



编程之道



<http://t.sina.com.cn/jamjar>

4, print(78)

人民邮电出版社
POSTS & TELECOM PRESS

图书在版编目 (C I P) 数据

C语言编程之道 / 刘彬彬, 孙秀梅, 李鑫编著. --
北京: 人民邮电出版社, 2011. 3
ISBN 978-7-115-24546-5

I. ①C… II. ①刘… ②孙… ③李… III. ①
C语言—程序设计 IV. ①TP312

中国版本图书馆CIP数据核字(2010)第239923号

内 容 提 要

这是一本集编程感悟、技巧、陷阱、谬误和算法为一体的书籍,旨在引导读者如何正确地学习编程,帮助读者丰富编程经验。全书分为5篇,共15章,全面介绍了学习和应用C语言进行程序开发的各种陷阱与谬误、技术细节与经验技巧、常用的算法等。其涉及的技术主要包括C语言关键技术、指针、数组、数据结构、数学应用、算法等相关技术细节、技巧。

本书附带有配套DVD光盘,光盘中提供了书中全部实例和项目的源代码,这些源代码都是经过作者精心调试通过的,保证能够在Windows XP、Windows Server 2003操作系统下编译和运行。

本书非常适合大中专院校在校生、毕业生、求职者、编程爱好者学习,同时也可以作为想要学习编程的初学者的指导用书。

C 语言编程之道

- ◆ 编 著 明日科技 刘彬彬 孙秀梅 李 鑫
责任编辑 蒋 佳
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
大厂聚鑫印刷有限责任公司印刷
- ◆ 开本: 800×1000 1/16
印张: 24.25
字数: 591千字 2011年3月第1版
印数: 1—3500册 2011年3月河北第1次印刷

ISBN 978-7-115-24546-5

定价: 59.00元(附光盘)

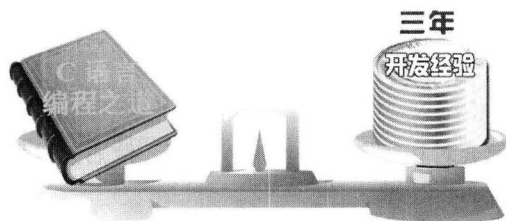
读者服务热线: (010)67132692 印装质量热线: (010)67129223

反盗版热线: (010)67171154

前言

许多求职人员在面试时经常遇到的一个问题是“您有几年开发经验？”这对于一个刚刚走出校园的毕业生来说简直就是一个“原子弹”。“经验”真的很重要吗？是的，它是衡量个人能力的重要准则，它将许多求职者挡在门外。因此，经验真的很重要！

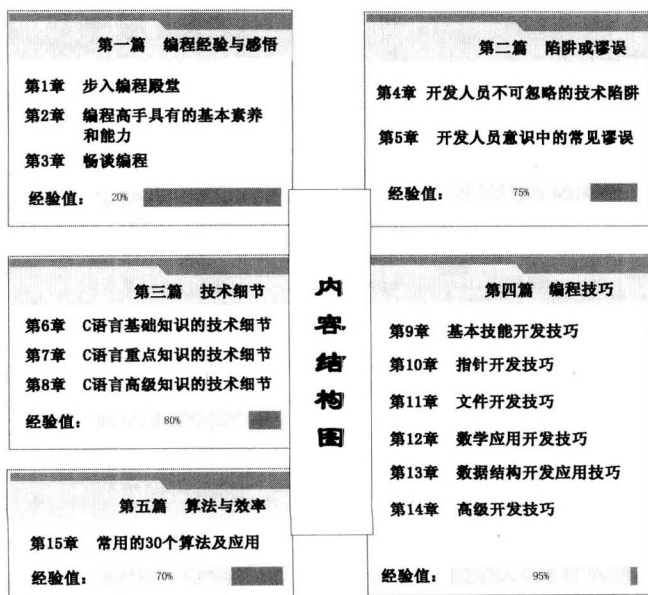
本书不是基础类书籍，不是项目、案例类书籍，而是一部丰富编程经验的书籍。它涵盖了 63 个编程细节、117 个编程技巧、22 个编程陷阱、36 个编程谬误、30 个优化算法。这些内容是作者结合自己多年的实际开发经验，精心筛选和提炼而成的。如果您想提高自己的编程经验，本书将是最佳的选择！



