

高等院校计算机专业教育改革推荐教材

Windows 编程技术

张志强 编著

徐汀荣 审

6.7



机械工业出版社
CHINA MACHINE PRESS



高等院校计算机专业教育改革推荐教材

Windows 编程技术

张志强 编著

徐汀荣 审



机械工业出版社

本书是教育部世界银行贷款,21 世纪初高等理工科教育教学改革项目中,“以培养创新人才为目标的地方院校计算机科学与技术专业教学改革的研究与实践”成果之一,也是“计算机科学与技术”品牌与特色专业的建设成果之一。

本书内容以 VC++、.NET 和 Windows XP 为开发平台,面向具有一定 C++ 基础的读者,由浅入深,系统地讲授了使用 C++ 语言进行 Windows 程序设计的基本方法。考虑到教学或自学的方便,全书内容共分 14 章,可安排 18 个单元进行学习,每次学习或自学 1 章左右内容(第 5 章需两个单元),另外 3 个单元可作为机动或练习时间。本书第 1 章介绍了直接面向 Windows 操作系统的程序设计方法,第 2 章介绍了基于 MFC 的 Windows 程序设计方法,第 3 章讲授了 MFC 中提供的一些通用类的使用方法,第 4 章至第 9 章讲授 MFC 文档视图程序的工作原理和 design 方法,第 10 章至第 12 章讲授 MFC 下对话框程序和常用控件的使用方法,第 13 章讲授在程序中创建和使用动态链接库的方法,第 14 章讲授使用进程、线程进行并行、并发程序设计的基本方法。

本书既可作为高等院校计算机及相关专业 C++ 后续课程的教材或主要参考书,也可作为继续教育或网络培训中的程序设计课程教材,同时适用于有关工程技术人员和计算机爱好者学习参考。

图书在版编目(CIP)数据

Windows 编程技术/张志强编著. —北京:机械工业出版社,2003.6

高等院校计算机专业教育改革推荐教材

ISBN 7-111-12384-0

I. W... II. 张... III. 窗口软件, Windows—程序设计—高等学校—教材 IV. TP316.7

中国版本图书馆 CIP 数据核字(2003)第 046407 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策 划: 胡毓坚

责任编辑: 陈振虹

责任印制: 闫 焱

北京京丰印刷厂印刷·新华书店北京发行所发行

2003 年 7 月第 1 版·第 1 次印刷

787mm×1092mm $1/16$ ·18.75 印张·460 千字

0 001—5 000 册

定价: 26.00 元

凡购本图书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话(010) 68993821、88379646

封面无防伪标均为盗版

高等院校计算机专业教育改革推荐教材

编委会成员名单

主 编 刘大有

副主编 王元元

编 委 (按姓氏笔画排序)

李师贤 刘晓明 张桂芸 徐汀荣

耿亦兵 黄国兴 顾军华 薛永生

编者的话

计算机科学技术日新月异的飞速发展和计算机科学技术专业教育的相对滞后,已是不争的事实。

有两个发人深省的现象:一是,由于非计算机专业的学生既具有一门非计算机专业的专业知识,又具有越来越高的计算机应用技术水平,从而使计算机专业的学生感受到一种强烈的冲击和压力;二是,创建软件学院的工作已有近两年的历史,但软件学院的计算机专业教育的定位仍在探讨之中。

我们认为计算机科学与技术专业(以下简称计算机专业)教育的改革势在必行,正确认识和划分计算机专业教育的层次,对该专业的教育改革无疑是一个非常重要的问题。我国的计算机专业教育主要分三个层次。一般说来,这三个层次通常分布在以下三类高等院校:

第一层次主要以具有计算机一级学科博士学位授予权的教育部属重点高等院校为代表(包括具有两个博士点的大学)。这一类大学本科着重培养理论基础比较坚实、技术掌握熟练、有一定研究和开发能力的计算机专业学科型人才,其中部分学生(约本科生的10%)可攻读博士学位。

第二层次主要以具有一个计算机二级学科专业博士点的教育部属高等院校为代表。这一类高等院校本科着重培养有一定的理论基础、技术掌握比较熟练、有一定的研究或开发能力的计算机专业人才,其中一部分培养成学科型人才,另一部分培养成应用型人才,一小部分学生(约本科生的5%)可攻读博士学位。

第三层次主要以具有计算机二级学科专业硕士点的省属高等院校为代表。这一类高等院校本科面向企业应用,侧重培养对计算机技术或部分计算机技术掌握比较熟练,有一定的开发、应用能力的计算机专业应用型人才,其中很小一部分学生(约本科生的2.5%)可攻读博士学位。

