

注意：下面的勘误中，红色字体为修改后的文字，提请各位读者注意。

1. 第 6 页，“1.6 语言标准”中的第 3 行，将 1987 年修改为 1978 年。
2. 第 22 页，“2. main () 函数”中的第 1 行，int main (void)后面的分号 (;) 删除。
3. 第 24 页，“5. 声明”的第 10 行，也就是一个变量、函数或其他实体的名称。
4. 第 27 页，图 2.3 中，下划线应该只包含括号中的内容；第 2 段的第 4 行，而不是存储在源代码中的指令。
5. 第 30 页，“2.5.4 打印多个值”的第 4 行，双引号后面的第 1 个变量。
6. 第 34 页，“2.7.3 程序状态”第 2 段的第 4 行，要尽量忠实于代码来模拟。
7. 第 35 页，“2.10 本章小结”第 2 段的第 1 句，声明语句为变量指定变量名，并标识该变量中存储的数据类型；本页倒数第 2 行，即检查程序每执行一步后所有变量的值。
8. 第 37 页，“2.12 编程练习”中第 1 题，把你的名和姓打印在一行……把你的名和姓分别打印在两行……把你的名和姓打印在一行……把示例的内容换成你的名字。
9. 第 40 页，第 1 行，用于把英磅常衡盎司转换为……
10. 第 44 页，“3.4 C 语言基本数据类型”的第 1 句，本节将详细介绍 C 语言的基本属性类型……
11. 第 46 页，“5. 八进制和十六进制”的第 4 句，十六进制数 3 的二进制数是 0011，十六进制数 5 的二进制数是 0101；“6.显示八进制和十六进制”的第 1 句，既可以使用也可以显示不同进制的数；将“回忆一下……程序在执行完毕后不会立即关闭执行窗口”放到一个括号里。
12. 第 47 页，“2.使用多种整数类型的原因”第 3 句，过去的一台运行 Windows 3.x 的机器上。
13. 第 53 页，图 3.5 下面的第 4 行“上面最后一个例子 (printf(“Gramps sez, \”a \” is a backslash.\”\n”);)”
14. 第 56 页，正文的第 2 行和第 4 行应该分别为 printf(“me32 = %” “d” “\n”, me32); printf(“me32 = %d\n”, me32);
15. 第 61 页，“无符号类型”的最后 1 句，相当于 unsigned int (即两者之间添加一个空格)。
16. 第 62 页，程序清单 3.8 中的第 1 行，将 /* typesize.c -- 打印类型大小 */ 中的第一个斜杠删除。
17. 第 63 页，“3.6 参数和陷阱”第 2 行，printf(“Hello,pal.”) (即 Hello,和 pal.之间没有空格)。
18. 第 64 页，程序清单 3.10 中的第 1 行，使用转义序列。
19. 第 75 页，倒数第 8 行，何时使用圆括号取决于运算对象是类型还是特定量。
20. 第 82 页，第 11 行，……格式字符串包含了两个待打印项 number 和 pies 对应的……
21. 第 83 页，表 4.4 中的“l”修饰符的含义介绍中，应该是示例：“%Lf”、“%10.4Le”
22. 第 84 页，表 4.5 中的第 1 行，即，从字段的左侧开始打印该项 (即，应该只保留一个项)；在“0”标记的含义中，添加一行：示例：“%010d”和“%08.3f”。
23. 第 86 页，第 1 段的第 2 行，……字段宽度是容纳待打印数字所需的……；倒数第 4 段中，根据 %x 打印出 1f，根据 %X 打印出 1F
24. 第 87 页，“4.4.4 转换说明的意义”第 2 段，……读者认为原始值被替换成转换后的值。
25. 第 89 页，“参数传递”第 2 行，把变量 n1、n2、n3 和 n4 的值传递给程序 (即，保留一个顿号)。
26. 第 93 页，第 5 行的 2121.45 的字体应该与第 4 行的 42 的字体保持一致；表 4.6 上面的最后一行，对于 double 类型要使用 l 修饰符。
27. 第 94 页，表中的第 3 行，把对应的数值存储为 unsigned short int 类型；把“j”转换说明的示例放到“z”转换说明中；在“j”转换说明的含义中添加：示例：“%jd”、“%ju”。
28. 第 95 页，“3 scanf () 的返回值”上面一段的倒数第 3 行，如果在格式字符串中把空格放到 %c 的前面。
29. 第 98 页，倒数第 2 段，strlen () 函数 (声明在 string.h 头文件中) 可用于……。