本书内容

本书分 5 篇，共 15 章。主要包括编程经验与感悟篇、陷阱与谬误篇、技术细节篇、编程技巧篇、算法和效率篇等内容。

本书内容结构如下图所示。



本书读者人群

本书非常适合以下人员阅读。

- C 语言求职人员
- 大中专院校的在校生和毕业生
- 相关培训机构的学员
- 语言已经入门，急需进一步提高的编程人员
- 编程爱好者
- 有多年开发经验的程序员

学习答疑

如果您在学习或使用本书的过程中遇到问题或疑惑，可以通过如下方式与我们联系。

- 服务网站: www.mingribook.com
- 服务 QQ: 100310286 (如要加入该 QQ, 请标明是本书读者)
- 服务信箱: mingrisoft@mingrisoft.com
- 服务电话: 0431-84978981/84978982
- 学习社区: bbs.mrbccd.com

我们承诺将在 5 个工作日内给您提供解答。

本书作者

本书由明日科技组织编写，参加编写的有刘彬彬、孙秀梅、李鑫、刘锐宁、梁水、李伟明、潘凯华、杨明、刘欣、李慧、董大永、李继业、尹强、张磊、赛奎春、高春艳、宋坤、安剑、巩建华、房大伟、吕双、刘云峰、王国辉、李钟尉、张振坤、陈丹丹、卢瀚、白伟明、王小科、梁晓岚、杨丽、顾丽丽、刘龄龄、王乐乐、杨明、朱晓、陈英、刘莹、李贺、肖鑫、王雷等。由于作者水平有限，疏漏和不足之处在所难免，请广大读者朋友批评指正。

明日科技
2010 年 12 月

目 录

第1篇 编程经验与感悟——站在前人肩膀上前行

第1章 步入编程殿堂.....2

- 感悟 01 编程我也行.....2
- 感悟 02 如何选择编程语言.....2
- 感悟 03 培养对编程的兴趣.....4
- 感悟 04 学习编程要打好基本功.....6
- 感悟 05 多与别人交流.....7
- 感悟 06 锻炼开放性思维.....7
- 感悟 07 思维方式要灵活.....9
- 感悟 08 不断温习学过的知识.....9
- 感悟 09 坚持比激情更重要.....9

第2章 编程高手具有的基本素质和能力.....11

- 经验 01 善于总结编程经验.....11
- 经验 02 创造或寻找方法.....11
- 经验 03 重视程序中的小错误.....12
- 经验 04 在调试与改错中成长.....12
- 经验 05 养成良好的编程风格.....13

- 经验 06 要重视实践练习.....14
- 经验 07 善于利用丰富的网络资源.....15
- 经验 08 学习与培训的必要性.....15
- 经验 09 多向他人学习.....15
- 经验 10 写程序之前要分析.....16
- 经验 11 写程序之后要总结.....17
- 经验 12 软件需求规格说明书.....18
- 经验 13 逐步培养个人的编程思想.....18
- 经验 14 面试经验谈.....19

第3章 畅谈编程.....21

- 感悟 01 程序员的成长之路.....21
- 感悟 02 程序员的生存法则.....22
- 感悟 03 程序员的生涯规划.....23
- 感悟 04 项目开发经验谈.....24
- 感悟 05 信息技术改变我们的生活.....26

第2篇 陷阱与谬误——汲取前人的教训

第4章 开发人员不可忽略的

22个技术陷阱.....30

- 陷阱 01 忽略了大小写的区别.....30
- 陷阱 02 “{}”与“()”使用不当
造成错误.....31
- 陷阱 03 多加分号导致出错.....32
- 陷阱 04 使用循环语句不慎
产生死循环.....33
- 陷阱 05 忽视 do...while 循环次数
产生的错误.....34
- 陷阱 06 数据溢出产生的陷阱.....35
- 陷阱 07 整数的除法导致
结果的偏差.....36

- 陷阱 08 变量未定义.....37
- 陷阱 09 输入数据时忘记添加&符号.....38
- 陷阱 10 使用 scanf 函数输入
格式不符.....39
- 陷阱 11 输入的数据类型不符的陷阱.....41
- 陷阱 12 switch 语句中忘记
使用 break.....42
- 陷阱 13 指针没被初始化.....43
- 陷阱 14 操作超出动态分配的内存.....44
- 陷阱 15 使用内存后未释放.....45
- 陷阱 16 内存释放了继续使用.....46
- 陷阱 17 没有说明函数的头文件.....47
- 陷阱 18 忽略了字符串结束符.....48