国家教育部、计委批准的或省教育厅批准的示范性软件学院,就其培养目标和办学特色而言,分别与第二层次中应用型人才培养部分以及第三层次比较相近,但在如下方面有所不同:将软件工程课程作为专业教学重点;更加强调英语教学,更加重视实践能力培养,并对两者有更高的要求。

我们本着对高等院校的计算机专业状况的认识,主要面向与上述第二、第三两个层次对应的院校及与之相近的软件学院,总结多年的计算机专业的教改经验,在一定程度上溶入了ACM & IEEE CC2001和CCC2002(中国计算机科学与技术学科教程)的教改思路,组织我国一直投身于计算机教学和科研的教师,编写了这套“高等院校计算机专业教育改革推荐教材”(以下简称“推荐教材”)。自然,“推荐教材”中所贯穿的改革思路和做法,也是针对上述第二、第三两个层次对应院校的计算机专业学生。这些思路和做法可概括成以下三句话:

- 适度调整电子技术基础、计算机理论基础和系统软件的教学内容。
- 全面强化计算机工具软件、应用软件的教学要求。
- 以应用为目标大力展开软件工程的教学与实践。

电子技术基础、计算机理论基础、系统软件教学关系到学生的基本素质、发展潜力和日后

的应变能力。“推荐教材”在调整它们的教学内容时的做法是:适度压缩电子线路、数字电路和信号系统的教学内容,变三门课程为两门,并插入数字信号处理的基础内容;合并“计算机组成原理”、“微型计算机接口技术”和“汇编语言”为“计算机硬件技术基础”一门课程;注意适当放宽“离散数学”课程的知识面,使之与 CCC2002 的要求基本接轨,但适度降低其深度要求;更新系统软件课程的教学内容,以开放代码的 Linux 作为操作系统原理的讲授载体,更加关注系统软件的实践性和实用性。

为了提高计算机专业人才的计算机应用能力,全面强化计算机工具软件、实用软件的教学要求是十分重要的,这也是上述改革思路的核心。为此,“系列教材”的做法是:强化程序设计技术,强化人机接口技术,强化网络应用技术。

为强化程序设计技术,“推荐教材”支持在单片机环境、微机平台、网络平台的编程训练;支持运用程序设计语言、程序设计工具以及分布式对象技术的编程训练。大大加强面向对象程序设计课程的组合(设计了三门课程:面向对象的程序设计语言 C++,面向对象的程序设计语言 JAVA 和分布式对象技术),方便教师和读者的选择。

为强化人机接口技术,“推荐教材”设计了“人机交互教程”,“计算机图形学”和“多媒体应用技术”等可供选择的、有层次特色的课程组合。

为强化网络应用技术,“推荐教材”设计了“计算机网络技术”,“计算机网络程序设计”,“计算机网络实验教程”和“因特网技术及其应用”等可供选择的、新颖丰富的课程组合。

将软件工程课程作为专业教学重点,以应用为目标大力展开软件工程的教学与实践,是“推荐教材”改革思路的又一亮点。为改变以往软件工程课程纸上谈兵的老毛病,“推荐教材”从工程应用出发,理论联系实际,突出建模语言及其实现工具的运用,设计了“软件工程的方法与实践”,“统一建模语言 UML 导论”和“ROSE 对象建模方法与技术”等可供选择的、创新独特的软件工程课程组合。对于各类软件学院,“推荐教材”的这一特色无疑是很有吸引力的。

强调实践也是计算机学科永恒的主题,对计算机应用专业的学生来说更是如此。重应用和重实践是“推荐教材”的一个整体特点。这一特点,一方面有利于解决本文开始所指出的计算机专业学生较之非计算机专业学生,在应用开发工作中上手慢的问题;另一方面,使计算机专业的学生能在更大范围内、更高层面上掌握计算机应用技术。这一特点正是许多高等院校计算机专业教育改革追求的一个目标,也是国家教育部倡导软件学院的初衷之一。

“推荐教材”由基础知识、程序设计、应用技术、软件工程和实践环节等五个模块组成。各模块有其对应的培养目标与功能,从而构架出一个创新的、完整的计算机应用专业的课程体系。模块化的设计,使各学校可根据学生及学校的特点做自由的选择和组合,既能达到本专业的总体要求,又能体现具有特色的个性发展。整套教材的改革脉络清晰,结构特色鲜明,值得各高等院校在改革教学内容、编制教学计划、挑选教材书目时借鉴和参考。当然,很多书目也适合很多相关学科的计算机课程用作教材。