30. 第 100 页, “4.8 编程练习”中的第 2 题, 将该题中的“名和姓”统一替换为“名字”; 并执行以下操作; 第 3 题, 将 a、b 项中的“输入”替换为“The input is”, 将“或”替换为“or”, 将末尾的分号换成点 (.)。
31. 第 105 页, 第 8 行, 由于 19.0 不小于 18.5, 所以该条件为假。
32. 第 107 页, 程序清单 5.3 下面的第 1 行, 首先把 68 赋给 jane。
33. 第 111 页, 图 5.3 下面的第 1 行, 如何让加法运算在除法运算之前执行。
34. 第 117 页, 程序清单 5.11 结束后的第 4 行, 而 pre_b 是 b 递增之后的值。
35. 第 118 页, 倒数第 2 行, 而不是 (x*y) ++。
36. 第 129 页, 程序清单 5.15 的第 4 行, //1 小时的秒数。
37. 第 134 页, “5.11 编程练习” 中的第 4 题, 168.7 cm = 5 feet, 6.4 inches
38. 第 143 页, 正文第 2 段, 假设你想跳过输入到达第 1 个既不是空白字符也不是数字的位置
39. 第 148 页, 倒数第 3 行, 高优先级组: < <= > >= (即在<和<=之间有空格, 在>和>=之间有空格)
40. 第 153 页, 第 7 行的“15”与下一行的“28”左对齐。
41. 第 161 页, “小结: do while 语句”中的倒数第 4 行, 在 expression 为假或 0 之前 (注意要用斜体)
42. 第 167 页, 程序清单 6.20 的名字应该是 power.c 程序 (即删除一个 w)
43. 第 170 页, “6.15 复习题” 第 1 题, 后 5 行中使用的是前一行生成的 quack 的值。
44. 第 175 页, 第 10 题的第 3 句话, 用户输入的上限整数等于或小于下限整数为止。
45. 第 178 页, 中间部分的文字中, if 语句指示几岁安及, 如果刚读取的值 (temperature) 小于 0。
46. 第 185 页, 正文第 2 段, 特别要注意的是, 如果 kwh 大于 360; 中间代码之后的第 1 句, 也就是说, 该程序由一个 if else 语句组成 (即, if 和 else 之间要有一个空格)
47. 第 187 页, 正文倒数第 2 段, 倒数第 3 行, 2 和 72、3 和 48、4 和 36。
48. 第 196 页, 代码中第 2 行, 达到单词的末尾。
49. 第 212 页, 复习题的第 4 题, 下列各表达式的值是多少。
50. 第 215 页, 第 2 题的第 2 句话, 每行打印 8 个“字符-ASCII 码”组合; 第 7 题的 a 项中, 10.00 美元/小时。
51. 第 222 页, “8.4 重定向和文件” 的第 2 句话, 输入设备 (我们假设) 是键盘; “8.4.1UNIX、Linux 和 DOS 重定向”的上面一段, 重定向的一个主要问题是它与操作系统有关; 苹果 OS X 运行在 UNIX 上, 故可用 Terminal 应用程序来使用 UNIX 命令行模式。
52. 第 224 页, “3.组合重定向” 中的第 2、4、6 行中, 应该是分别是 ./echo_eof < mywords > savewords、./echo_eof > savewords < mywords、./echo_eof < mywords > mywords....; 第 13 行应该是 ./echo_eof<words; 第 16、17、18、19 行的多买中, 均在最前面添加 ./
53. 第 225 页, “小结: 如何重定向输入和输出”中的 4 行代码中, 均在前面添加 ./
54. 第 227 页, 正文中间, 该程序还是会把 f 视为 n (即这里将“被”删除)。
55. 第 245 页, 倒数第 6 行中, 程序中 starbar() 和 main() 的定义形式相同。
56. 第 247 页, “9.1.3 函数参数”中第 2 段最后 1 行, 因此, 可以调用 show_n_char(' ', 12) (即两个单引号之间是一个空格)
57. 第 260 页, 第 19 行, 因此, n 乘以 n-1 的阶乘就得到 n 的阶乘。
58. 第 268 页, 程序清单 9.13 上面的一行, 在 interchange() 中使用 u 和 v。
59. 第 272 页, 倒数第 7 行, 让 interchange() 访问这两个变量。
60. 第 273 页, “变量: 名称、地址和值”中第 3 段第 2 行, 使用变量名即可获得变量的数值。
61. 第 276 页, “9.11 编程练习”第 6 题, 把最小值放入第一个变量; 第 10 题, 编写一个 to_base_n()

