

<<轻松学C++>>

图书基本信息

书名：<<轻松学C++>>

13位ISBN编号：9787121198090

10位ISBN编号：7121198096

出版时间：2013-5

出版时间：电子工业出版社

作者：邹国华

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<轻松学C++>>

内容概要

《轻松学C++(图解版)》由浅入深,全面、系统地介绍了C++编程语言。

《轻松学C++(图解版)》最大的特色就是提供了大量的插图,一改过去编程书籍枯燥乏味的文字讲解,利用各种说明插图和运行结果示意图,生动形象地再现了C++语言各项内容,使读者能够轻松地掌握学习内容。

另外,作者专门为每一章编写了一些习题,以便读者对该章的学习水平进行检测。

《轻松学C++(图解版)》还录制了大量的配套教学视频,这些视频和书中的实例源代码一起收录于《轻松学C++(图解版)》的配书光盘中。

<<轻松学C++>>

书籍目录

第1篇 C++概述篇

第1章 C++基础 2

1.1 C++概述 2

1.1.1 C++语言发展历程 2

1.1.2 C++语言特点 2

1.1.3 C++程序开发基本过程 3

1.2 C++开发环境的搭建 4

1.2.1 Visual C++ 6.0安装 4

1.2.2 Visual C++ 6.0开发环境简介 6

1.3 第一个C++程序 6

1.3.1 创建源程序 6

1.3.2 编译链接 9

1.3.3 调试运行 10

1.3.4 典型C++程序的执行过程 12

1.3.5 使用C++解决问题的流程 12

1.4 C++程序的结构 14

1.4.1 注释 14

1.4.2 编译预处理和旧标准 14

1.4.3 程序主体 15

1.5 小结 15

1.6 习题 15

第2篇 C++基础语法篇

第2章 数据的表示 18

2.1 数 18

2.1.1 二进制 18

2.1.2 八进制 19

2.1.3 十六进制 20

2.2 数据的描述 21

2.2.1 C++的构词方式——标识符 21

2.2.2 特殊的标识符——关键字 22

2.2.3 整数类型 23

2.2.4 浮点类型 24

2.2.5 字符类型 25

2.2.6 布尔类型 27

2.3 变量 27

2.3.1 变量的声明和定义 28

2.3.2 不变的变量——常量 28

2.4 小结 29

2.5 习题 29

第3章 程序的基本单位——语句 33

3.1 语句的构成 33

3.1.1 表达式语句 33

3.1.2 输入/输出语句 33

3.2 运算符 35

3.2.1 运算符概述 35

<<轻松学C++>>

- 3.2.2 运算符的分类 35
- 3.3 赋值运算符 36
 - 3.3.1 赋值运算符——“=” 36
 - 3.3.2 数据类型转换——隐式转换 37
 - 3.3.3 显式转换 37
- 3.4 其他常用运算符 39
 - 3.4.1 算术运算符 39
 - 3.4.2 自增自减运算符 40
 - 3.4.3 位运算符 41
 - 3.4.4 复合赋值运算符 43
 - 3.4.5 逗号运算符 43
 - 3.4.6 sizeof运算符 44
 - 3.4.7 逻辑运算符 45
 - 3.4.8 运算符的优先级和结合性 45
- 3.5 语句块 48
 - 3.5.1 语句块的构成 48
 - 3.5.2 作用域——变量的作用范围 48
- 3.6 小结 49
- 3.7 习题 49
- 第4章 程序控制结构 56
 - 4.1 语句块的执行方式——顺序结构 56
 - 4.2 条件的表达 56
 - 4.2.1 单一条件的表达——关系运算符 56
 - 4.2.2 多条件的表达——逻辑运算符 57
 - 4.3 选择结构 58
 - 4.3.1 最简单的选择——条件运算符 58
 - 4.3.2 单分支条件语句——if语句 59
 - 4.3.3 双分支条件语句——if...else语句 61
 - 4.3.4 多重if...else语句 62
 - 4.3.5 情况语句——switch语句 64
 - 4.3.6 应用实例 65
 - 4.4 循环结构 66
 - 4.4.1 for语句 67
 - 4.4.2 while语句 68
 - 4.4.3 do...while语句 69
 - 4.4.4 多重循环 70
 - 4.4.5 应用举例 71
 - 4.5 意外情况的表达——转向语句 72
 - 4.5.1 跳出语句——break语句 72
 - 4.5.2 继续语句——continue语句 72
 - 4.5.3 转向语句——goto 73
 - 4.6 综合实例 74
 - 4.7 小结 76
 - 4.8 习题 76
- 第5章 数组 81
 - 5.1 数组概述 81
 - 5.2 数组的来源 82