“推荐教材”的组成模块和书目详见封底。显然它不能说是完备的(实践环节模块更是如此),其改革的思路、改革的举措也可能有值得探讨的地方。我们衷心希望得到计算机教育界同仁和广大读者的批评指正。

高等院校计算机专业教育改革推荐教材

编委会

前 言

本书是教育部世界银行贷款,21 世纪初高等理工科教育教学改革项目中“以培养创新人才为目标的地方院校计算机科学与技术专业教学改革的研究与实践”成果之一,也是“计算机科学与技术”品牌与特色专业的建设成果之一。

Windows 编程是目前应用最广泛的一种编程技术,但由于 Windows 操作系统本身比较复杂,要想真正掌握 Windows 编程的核心技术确实比较困难。目前 Windows 下软件开发方式主要有两种:

一种是使用 VB、Delphi 等进行开发,由于开发工具将大量的 Windows 内部技术细节屏蔽掉,从而简化了 Windows 编程的难度,极大地提高了开发的速度。

另一种是使用 C、C++ 等语言采用直接面向 Windows 系统的开发方法,由于 Windows 系统本身主要采用 C 语言编写,所以采用此方法可以实现与操作系统的无缝对接,最大限度地发挥系统的潜能,但难度较大。

两种方法各有优缺点,好比自动挡与手动挡汽车,若追求方便和上手容易,请选择自动挡,但作为专业车手,手动挡将是惟一选择。因此,目前市场上可见的通用商业软件,不论是 MSOffice、WPS 等办公软件,AutoCAD、PhotoShop 等专业软件,还是著名的 CS、星际等游戏软件无一不是 C、C++ 的杰作。本书主要面对想成为专业程序员的读者,讲授以 VC++ .NET 为开发平台,使用 C++ 进行 Windows 应用软件开发的标准方法。

本书的编写目标是使学生向开发人员过渡,使计算机编程爱好者由业余水平向专业水平过渡。本书属于 C++ 高级编程技术,要求读者已经掌握 C++ 基本知识,所以最好安排在 C++ 课程之后。本书内容由 14 章组成,采用原理和实例相结合的方法系统讲授 Windows 编程技术,目标是使读者在学习完本书后可以参照 VC++ .NET 的联机文档,编写出能满足一般需要的各种 Windows 应用程序,并以此为基础,通过自学掌握更加高级的 Windows 编程技术。

本书由张志强编著。由于掌握基于 C++ 的 Windows 编程技术确实有一定的难度,为了使读者少走弯路,金震江、杨跃、唐越峰、陈哲凡等多位有经验的 Windows 开发人员结合自己的学习经验为本书内容的编排提供了大量的帮助。本书由徐汀荣教授进行了审阅,并提出了大量宝贵意见,唐金华老师也为本书的完成花费了大量的精力,在此谨向他(她)们表示诚挚的谢意。

由于作者水平所限,书中难免有错误和疏漏,欢迎读者批评指正。

编 者

目 录

编者的话

前言

第 1 章 Windows 程序设计基础	1
1.1 Windows 概述	1
1.1.1 Windows 的发展历史	1
1.1.2 Windows 的技术特点	2
1.2 Windows 程序特点	3
1.2.1 程序组成	3
1.2.2 数据类型	4
1.2.3 匈牙利命名法	6
1.2.4 Windows 消息	7
1.2.5 运行模式	7
1.3 简单的 Windows 程序介绍	8
1.3.1 开始之前	9
1.3.2 WinMain 函数	9
1.3.3 注册窗口类	10
1.3.4 建立窗口	12
1.3.5 窗口函数	13
1.3.6 窗口内的输出	14
1.4 上机步骤	15
1.4.1 进入开发环境	15
1.4.2 新建项目	16
1.4.3 修改项目选项	17
1.4.4 添加程序文件	17
1.4.5 编辑程序	17
1.4.6 调试运行	17
1.5 小结	18
1.6 练习题	18
第 2 章 MFC 程序设计基础	19
2.1 MFC 概述	19
2.1.1 MFC 与应用程序框架	19
2.1.2 MFC 的组成	19
2.1.3 MFC 中的类	20