函数接受两个参数，且第 2 个参数在 2~10 范围内，然后以第 2 个参数……。

62. 第 285 页，第 11 行，`float rain[5][12]`；（即 `float` 和 `rain` 之间有一个空格）；图 10.1 上面的一句话，则使用 `rain[1][2]`；顺便将括号以及括号中的文字删除。
63. 第 289 页，图 10.3 上面一段的第 2 行，这意味着加 1 后的地址是下一个元素的地址（即，将“把”删除）
64. 第 290 页，第 1 行，`dates + 2 == &dates[2]`
65. 第 295 页，第 3 行，至于 C 语言，`ar[i]`和`*(ar+i)`这两个表达式都是等价的。
66. 第 305 页，正文倒数第 3 段，第 2 行，指向一个内含 3 个 `int` 类型元素的数组；`pa` 指向一个内含 3 个 `int` 类型元素的数组。
67. 第 307 页，程序清单 10.17 上面的一段，这样的变量稍后能以同样的方式用作 `junk`。
68. 第 316 页，第 6 题，在 `a`、`b`、`c` 这 3 项的后面添加“的地址”
69. 第 322 页，上面第 2 行代码，`I am a symbolic string constant.`（即，将 `an` 换成 `a`，将 `old-fashioned` 删除）
70. 第 326 页，“5.字符串数组”上面的一句，如果打算修改字符串，就不要用指针指向字符串字面量；“5.字符串数组”下面的一句，创建一个字符串通常很方便（即将“如果”删除，将“会”换成通常）。
71. 第 332 页，最后一段的第 1 句，`fgets()`函数返回指向 `char` 的指针。
72. 第 336 页，图 11.3 中“输入语句”栏，将这三个均修改为 `scanf("%5s",name);`;
73. 第 348 页，正文倒数第 2 段，并编写一个函数把输入的内容都转换成大写
74. 第 356 页，正文最后一段的第 1 句，程序清单 11.28 中的程序用 `sprintf()`把 3 个项
75. 第 358 页，第一行，该函数返回指向 `s` 字符串首次出现的 `c` 字符的指针
76. 第 366 页，正文第 3 段，如果字符串仅以整数开头，`atoi()`函数也能处理
77. 第 370 页，第 5 题的 `e` 项，如果用 `*pc--`替换`*--pc`，会打印什么
78. 第 371 页，“11.13 编程练习”第 1 题，从输入中获取 `n` 个字符（即将“下”删除）
79. 第 372 页，第 8 题，如果第 2 个字符串包含在第 1 个字符串中；第 10 题，该程序应该应用该函数读取每个输入的字符串，并显示处理后的结果；第 11 题，编写一个程序
80. 第 374 页，第 2 段，内含这些字符值的字符串字面量就是一个对象，由于字符串字面量中的每个字符都能被……
81. 第 382 页，“12.1.7 外部链接的静态变量”第 3 行，放在所有函数的外面（即将其中一个“在”删除）
82. 第 383 页，正文最后一段第 2 行，外部变量 `Hocus` 对 `main()`和 `magic()`均不可见
83. 第 391 页，正文第 1 段，在这个文件中不要求写出该函数定义。
84. 第 394 页，正文最后一段，都要包含 `diceroll.h` 头文件。
85. 第 396 页，正文第 2 段，对于读取骰子的点数
86. 第 406 页，正文第 3 段第 1 行，但是 `memmove()`没有这样的要求。
87. 第 407 页，将正文第 2 段删除。
88. 第 422 页，中间的代码中，第 1 行和第 11 行应该是 `./addaword`
89. 第 423 页，第 4 行，如果未遇到 EOF 则返回之前传给它的第一个参数地址。
90. 第 425 页，正文第 2 段，`ftell()`函数返回类型是 `long`，它返回的是参数指向文件的当前位置距文件开始处的字节数。
91. 第 426 页，“13.5.4 `fgetpos()`和 `fsetpos()`函数”中的正文第 3 段，该值描述了文件中的当前位置距文件开头的字节数；第 5 段，来设置文件指针指向偏移值后指定的位置。
92. 第 427 页，第 2 段第 2 行，文件中的缓冲大小数据块就被拷贝到缓冲区中；图 13.2 的图题是 `ungetc()`