陷阱 19	数组长度使用变量	49	谬误 14	路径名错误	61
陷阱 20	数组下标“越界”	51	谬误 15	#include <stdio.h>和 #include "stdio.h"的区别	61
陷阱 21	子函数未定义产生的陷阱	52	谬误 16	函数调用的错误	61
陷阱 22	子函数在说明和定义时 类型不一致	53	谬误 17	空指针并非空字符串	62
第 5 章	36 个开发人员意识中的常见谬误	55	谬误 18	枚举类型容易出错的 几个问题	62
谬误 01	Main 大写错误	55	谬误 19	&、 、~与&&、 、! 的 谬误	63
谬误 02	“=”与“==”	55	谬误 20	system()和 clrscr()的区别	63
谬误 03	scanf()错误	56	谬误 21	使用 getchar()函数的误区	63
谬误 04	循环错误	57	谬误 22	使用 errno 检测错误	64
谬误 05	在定义数组时, 将定义的 “元素个数”误认为是可 使用的最大下标值	57	谬误 23	宏定义中不可忽视的空格	65
谬误 06	混淆了字符与字符串 常量	57	谬误 24	*p++是自增 p 还是 p 做 指向的变量	65
谬误 07	使用==比较字符串	58	谬误 25	char a[]和 char *a 是一样的	65
谬误 08	没有使用零字符串结束 字符串	58	谬误 26	指针变量与零值的比较	66
谬误 09	不正确的使用 fgetc()等 函数	58	谬误 27	const 和#define 是一样的	66
谬误 10	a 和&a 的区别	59	谬误 28	浮点变量与零值的比较	66
谬误 11	gets()函数的使用	60	谬误 29	宏定义中的常见谬误	67
谬误 12	忽略了变量的类型, 进行了 不合法的运算	60	谬误 30	a[i]=i++;代码正确吗	68
谬误 13	忘记加分号	60	谬误 31	数组和指针等价吗	68
			谬误 32	指针和数组的区别	68
			谬误 33	指针比数组执行快	69
			谬误 34	++i 比 i=i+1 执行快	69

第 3 篇 技术细节——修炼编程内功心法

第 6 章	26 个 C 语言基础知识技术细节	72	细节 05	字符数据类型	74
细节 01	标识符的使用	72	细节 06	说明变量和定义变量	75
细节 02	限定词 const	73	细节 07	typedef 应用的技术要点	76
细节 03	限定词 volatile	73	细节 08	使用 static 关键字声明 静态变量	77
细节 04	变量的存储	74			

细节 09	如何区分“,”是运算符 还是分隔符	78
细节 10	基本的算术运算	79
细节 11	自增、自减操作符的使用	81
细节 12	C 语言中的特殊表达式	82
细节 13	特殊表达式 e1?e2:e3	83
细节 14	使用 printf 函数输出%	83
细节 15	%lf 与 %f 差异	84
细节 16	scanf 与 gets	85
细节 17	什么时候用 switch	86
细节 18	if 语句的多个判断问题	88
细节 19	break 在 switch 语句中的 使用	89
细节 20	while 与 do...while 的异同	90
细节 21	while、do...while、for 循环相互转换	91
细节 22	循环不能交叉	92
细节 23	goto 语句使用规则	93
细节 24	合理使用 return 语句	93
细节 25	合理使用空格使 代码更规范	94
细节 26	C 语言如何解释 $a < b < c$	94
第 7 章	19 个 C 语言重点知识技术细节	96
细节 01	判断数组中元素的 个数	96
细节 02	向函数传递二维数组	97
细节 03	字符数组的结束标志	97
细节 04	extern 声明数组	98
细节 05	数组下标从 0 开始	101
细节 06	数组名作函数参数与 数组元素做函数参数的 不同之处	101