2.2	MFC 程序的特点	21
2.2.1	MFC 与消息处理	21
2.2.2	运行模式	21
2.3	简单的 MFC 程序	22
2.3.1	开始之前	22
2.3.2	派生应用程序类	23
2.3.3	派生窗口类	23
2.3.4	建立窗口	24
2.3.5	加入消息映射	24
2.3.6	建立 MFC 应用程序对象	25
2.4	上机步骤	25
2.4.1	建立项目	25
2.4.2	修改项目属性	26
2.4.3	调试运行	26
2.5	使用资源	26
2.5.1	建立图标资源	26
2.5.2	在程序中使用图标资源	28
2.6	小结	28
2.7	练习题	29
第 3 章	常用 MFC 通用类	30
3.1	CRect、CPoint、CSize 类	30
3.1.1	CRect 类	30
3.1.2	CPoint 类	30
3.1.3	CSize 类	30
3.2	CString 类	31
3.2.1	创建字符串	31
3.2.2	访问字符串数据	32
3.2.3	字符串的比较	33
3.2.4	子串函数	34
3.2.5	字符串处理函数	35
3.3	集合类	36
3.3.1	列表类 CList	36
3.3.2	数组类 CArray	39
3.3.3	映射类 CMap	41
3.4	时间处理类	42
3.4.1	COleDateTime 类	42
3.4.2	COleDateTimeSpan 类	46
3.5	异常处理类	47
3.5.1	异常处理类的使用	47

3.5.2 MFC异常处理类简介	48
3.6 小结	49
3.7 练习题	49
第4章 文档视图程序	51
4.1 概述	51
4.2 文档	51
4.2.1 单文档对象程序	51
4.2.2 多文档对象程序	51
4.2.3 多文档类型程序	52
4.3 视	53
4.3.1 一个文档对象对应一个视对象	53
4.3.2 一个文档对象对应多个视对象	53
4.3.3 一个文档对应多个视类型	54
4.4 窗口框架	54
4.5 文档模板	55
4.6 建立标准单文档程序框架	55
4.6.1 新建项目	56
4.6.2 修改项目选项	56
4.6.3 完成向导	59
4.6.4 运行结果	59
4.7 单文档程序框架分析	60
4.7.1 Stdafx.h	60
4.7.2 Stdafx.cpp	61
4.7.3 0401.h	61
4.7.4 0401.cpp	62
4.7.5 MainFrm.h	66
4.7.6 MainFrm.cpp	67
4.7.7 TextDoc.h	70
4.7.8 TextDoc.cpp	71
4.7.9 TextView.h	73
4.7.10 TextView.cpp	75
4.8 小结	77
4.9 练习题	77
第5章 绘图输出	79
5.1 GDI与MFC	79
5.1.1 设备环境类	79
5.1.2 GDI对象类	80
5.2 绘制图形	81

5.2.1 开始绘图	81
5.2.2 更改画笔	84
5.2.3 使用画刷	85
5.2.4 使用位图	86
5.3 输出文本	91
5.3.1 文本输出函数	91
5.3.2 更改文本颜色	92
5.3.3 更改字型、字体	94
5.4 坐标与坐标模式	97
5.4.1 坐标模式	98
5.4.2 MM_ANISOTROPIC 坐标模式	99
5.4.3 MM_ISOTROPIC 坐标模式	101
5.4.4 平移坐标	102
5.4.5 修改坐标方向	102
5.5 小结	104
5.6 练习题	105
第6章 用户输入	106
6.1 鼠标和键盘	106
6.1.1 鼠标消息	106
6.1.2 键盘消息	110
6.2 使用菜单	111
6.2.1 编辑菜单	112
6.2.2 处理菜单命令	113
6.2.3 修改菜单状态	114
6.2.4 使用弹出式菜单	117
6.3 小结	118
6.4 练习题	119
第7章 工具栏与状态栏	120
7.1 工具栏	120
7.1.1 编辑工具栏	120
7.1.2 处理工具栏命令	121
7.1.3 工具栏按钮状态	124
7.1.4 自定义工具栏	124
7.2 状态栏	129
7.2.1 默认状态栏	129
7.2.2 操作状态栏	131
7.3 小结	136
7.4 练习题	136