函数。

93. 第 444 页，程序清单 14.2 中，建立 book 模板
94. 第 450 页，倒数第 5 行，如果 barney 是一个 guy 类型的结构变量，可以这样写；倒数第 3 行，结构变量名并不是结构的地址，因此要在结构变量名前面加上&运算符。
95. 第 451 页，“14.6.2 用指针访问成员”第 3 段正文，换句话说，指向结构的指针后面的->运算符和结构变量名后面的.运算符工作方式相同；下一段，但是 him->income 是该指针所指向结构的一个成员（即，将字母 e 去掉）。
96. 第 453 页，正文第 2 段，使得指针 money 指向变量 stan；下一段的第 2 行，结构变量名不是其地址的别名。
97. 第 470 页的代码中，将\$ booksave 修改为\$./booksave
98. 第 471 页第 5 行，rewind()函数确保文件指针位于文件开始处。
99. 第 474 页，“总结：结构和联合运算符”中，该运算符与结构变量或联合变量名一起使用，指定结构变量或联合变量的一个成员。如果 name 是一个结构变量的名称……下面标识了该结构变量的这个成员；该运算符和指向结构或联合的指针一起使用，标识结构变量或联合变量的一个成员。
100. 第 530 页，正文第 2 行和第 4 行的代码，开头应该是 printf（即当前少了字母 f）
101. 第 533 页，倒数第 8 行，将第 1 句中的 getc(stdin)和 getchar()互换。
102. 第 548 页，倒数第 4 行，即在原函数名后加上 f 或 l 后缀。
103. 第 553 页，第 1 行，qsort()原型中的第 4 个参数确定了比较函数的形式
104. 第 574 页，将第 5 行代码 current = head;删除；将第 7 行代码移动到原来第 5 行的位置；
105. 第 579 页，正文第 4 段，而不是一个指向节点的指针或一个结构；最后一行，防止多次包含一个头文件。
106. 第 596 页，程序清单 17.7 上面这一行，包括 EnQueue()函数中用到的 CopyToNode()函数。
107. 第 611 页，第 12 行，如果在树中找到指定项，该返回返回 true。
108. 第 614 页，第 2 行，AddItem()、InTree()和 DeletetItem()。
109. 第 631 页，第 4 题，重写 mall.c（程序清单 17.9）
110. 第 633 页，“A.2 第 2 章复习题答案”中，第 4 题，第 6 行，把:=改成=，赋值用=，
111. 第 636 页，第 10 题，第 4 行：count? \n;（即将 ry 删除）；“A.4 第 4 章复习题答案”，第 1 题的第 2 句话，下一条 scanf()语句在输入缓冲区查找重量时
112. 第 646 页，最后一行，sacnf(“%s”, &score);
113. 第 650 页，第 8 题的 d 项，[10] = 101, 101, 101, [3] = 101];
114. 第 669 页，“1.逻辑表达式”中的第 2 行，两个表达式中至少有一个为真时，expression 1 || expression 2 的值就为真
115. 第 679 页，表 B.5.1 中，如果 exprs 为 1（或真），宏什么也不做。