<<轻松学C++>>

- 5.3 一维数组 83
 - 5.3.1 一维数组的声明和定义 83
 - 5.3.2 一维数组的引用 84
- 5.4 二维数组 86
 - 5.4.1 二维数组的声明和定义 86
 - 5.4.2 二维数组的初始化 86
 - 5.4.3 二维数组的引用 88
 - 5.4.4 多维数组在内存中如何排列元素 89
- 5.5 字符串 90
 - 5.5.1 字符数组 90
 - 5.5.2 字符串的存储形式 91
 - 5.5.3 字符数组与字符串的区别 91
 - 5.5.4 字符串处理函数 92
- 5.6 综合实例——杨辉三角 94
- 5.7 小结 97
- 5.8 习题 97
- 第6章 函数 106
 - 6.1 函数概述 106
 - 6.1.1 函数的基本概念 106
 - 6.1.2 函数的分类 107
 - 6.2 函数的组成 107
 - 6.2.1 函数头 108
 - 6.2.2 函数体 108
 - 6.3 函数的声明和定义 110
 - 6.3.1 函数原型——函数的声明 110
 - 6.3.2 函数实现——函数的定义 111
 - 6.4 函数参数传递 112
 - 6.4.1 函数的形参和实参 112
 - 6.4.2 值传递 113
 - 6.5 函数的调用 115
 - 6.5.1 函数的调用过程 115
 - 6.5.2 无参函数的调用 115
 - 6.5.3 带参函数的调用 116
 - 6.5.4 默认形参值的调用 117
 - 6.5.5 嵌套调用 120
 - 6.5.6 数组作为函数参数 121
 - 6.6 递归函数 123
 - 6.6.1 直接递归 124
 - 6.6.2 间接递归 125
 - 6.7 main()函数 127
 - 6.7.1 不带参数的main()函数 127
 - 6.7.2 带参数的main()函数 127
 - 6.8 函数的综合应用 128
 - 6.9 小结 128
 - 6.10 习题 128
- 第7章 指针与引用 136
 - 7.1 指针概述 136

<<轻松学C++>>

- 7.1.1 指针的基本概念 136
- 7.1.2 定义指针变量 137
- 7.1.3 初始化指针 138
- 7.2 指针的访问 139
 - 7.2.1 指针的值 140
 - 7.2.2 访问指针数据 140
 - 7.2.3 小结指针p 141
- 7.3 指针的算术运算 142
 - 7.3.1 指针与整数的加减运算 142
 - 7.3.2 指针加减1运算 143
 - 7.3.3 指针的相减运算 144
- 7.4 特殊指针 144
 - 7.4.1 数组指针 144
 - 7.4.2 指向函数的指针——函数指针 145
 - 7.4.3 指针数组 146
 - 7.4.4 二级指针——指针的指针 147
 - 7.4.5 多级指针——二级以上的指针 148
- 7.5 指针的应用 149
 - 7.5.1 指向一维数组的指针 149
 - 7.5.2 指向二维数组的指针 150
 - 7.5.3 指向字符串的指针 152
 - 7.5.4 指针作为函数参数 153
 - 7.5.5 指针作为函数的返回值——指针函数 155
 - 7.5.6 动态内存分配 156
- 7.6 引用 158
 - 7.6.1 引用的应用 158
 - 7.6.2 引用与指针 159
 - 7.6.3 引用作为函数参数 160
- 7.7 小结 161
- 7.8 习题 162
- 第8章 复合数据类型 169
 - 8.1 结构体 169
 - 8.1.1 结构体概述 169
 - 8.1.2 定义结构体类型 170
 - 8.1.3 声明结构体变量 172
 - 8.2 结构体的应用 173
 - 8.2.1 初始化结构体变量 173
 - 8.2.2 引用结构体变量成员 175
 - 8.2.3 结构体指针 175
 - 8.2.4 结构体数组 176
 - 8.2.5 结构体和数组的比较 178
 - 8.3 联合 178
 - 8.3.1 定义联合类型 178
 - 8.3.2 声明联合变量 179
 - 8.3.3 引用联合类型成员 179
 - 8.4 枚举 180
 - 8.4.1 定义枚举类型 180

