'AA',0
pagel proc
    mov dx,3c4h
    mov al,0eh
    out dx,al
    inc dx
    mov al,ah
    xor al,02
    out dx,al
    ret
pagel endp
stt:mov ax,5dh
    int 10h
    mov ax,cs
    mov ds,ax
    mov es,ax
    lea dx,fn
    xor cx,cx
    mov ax,3d00h;打开文件读
    int 21h
    jnc nl
    jmp el
nl:mov bx,ax;文件句柄
    mov cx,32*1024
    push ds
    mov ax,0a000h
    mov ds,ax
    mov ah,0
a:  push ax
    call pagel
    xor dx,dx
    mov ah,3fh
    int 21h
    mov dx,cx
    mov ah,3fh
    int 21h

```

```

    pop ax
    inc ah
    cmp ah,4
    je b
    jmp a
b:  call pagel
    mov cx,45056
    xor dx,dx
    mov ah,3fh
    int 21h
    pop ds
    mov dx,offset dac
    mov cx,300h
    mov ah,3fh;读出颜色数据
    int 21h
    mov ah,3eh
    int 21h
    xor bx,bx
    mov cx,256
    push dx
    mov dx,3dah
    in al,dx
    test al,08h
    jz 12
11: in al,dx
    test al,08h
    jne 11
12: in al,dx
    test al,08h
    je 12
    pop dx
    mov ax,1012h;设置 DAC 寄存器组
    int 10h
    cmp byte ptr fn,'z'
    jz sl
    inc byte ptr fn
    jmp e
sl: inc byte ptr fn[1]
    mov byte ptr fn,'A'
e:  xor ax,ax;读键盘
    int 16h
    cmp al,lbh;ESC 键?
    je el
    jmp stt
el: mov ax,3;80 列文本方式
    int 10h
    mov ax,4c00h;返回 DOS
    int 21h
show endp
code ends
    end show

```

GW-500 显示器维修两例

解放军军事经济学院计算中心 刘亮生 陈名海 (武汉 430035)

【例一】一台 GW500 高分辨彩色显示器,在显示彩色图形时右上角有一小块色斑。

检查与维修

首先利用机内自动消磁电路进行消磁处理,效果不明显。采用机外强消磁手段,现象依旧,说明故障是显象管本身色纯不良。显示器在出厂前已根据技术要求将色纯度调好,彩显管的三个电子束通过具有选色机能的荫罩后都能准确地轰击其所对应的荧光粉,使其发光。但如果显示器在运输搬动时受到机械振动,使电子枪或金属荫罩移位,将会造成电子束不能准确着屏,轻则局部亮度变暗(荧光粉条局部发光),重则出现混色(电子束打在相邻的荧光粉条上)形成色斑。这时需要重调偏转线圈组件上的色纯度调节磁片,但因条件所限不易搞好。这时可以用另一种简单的施加外磁场的方法,来校正电子束的运动轨迹。先做一个小试验,用一把带磁性的金属起子靠近色斑,可以看到色斑被吸引而跟随起子移动到屏幕中间和屏外。同样道理,打开显示器后盖,选择一个小磁片(可用废耳机里的磁芯或把旧晶体管收音机里小喇叭的磁钢砸碎代用),放在彩显管后背上并调整其位置(注意安全,不要碰及高压帽),直到色斑在屏幕上消失,然后用胶布固定,即可恢复显示器的正常工作。

【例二】开机指示灯亮,有正常的行频“吱”声,但无显示,随后发现机内冒烟。

检查与维修

1. 由于机内有元件烧毁,首先用直观法和静态测量法进行检查。经仔细观察发现场扫描集成电路 IC301(TDA1170N)第②脚的外接电阻 R336 表面烧糊(经测量已开路),该电阻是 IC301 的供电限流电阻。R336 烧坏主要有三个原因:一是输入的 25V 电压过高;二是滤波电容 C317、C321、C322 短路或严重漏电;三是集成电路 IC301 损坏。检查电容是好的。

2. 检测 25V 供电是否正常。在没有更换 R336 之前,测量 25V 电源正常,由此判断开关电源是好的。

3. 检查场扫描集成电路 IC301。更换一只新 R336(1Ω/1W),再串接上一只 1A 保险丝后通电发现保险丝断,说明 IC301 的电流远大于正常值。为判断 IC301 是否损坏,测量其在路对地电阻,发现第④、第⑤脚对地电阻接近 0Ω。第④脚为 IC301 的输出脚,这说明场放大输出电路内部短路,故而场扫描电路电流增大。检查 D302 及其它外围元件均正常,更换一个新的 TDA1170N 后加电,显示正常,故障排除。

· 书 讯 ·

本刊有以下技术资料可供邮购(不另加邮费):

《计算机应用研究》90 年精装合订本	18 元
《计算机应用研究》91 年精装合订本	19 元
《计算机应用研究》92 年精装合订本	20 元
全国计算机应用研究学术交流会 89 年论文集	10 元
全国计算机应用研究学术交流会 90 年论文集(上下)	20 元
全国计算机应用研究学术交流会 91 年论文集	19 元
怎样使用电子现金出纳机	1.65 元
《计算机应用研究》92 技术专辑(3)	20 元

《计算机病毒大观》(上下)	8 元
计算机病毒分析与防治	5 元
IBM-PC 磁盘操作系统	3 元
DOS/BIOS 功能调用及程序实例	22.5 元
《四川省计算机开发应用指南》	7 元
APPLE II 微型机实用维修技术	1.5 元
开户行:成都工商行跳伞塔分理处	
帐号:89501299	
户名:四川省电子计算机应用研究中心	

好消息

本刊 92 年精装合订本和 92 专辑(3)现已出版,前者为烫金精装硬壳封面,后者为彩色封面,刊录 129 篇新颖实用的技术文章,适宜于图书馆上架、存档及个人查阅保管,数量有限,欲订购者可直接通过银行或邮局汇款,并注明汇款用途和所需书名、册数及订购者通讯地址及收件人姓名。