细节 07	‘\0’ 和 “\0” 的区别	102
细节 08	静态全局变量与全局变量	102
细节 09	访问寄存器变量地址	102
细节 10	参数校验	103
细节 11	什么是空指针	103
细节 12	if (p) 检查指针 是否为空	104
细节 13	指针变量赋值相关	104
细节 14	const 定义指针变量	105
细节 15	char a[] 与 char *a	106
细节 16	指针应用	107
细节 17	指针类型和指针 所指向的类型	108
细节 18	指针运算与取地址 运算符的联系	109
细节 19	函数型指针	109
第 8 章	18 个 C 语言高级技术细节	111
细节 01	定义结构体对结构体 成员的类型限制	111
细节 02	成员运算符 (.) 和 (->) 的差异	111
细节 03	结构体和联合体的 区别和联系	111
细节 04	enum 的好处	112
细节 05	.h 文件	113
细节 06	宏定义问题	114
细节 07	条件编译	115
细节 08	文件打开	115
细节 09	C 语言中的预处理功能	117
细节 10	使用宏时需要 注意些什么	117
细节 11	文件包含的嵌套	117

细节 12	malloc()与 calloc()的区别	118
细节 13	什么是“内存泄露”	119
细节 14	内存耗尽了怎么办	119
细节 15	常见的内存管理错误有哪些	120

细节 16	标准文件和一般文件有什么区别	120
细节 17	C 语言的常用库文件包含哪些	120
细节 18	为什么文件打开以后要及时关闭	122

第 4 篇 编程技巧——程序员的神兵利器

第 9 章 C 语言开发人员必知必会的

基本技能开发技巧

技巧 01	十进制数转换为十六进制	124
技巧 02	N 进制转换为十进制	125
技巧 03	IP 地址形式输出	126
技巧 04	数字由小到大排序	127
技巧 05	阶梯问题	129
技巧 06	判断闰年	130
技巧 07	对调数问题	131
技巧 08	评定成绩等级	133
技巧 09	打印乘法口诀表	134
技巧 10	打印杨辉三角	135
技巧 11	特殊等式	137
技巧 12	统计各种字符个数	138
技巧 13	猴子吃桃	139
技巧 14	删除字符串中连续字符	140
技巧 15	求学生的平均身高	141
技巧 16	灯塔数量	143
技巧 17	老师分糖果	144
技巧 18	小球下落问题	145
技巧 19	彩球问题	146

第 10 章 指针开发技巧

技巧 01	使用指针实现整数排序	148
技巧 02	使用指针实现数据交换	149

技巧 03	指向结构体变量的指针	151
技巧 04	使用指针输出数组元素	152
技巧 05	用指针实现逆序存放数组元素值	153
技巧 06	使用指针查找数列中最大值/最小值	154
技巧 07	使用指针实现字符串复制	156
技巧 08	使用指针实现字符串的连接	158
技巧 09	使用指针实现字符串插入	159
技巧 10	使用指针实现字符串的匹配	160
技巧 11	用指向函数的指针比较大小	162
技巧 12	使用返回指针的函数查找最大值	163
技巧 13	用指针函数实现求学生成绩	164
技巧 14	寻找指定元素的指针	165
技巧 15	寻找相同元素的指针	167
技巧 16	使用指针的指针输出字符串	168
技巧 17	实现输入月份输出该月份英文名	170

技巧 18	使用指向指针的指针对 字符串排序	171	技巧 09	满足 $abcd=(ab+cd)^2$ 的数	209
第 11 章	文件开发技巧	173	技巧 10	神奇的数字 6174	210
技巧 01	读取磁盘文件	173	技巧 11	一数三平方	211
技巧 02	将数据写入磁盘文件	175	技巧 12	求等差数列	213
技巧 03	格式化读写文件	176	技巧 13	求整数的绝对值	214
技巧 04	成块读写操作	177	技巧 14	正弦、余弦、正切值	215
技巧 05	随机读写文件	179	技巧 15	自然对数的底 e 的计算	216
技巧 06	复制文件内容到另一文件	181	技巧 16	最大公约数及最小公倍数	216
技巧 07	合并两个文件信息	182	技巧 17	求解二元一次不定方程	217
技巧 08	文件分割	183	技巧 18	二分法求解方程	219
技巧 09	文件加密	185	技巧 19	牛顿迭代法解方程的根	220
技巧 10	统计文件内容	186	技巧 20	打印特殊方阵	221
技巧 11	创建文件	188	技巧 21	求 3×3 矩阵对角元素之和	222
技巧 12	创建临时文件	189	技巧 22	矩阵的加法运算	223
技巧 13	重命名文件	190	技巧 23	矩阵的乘法运算	224
技巧 14	删除文件	191	技巧 24	打印 n 阶螺旋方阵	225
技巧 15	删除文件中的记录	192	技巧 25	求车运行速度	226
技巧 16	查找文件	193	技巧 26	卖西瓜	228
技巧 17	显示目录内同类型文件	194	技巧 27	巧分苹果	228
技巧 18	关闭打开的所有文件	196	第 13 章	C 语言开发人员必知必会的	
技巧 19	错误处理	198		数据结构开发应用技巧	230
第 12 章	数学应用开发技巧	201	技巧 01	结构体简单应用	230
技巧 01	求 100~200 之间的 素数	201	技巧 02	找最高分	233
技巧 02	可逆素数	202	技巧 03	平均成绩	235
技巧 03	回文素数	203	技巧 04	比较计数	235
技巧 04	特殊的完全平方数	204	技巧 05	信息查询	237
技巧 05	求 1000 以内的完数	205	技巧 06	计算开机时间	238
技巧 06	三重回文数	206	技巧 07	创建双向链表	240
技巧 07	亲密数	207	技巧 08	创建循环链表	244
技巧 08	自守数	208	技巧 09	创建顺序表并插入元素	246
			技巧 10	向链表中插入节点	248
			技巧 11	从链表中删除节点	250