<<轻松学C++>>

- 8.4.2 声明枚举变量 182
- 8.4.3 引用枚举变量成员 182
- 8.5 用户自定义数据类型 183
- 8.6 综合实例 184
- 8.7 小结 187
- 8.8 习题 187
- 第3篇 C++面向对象篇
- 第9章 类和对象 194
- 9.1 类和对象概述 194
- 9.2 类和对象的基础语法 195
- 9.2.1 类的声明 195
- 9.2.2 实例化对象 196
- 9.3 类的属性——数据成员 196
- 9.3.1 类的属性的定义 196
- 9.3.2 类的数据成员的特例——静态数据成员 196
- 9.4 类的方法——成员函数 197
- 9.4.1 类的方法的定义 197
- 9.4.2 静态成员函数 198
- 9.4.3 成员函数的类别 (const的另一种用法) 198
- 9.5 特殊的成员函数——构造函数和析构函数 198
- 9.5.1 构造函数的概念 198
- 9.5.2 构造函数的声明和定义 200
- 9.5.3 构造函数的调用 201
- 9.5.4 不带参数的构造函数 202
- 9.5.5 带有默认参数的构造函数 202
- 9.5.6 构造函数的重载 203
- 9.5.7 特殊的构造函数——复制构造函数 205
- 9.5.8 析构函数 207
- 9.5.9 类和函数的联系 208
- 9.5.10 this指针 209
- 9.6 小结 209
- 9.7 习题 209
- 第10章 继承与派生 217
- 10.1 继承与派生的基础语法 217
- 10.1.1 继承与派生概述 217
- 10.1.2 声明派生类 218
- 10.2 成员的访问 219
- 10.2.1 类的成员的访问说明符 219
- 10.2.2 类的成员的访问权限 219
- 10.3 继承的访问控制 219
- 10.3.1 私有继承 220
- 10.3.2 公有继承 221
- 10.3.3 保护继承 223
- 10.3.4 特殊方法的继承——派生类的构造函数和析构函数 225
- 10.4 多重继承 227
- 10.4.1 声明多重继承 227
- 10.4.2 二义性问题 227

<<轻松学C++>>

- 10.4.3 多重继承的构造函数和析构函数 231
- 10.5 虚基类 233
 - 10.5.1 声明虚基类 233
 - 10.5.2 虚基类的构造函数和初始化 234
- 10.6 友元 235
 - 10.6.1 友元的引入 235
 - 10.6.2 友元函数 235
 - 10.6.3 友元成员 237
 - 10.6.4 友元类 239
- 10.7 综合实例 240
- 10.8 小结 244
- 10.9 习题 244
- 第11章 多态 251
 - 11.1 多态概述 251
 - 11.1.1 什么是多态 251
 - 11.1.2 多态的引入 252
 - 11.1.3 联编 253
 - 11.2 函数重载 254
 - 11.3 虚函数 255
 - 11.3.1 定义虚函数 256
 - 11.3.2 多级继承和虚函数 258
 - 11.4 纯虚函数与抽象类 259
 - 11.4.1 纯虚函数 259
 - 11.4.2 抽象类 261
 - 11.5 综合实例 263
 - 11.6 小结 264
 - 11.7 习题 264
- 第12章 运算符重载 269
 - 12.1 运算符重载概述 269
 - 12.1.1 什么是运算符重载 269
 - 12.1.2 运算符重载的特点 270
 - 12.2 运算符重载形式 271
 - 12.2.1 运算符重载为类的成员函数 272
 - 12.2.2 运算符重载为类的友元函数 275
 - 12.2.3 运算符成员函数与友元运算符函数的比较 278
 - 12.3 特殊运算符重载 278
 - 12.3.1 “++”和“--”重载 278
 - 12.3.2 赋值运算符“=”重载 280
 - 12.3.3 下标运算符“[]”重载 281
 - 12.4 类类型转换 283
 - 12.5 小结 284
 - 12.6 习题 284
- 第13章 类模板 290
 - 13.1 什么是类模板 290
 - 13.2 定义类模板 292
 - 13.2.1 语法 292
 - 13.2.2 非类型参数 294