第 8 章 使用文件	137
8.1 CFile 类	137
8.1.1 文件的建立、打开和关闭	137
8.1.2 文件的读、写	139
8.1.3 文件内容的定位与锁定	140
8.1.4 获取、设置文件状态	141
8.2 序列化	142
8.2.1 序列化的概念	142
8.2.2 Serialize 函数	142
8.2.3 CArchive 类	143
8.2.4 序列化应用实例	145
8.3 文件管理	148
8.3.1 文件操作	148
8.3.2 目录操作	149
8.4 小结	150
8.5 练习题	150
第 9 章 常用消息	151
9.1 消息的分类	151
9.2 系统消息	151
9.2.1 WM_CREATE	151
9.2.2 WM_CLOSE	152
9.2.3 WM_QUERYENDSESSION	152
9.2.4 WM_DESTROY	152
9.2.5 WM_NCDESTROY	153
9.2.6 WM_TIMER	153
9.2.7 WM_PAINT	154
9.2.8 程序实例	155
9.3 用户自定义消息	157
9.3.1 消息标识	157
9.3.2 消息映射宏	157
9.3.3 消息发送	158
9.3.4 消息接收	158
9.3.5 使用系统热键消息	159
9.3.6 使用进程间消息	162
9.4 小结	165
9.5 练习题	165
第 10 章 对话框	167
10.1 概述	167

10.2 对话框模板资源	167
10.2.1 创建对话框模板资源	167
10.2.2 编辑对话框模板属性	167
10.2.3 在对话框模板中添加或删除控件	169
10.2.4 在对话框模板上调整控件	170
10.2.5 修改控件的属性	170
10.2.6 设定控件跳格次序	171
10.2.7 测试对话框模板资源	172
10.3 对话框类	172
10.3.1 模式对话框	172
10.3.2 模式对话框例子	172
10.3.3 非模式对话框	173
10.3.4 非模式对话框例子	173
10.4 自定义对话框	175
10.4.1 建立模板资源	175
10.4.2 建立 CDialog 派生类	175
10.4.3 为控件添加消息映射	175
10.4.4 为控件添加成员变量	177
10.4.5 使用 CDialog 派生类对象	178
10.5 基于对话框的 MFC 程序	178
10.5.1 建立项目	179
10.5.2 对话框程序框架分析	180
10.6 通用对话框	185
10.6.1 CFileDialog	186
10.6.2 CColorDialog	188
10.6.3 CFontDialog	189
10.7 小结	190
10.8 练习题	191
第 11 章 常用控件	192
11.1 概述	192
11.1.1 控件消息	192
11.1.2 通知消息	192
11.1.3 控件的创建	192
11.2 按钮(BUTTON)	192
11.2.1 按钮样式	193
11.2.2 按钮控件的通知消息	193
11.2.3 在对话框中使用按钮的例子	193
11.3 文本编辑框(EDITBOX)	195
11.3.1 文本编辑框的样式	195

11.3.2	文本编辑框控件的通知消息	195
11.3.3	在对话框中使用编辑框控件	196
11.4	静态控件(STATIC)	197
11.4.1	静态控件的样式	197
11.4.2	静态控件的通知消息	198
11.4.3	在对话框中使用静态控件	198
11.5	复选框(CHECKBUTTON)	199
11.5.1	复选框的样式	199
11.5.2	复选框的通知消息	199
11.5.3	在对话框中使用复选框	200
11.6	单选按钮(RADIOBUTTON)	200
11.6.1	单选按钮的样式	200
11.6.2	单选按钮的通知消息	200
11.6.3	在对话框中使用复选框和单选按钮	200
11.7	进程条(PROGRESS)	201
11.7.1	进程条的样式	201
11.7.2	进程条的成员函数	201
11.7.3	在对话框中使用进程条控件	202
11.8	滑动条(SLIDER)	203
11.8.1	滑动控件的样式	203
11.8.2	滑动控件的通知消息	203
11.8.3	微调控件的成员函数	203
11.8.4	在对话框中使用滑动条控件	204
11.9	Spin 控件	204
11.9.1	微调控件的样式	205
11.9.2	微调控件的通知消息	205
11.9.3	微调控件的成员函数	205
11.9.4	在对话框中使用微调控件	206
11.10	组合框(COMBOBOX)	207
11.10.1	组合框的样式	207
11.10.2	组合框的常用通知消息	208
11.10.3	在对话框中使用组合框	208
11.11	小结	209
11.12	练习题	209
第 12 章	使用树控件和列表控件	211
12.1	图像列表	211
12.1.1	建立图像列表	211
12.1.2	图像列表的操作	212
12.2	树控件	213