技巧 12	合并两个链表	253
技巧 13	应用栈实现进制转换	255
技巧 14	括号匹配检测	258
技巧 15	串的模式匹配	261
技巧 16	简单的文本编辑器	262

第 14 章 C 语言开发人员必知必会的

高级开发技巧	267
技巧 01 设置 DOS 系统时间	267
技巧 02 设置 DOS 系统日期	268
技巧 03 函数实现字符匹配	269
技巧 04 查找位置信息	270
技巧 05 建立目录	272
技巧 06 复制当前目录	273

技巧 07	删除目录	274
技巧 08	访问系统 temp 中文件	276
技巧 09	设置组合键	277
技巧 10	获取当前日期与时间	278
技巧 11	设置系统日期	279
技巧 12	获取 BIOS 常规内存 容量	281
技巧 13	获取 CMOS 密码	282
技巧 14	鼠标中断	284
技巧 15	备份/恢复硬盘分区表	286
技巧 16	获取系统配置信息	290
技巧 17	恢复内存文本	292
技巧 18	控制扬声器声音	295

第 5 篇 算法和效率——程序员的得力助手

第 15 章 常用的 30 个算法及应用

算法 01	冒泡排序	298
算法 02	选择排序	300
算法 03	直接插入排序	301
算法 04	归并排序	303
算法 05	希尔排序	305
算法 06	快速排序	307
算法 07	顺序查找	309
算法 08	二分查找	310
算法 09	分块查找	312
算法 10	哈希查找	314
算法 11	斐波那契数列	317
算法 12	角谷猜想	319
算法 13	歌德巴赫猜想	320
算法 14	四方定理	322
算法 15	尼科彻斯定理	323

算法 16	创建单向 链表	325
算法 17	约瑟夫环	328
算法 18	双链表逆序输出	330
算法 19	二叉树的遍历	332
算法 20	用栈设置密码	334
算法 21	链队列	338
算法 22	SHA 加密	342
算法 23	MD5 加密	349
算法 24	RSA 加密	359
算法 25	DES 加密	361
算法 26	RC4 加密	366
算法 27	打渔晒网问题	369
算法 28	水池注水问题	371
算法 29	迪杰斯特拉算法	372
算法 30	百钱买百鸡	374

1

第 1 篇 编程经验与感悟 ——站在前人臂膀上前行

第 1 章 步入编程殿堂

第 2 章 编程高手具有的基本素质和能力

第 3 章 畅谈编程



第1章 步入编程殿堂

历史的发展是无数先辈共同努力的结果，当代的我们可以说是“站在巨人的肩膀上”，用前人积累的成果，不断地创造着下一个辉煌。

软件开发的工作在今天常被视为一种体力活，我们拿着前人的成果可以更轻松地创造更有价值的东西。本人在软件开发行业摸爬数年，不敢说有多深厚的见解，但是也总结了一些软件开发学习的心得，希望与君共勉。

感悟 01 编程我也行

也许很多想要从事编程工作的人都会问，我真的可以胜任这份工作吗？其实这种想法的产生与计算机的发展历程有关。自计算机问世以来，软件开发就成为了令人羡慕的职业。在软件行业刚兴起的时候，程序开发被很多人追捧为高智商、高水平的高端技能，程序员就犹如高深莫测的武林高手般被世人崇拜敬仰，那时候的程序员充满了个人英雄主义色彩。计算机产业被誉为高新技术，计算机从业者被称为 IT 精英，让人觉得可望而不可及。