<<轻松学C++>>

- 13.2.3 模板参数的默认实参 294
- 13.3 生成类模板的实例 295
 - 13.3.1 类型参数的模板实例化 295
 - 13.3.2 非类型参数的模板实例化 296
 - 13.3.3 类模板示例 296
- 13.4 类模板的静态成员 298
- 13.5 类模板的友元 300
 - 13.5.1 非模板的友元类和友元函数 301
 - 13.5.2 与模板参数不绑定的友元类和友元函数模板 301
 - 13.5.3 与模板参数绑定的友元类和友元函数模板 302
- 13.6 类模板的特化 303
 - 13.6.1 类模板的全特化 303
 - 13.6.2 类模板的偏特化 304
 - 13.6.3 类模板的匹配规则 304
- 13.7 小结 305
- 13.8 习题 305
- 第4篇 C++应用技术篇
- 第14章 输入/输出流 308
 - 14.1 输入/输出流的引入 308
 - 14.1.1 C语言中的输入/输出缺陷 308
 - 14.1.2 输入/输出流简介 309
 - 14.1.3 输入/输出流类层次 310
 - 14.2 标准输入/输出流 311
 - 14.2.1 标准输出流对象 312
 - 14.2.2 标准输入流对象 312
 - 14.3 输入/输出流成员函数 313
 - 14.3.1 get()函数 313
 - 14.3.2 getline()函数 314
 - 14.3.3 put()函数 315
 - 14.3.4 read()和write()函数 315
 - 14.3.5 其他成员函数 316
 - 14.4 输入/输出格式控制 317
 - 14.4.1 用ios类的成员函数进行格式控制 317
 - 14.4.2 使用格式控制符进行格式控制 320
 - 14.5 用户自定义数据类型的输入/输出 322
 - 14.5.1 重载输出运算符“ ” 323
 - 14.6 命名空间 325
 - 14.6.1 命名空间概述 325
 - 14.6.2 定义命名空间 325
 - 14.6.3 使用命名空间 326
 - 14.7 小结 327
 - 14.8 习题 327
- 第15章 预处理和宏 331
 - 15.1 预处理概述 331
 - 15.2 宏 331
 - 15.2.1 宏展开 332
 - 15.2.2 替代常量 332

<<轻松学C++>>

- 15.2.3 替代运算符 334
- 15.3 带参数的宏 334
 - 15.3.1 定义带参数的宏 335
 - 15.3.2 注意宏展开的结果 336
 - 15.3.3 带参数的宏与函数的比较 338
- 15.4 条件编译 338
 - 15.4.1 宏指令 338
 - 15.4.2 使用条件编译 340
- 15.5 文件包含和头文件卫士 342
 - 15.5.1 包含文件指令 342
 - 15.5.2 搜索头文件 343
 - 15.5.3 头文件卫士 344
- 15.6 预定义的宏 346
- 15.7 小结 347
- 15.8 习题 347
- 第16章 标准模板库 349
 - 16.1 标准模板库概述 349
 - 16.1.1 C++标准库 349
 - 16.1.2 STL的形成 350
 - 16.1.3 STL的组成 350
 - 16.1.4 STL的引入 351
 - 16.2 算法 353
 - 16.3 容器 354
 - 16.3.1 容器概述 355
 - 16.3.2 向量 355
 - 16.3.3 列表 356
 - 16.3.4 集合 358
 - 16.3.5 双端队列 358
 - 16.3.6 栈 360
 - 16.3.7 映射和多重映射 361
 - 16.4 迭代器 361
 - 16.5 小结 363
 - 16.6 习题 363
- 第17章 程序调试与异常处理 365
 - 17.1 程序错误 365
 - 17.1.1 编译错误 365
 - 17.1.2 逻辑错误 366
 - 17.1.3 运行错误 367
 - 17.1.4 程序调试 368
 - 17.2 异常处理 369
 - 17.2.1 基本思想 369
 - 17.2.2 抛出异常 370
 - 17.2.3 捕获异常 371
 - 17.2.4 自定义异常对象 373
 - 17.3 异常处理实例 374
 - 17.4 小结 375
 - 17.5 习题 375

| | |
|-----------------|-----|
| 第18章 文件 | 378 |
| 18.1 文件概述 | 378 |
| 18.1.1 操作文件的过程 | 379 |
| 18.1.2 处理文件流的类 | 381 |
| 18.2 文件的打开与关闭 | 384 |
| 18.2.1 打开文件 | 384 |
| 18.2.2 关闭文件 | 385 |
| 18.3 文件的顺序读/写 | 386 |
| 18.3.1 读/写文本文件 | 386 |
| 18.3.2 读/写二进制文件 | 387 |
| 18.4 文件的随机读/写 | 391 |
| 18.5 小结 | 394 |
| 18.6 习题 | 394 |

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>