12.2.1	树控件的创建	214
12.2.2	树控件的操作	217
12.3	列表控件	220
12.3.1	列表控件的创建	220
12.3.2	列表控件的操作	224
12.4	小结	227
12.5	练习题	227
第 13 章	动态链接库	228
13.1	概述	228
13.2	创建标准 Win32 动态库	228
13.2.1	新建标准 Win32 动态库项目	228
13.2.2	添加函数	229
13.2.3	添加类	231
13.2.4	导出函数	232
13.2.5	导出自定义类	232
13.2.6	生成 DLL	232
13.3	创建标准 MFC 动态库	232
13.3.1	新建 MFC 动态库项目	232
13.3.2	添加自定义函数	234
13.3.3	添加自定义类	236
13.3.4	导出自定义函数	236
13.3.5	导出自定义类	236
13.4	在程序中使用动态库	237
13.4.1	加载 DLL 的方式	237
13.4.2	使用隐式加载	237
13.4.3	使用显式加载	239
13.5	小结	240
13.6	练习题	241
第 14 章	使用多任务	242
14.1	进程	242
14.1.1	进程的优先级	242
14.1.2	启动进程	243
14.1.3	进程的管理	245
14.2	线程	252
14.2.1	线程的优先级	252
14.2.2	线程的创建和终止	253
14.2.3	使用工作者线程	254
14.2.4	使用用户接口线程	255

14.3 进程、线程间的同步	256
14.3.1 等待函数	257
14.3.2 使用事件	258
14.3.3 使用临界段	259
14.3.4 使用互斥量	261
14.3.5 使用信号量	263
14.4 小结	264
14.5 练习题	265
附录	266
附录 A Windows 窗口样式	266
附录 B Windows 虚键码表	269
附录 C 常用数据结构	271
参考文献	282

第 1 章 Windows 程序设计基础

本章介绍 Windows 操作系统的发展历史及技术特点,Windows 应用程序的运行模式、基本结构以及开发流程。

1.1 Windows 概述

操作系统是计算机系统中最重要软件,它负责管理计算机系统各种软、硬件资源,负责启动、控制、结束并协调各种应用程序的运行。一个应用程序若想正确运行必须严格执行操作系统中定义的各种规则;一个应用程序若想使用计算机中的软、硬件资源必须借助操作系统才能够使用。所以若要进行实际的软件开发,仅仅掌握一门或几门程序设计语言是不够的,还必须学习在相应操作系统下应用程序的开发规则和方法。

Windows 是目前最流行的操作系统,有着最大量的软件开发需求。同时,Windows 也是广大用户最熟悉和最容易得到的操作系统,所以 Windows 应用程序的开发有着广泛的应用前景和方便的学习环境。在学习 Windows 下应用程序的开发方法之前,读者有必要先了解一下 Windows 操作系统的发展历史和技术特点。

1.1.1 Windows 的发展历史

所有主要的计算机技术都经历了一个从产生到发展到成熟的过程:

(1) 最早的计算机系统中并没有操作系统,程序设计全部使用机器语言,人们使用电路插板及硬连线来控制计算机。

(2) 20 世纪 50 年代后期,随着 Fortran、Algol、Cobol 等语言的发明和计算机功能的增强,人们开发了一些监督程序来对计算机的运行进行简单的控制和管理。

(3) 20 世纪 60 年代以后,这些监督、管理程序的功能不断增强,逐渐发展成为简单的操作系统,其中比较典型的有 Fortran 监控系统和 IBMSYS 等。

(4) 20 世纪 70 年代初,AT&T 发明了 C 语言并利用它开发了 UNIX 操作系统,这是计算机技术的一场革命,它标志着现代操作系统的产生。到目前为止,所有流行的主要操作系统都受到了 UNIX 的影响,且都采用 C 语言开发完成,Windows 操作系统也不例外。

较早的操作系统都是命令行界面,需要通过键盘键入不同的命令来完成不同的操作,为了完成一个复杂的功能往往需要键入多个命令,不但工作效率低,而且有大量的命令需要记忆,使一般用户很难掌握。

1981 年,美国 Xerox 公司开发了世界上第一个商用图形界面操作系统,它在 Star8010 工作站上运行,其新颖的操作界面引起了当时一些新兴计算机公司的兴趣。苹果电脑公司的主要创始人 Steve Jobs 在参观了 Xerox 公司的 PARC 研究中心后认识到了图形用户接口(通常简称为 GUI)的重要性以及广阔的市场前景,开始着手进行自己的 GUI 系统研究开发工作,并于 1983 年研制成功了苹果公司第一个基于图形界面的操作系统“Apple Lisa”。随后不久,苹