随着计算机的迅速普及，发展到今天，软件可以说无处不在，软件开发也不再是传说。而个人英雄主义时代也已经过去，当前的程序员，需要有高度的团队合作精神，一个庞大的系统是由众多程序员共同完成的。只要热爱编程并为之努力，编写出自己的程序并不是遥不可及的事情。你可以是一名程序员，你也可以只是出于兴趣，只要你喜欢，就可以进行编程。

有位编程前辈曾经说过：任何人都能编程，那么我们就开始自己的程序开发之旅吧。

感悟 02 如何选择编程语言

要想学习编程，第一件事就是要选择一个开发工具。“工欲善其事，必先利其器”，这是《论语》中的名句，大家在学习编程的时候都希望选择一门优秀的编程语言。现今国内外比较常用的编程语言不下几十种，选择一门适合自己的编程语言是学习编程的第一关，这涉及程序员的未来工作方向，甚至是职业生涯规划，笔者根据数十年的编程经验，并参考相关权威资料总结出以下几点。

● 选择与自己兴趣相近的方向

只有把工作当作一件快乐的事，才能使自己愿意为之奋斗终生，正因为这样，所以才有了中国

的医药学家李时珍在无病的情况下亲自尝试各种药材,才有美国著名的物理学家富兰克林在雷电天气里放风筝等名人轶事。兴趣是指你未来最希望从事的工作方向,根据自己的编程兴趣去选择编程语言,发挥自己最大的优点。接下来笔者列举了在软件开发行业中几个比较热门的设计方向,大家可以根据个人的兴趣来选择适合自己的编程语言。

● 嵌入式开发

这个方向的开发语言可以选择 C 或 C++等语言,它们灵活性好、兼容性强、效率高、能够适应比较底层技术的开发。

● 网络及 Web 应用开发

如今是互联网的天下,这个方向的开发语言比较多,比较流行的有 Java、C#、PHP、Python、Perl 等,其中 Java 是 JSP 开发技术的后台开发语言,C#是如今火热流行的 ASP.NET 开发技术的后台编程语言,Perl 的技术有些陈旧,有被 Python 取代之势。

● 应用软件开发

相对 Web 网络开发,桌面版的应用软件开发技术发展有些缓慢,但在我们的工作和生活中还经常用到,如 QQ、各种视频播放器等,这方面常见的开发语言或开发工具有 VC++、VB、Delphi、C#等。

● 数据库应用开发

数据库应用开发技术是许多编程语言必有的核心开发技术之一,常用的语言包括 Java、C#、VC++、VB、Delphi、PowerBuilder 等。

● 游戏软件开发

游戏开发目前多用 C、C++、Java 等编程语言。

● 尽量选择就业需求量较大的语言

选择编程语言时,应该查看一下近年来就业市场的需求情况,这个比较容易理解,例如,20 世纪 90 年代中后期,C/S 模式的数据库应用软件开发掀起浪潮,当时使用 PB、VB、Delphi 等编程语言开发的软件几乎占据了全部的市场,市场招聘异常火热,但时至今日,随着 Web 时代的到来,PB、VB、Delphi 等编程语言的程序开发人员招聘逐年萎缩。

● 尽量选择国际大公司所研发出的语言

选择编程语言及开发工具,建议选择国际大型软件公司所研发出的语言,因为其技术支持稳定、完善,而且它的战略发展直接引导着整个软件开发界的未来,所以使用它们的开发语言总体来说相对稳定,即使如 Sun 这样的大公司倒下了,但其开发语言 Java 仍经久不衰,更是后继有人(Sun 公司被 Oracle 公司收购),使用微软开发工具的程序员,更能够感受到微软公司长期给其带来的稳

定的技术保障，以及引领软件开发未来的编程思想。

● 选择现今流行且语法规范性强的语言

现今比较流行网络开发和面向对象的技术开发，那么选择如 C++、Java、C# 等面向对象规范性相对较强的开发语言，是一个不错的选择，而 C# 语言可以说是目前众多编程语言中面向对象最完善的一种。

C 语言是广泛使用的教学语言，具有超强的生命力。可以作为编程入门语言来进行学习。深厚的 C 语言知识基础可以为以后的程序学习和开发提供强大的思想准备。另外，我要说的是，不要过分讨论哪个语言好、那个语言不好，或者今天学这个，明天又学那个，这样终究只会一无所长。“十鸟在林，不如一鸟在手”，就算只用 C 语言，用好了，同样可以成为大师级人物。

感悟 03 培养对编程的兴趣

在学习编程和进行软件开发的过程中，总是要面对许多程序代码，因此对于初学者而言，学习编程可以说是一件非常枯燥的事情，为什么说对于初学者来说是枯燥的事情呢？这是由于初学者没有编程经验，只知道照着书本或资料去敲代码，面对诸多的程序代码必然会感觉乏味，而对于那些中高级程序员来说则不一样，他们不但有编程经验，而且编程的知识也已经系统化，能够编写出所要实现功能的程序代码，如编个小游戏，做一个实用的软件等，所以编程对于那些中高级程序员来说是一种乐趣。

说明：之所以对初学者与中高级程序员进行对比，只是为了说明兴趣的重要性，并不是说初学者面对那些枯燥的代码就不能学好编程，因为很多程序员都是从这个过程走过来的。

既然众多的代码令人厌烦，那么初学者应该想办法弥补由于程序代码所带来的烦恼，最好的办法就是在编程的过程中加入一些身边的或者曾经经历过的感兴趣的内容，比如，可以将原来输出的字符串，改为输出一个特殊的符号或图形，原来输出的是一行星号，可以将其改为输出一个由星号组成的三角形、棱形等，久而久之，对编程就会产生兴趣，而不会再觉得编程是一件繁琐的事情了。兴趣是最好的老师，所以一旦对编程产生了兴趣，学习编程也就会觉得轻松了，只要是自己感兴趣的，即使遇到困难也会以解决困难的过程而感到高兴。

培养编程兴趣的另一种好办法，就是多留意身边的一些软件，比如 Windows 提供的扫雷游戏，如图 1.1 所示，通过自制这样的游戏既可以开发智力，又可以提高对编程的兴趣。

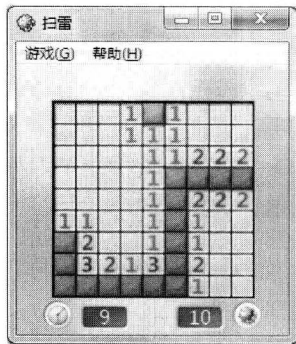


图 1.1 扫雷游戏

再来看一个 Windows Vista 和 Windows 7 自带的 Purple Place 游戏，该游戏由 3 种游戏 Comfy Cakes、Purple Shop 和 Purple Pairs 组成。

其中 Comfy Cakes 游戏将你置于 Purple 厨师的面包店中，你的工作就是完成一批蛋糕的订单。由于面包店的客户非常特殊，因此需要采用正确的形状、口味、夹心、糖衣和点缀组合制作蛋糕，如图 1.2 所示。

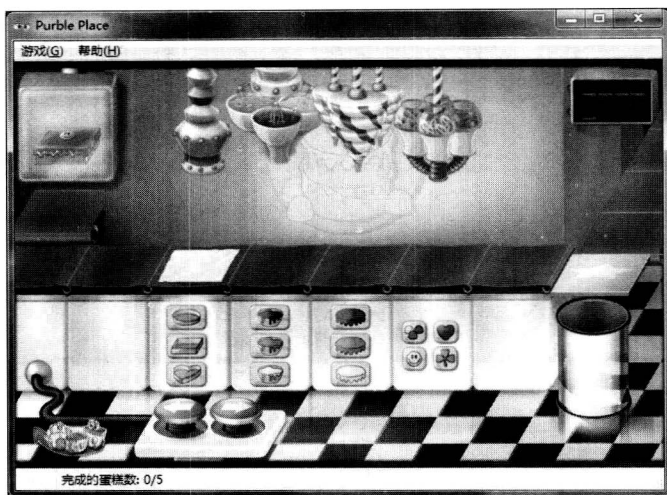


图 1.2 Comfy Cakes 游戏

Purple Shop 游戏用于测试推断能力。游戏目标就是使 Purple 的脸与窗帘后面的神秘 Purple 相匹配。从架子上选择头发、眼睛、鼻子和嘴唇，如图 1.3 所示。

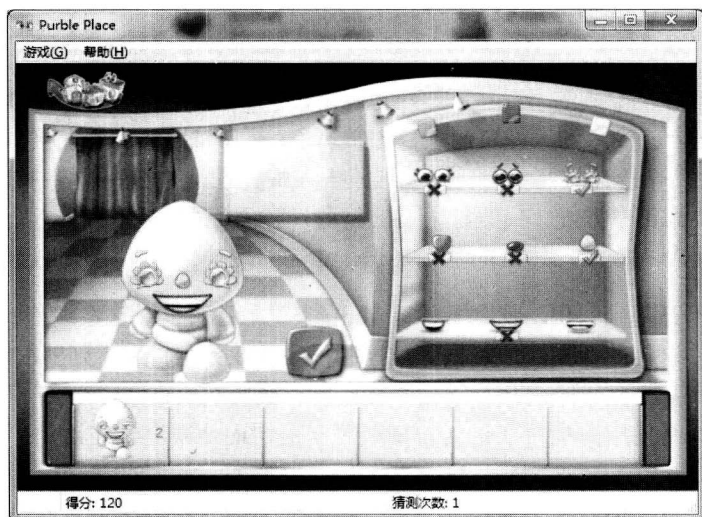


图 1.3 Purple Shop 游戏

Purple Pairs 游戏的目标是在时间用完之前找到所有完全相同的图片。通过翻转一个图块以显示图片，然后尝试找到与其配对的图片。使用“秘技”点可以对整局游戏进行浏览，如图 1.4 所示。

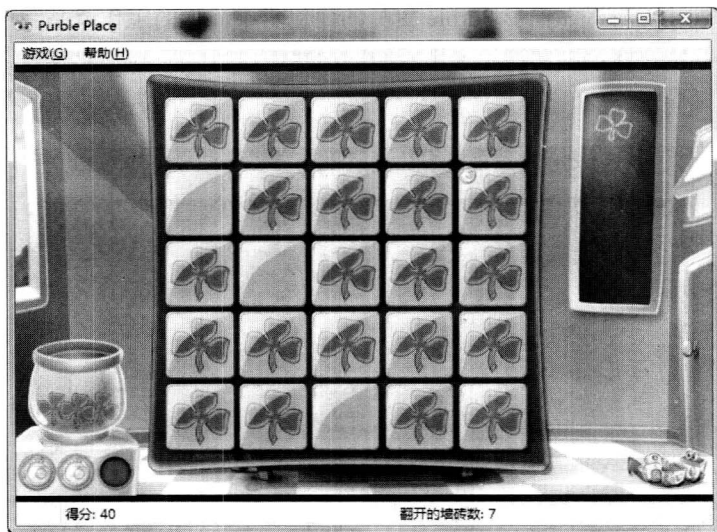


图 1.4 Purble Pairs 游戏

说明：通过身边的这些小游戏和小软件，完全能够使你对编程产生兴趣，因为这些小游戏也好，或是一些实用的软件也罢，都是通过程序语言编写出来的，只要有了兴趣，自然也就能够学好编程了。

感悟 04 学习编程要打好基本功

万丈高楼平地起，空中楼阁不存在，打好基础是你勇攀高峰的基石。有些编程的疯狂爱好者简单看看语法之后，马上就投入到大型实例项目的学习中，然后仿照实例开发实际项目。在这些人中，确实有一部分人通过学习实例学会了编程，但这些人编程过程中遇到的各种困难和障碍只有他们自己心里最清楚。要真正掌握好一门编程语言的精髓，学好其语言基础及其运行机制与原理是十分重要的，只有扎实地掌握了这门语言的基础知识，才能够比较快速地提升自己的编程技能。

可以说编程离不开基础知识，就拿条件语句 `if` 来说吧，可以说它在编程的过程中是无处不在的，也正是由于条件语句的存在，才能很好地控制程序的逻辑，否则就很难对程序进行控制，程序代码就会按顺序从上到下一直执行下去，直到所有代码都执行完，而 `if` 语句又有多种用法，这就要根据需要使用简单的 `if` 语句、复杂的 `if` 语句还是嵌套的 `if` 语句。

注意：基础知识对于每个程序员来说都是非常重要的，如果能熟练掌握并灵活应用基础知识，对于以后的工作和学习都是非都有帮助的，因此必须要学好基础知识。但是，并不是说刚开始学习编程时就能够熟练掌握那些基础知识，可以在学习一段时间后，再回过头来看一下那些基础知识，这样会更容易掌握，理解会更深刻。