


21世纪重点大学规划教材

黄品梅 编著

C++程序设计教程

——化难为易地学习C++



 机械工业出版社
CHINA MACHINE PRESS


配电子教案

21 世纪重点大学规划教材

C++程序设计教程

——化难为易地学习C++

黄品梅 编著

机械工业出版社

本书的目的在于解决高级算法语言 C++难学的问题,让读者能学好 C++,用好 C++,打好计算机软件基础。围绕着“化难为易”这个核心,作者结合自己多年的教学实践和教研科研经验,以全新的教学方法、全新的教材结构、全新的教学语言编写了此教材,采用“寓德于教(学),以德促教(学),人定胜机,机随人行”的十八方针及跟踪追击、人脑运行等教学方法,化难为易地介绍了 C++的各项知识,以及学习和运用 C++知识的方法、经验和技巧。作者运用上述方法,设计并调试成功了本书所有例题和例句的程序,在教学中也取得了良好的效果。

本书既可作为高校计算机及相关专业“C++程序设计”课程的教材,也可供读者自学 C++使用。

本书配套电子教案,需要的教师可登录机工教材服务网(www.cmpedu.com)进行注册,待审核通过后即可免费下载,也可直接联系编辑获取(QQ: 241151483,电话 010-88379753)。

图书在版编目(CIP)数据

C++程序设计教程:化难为易地学习 C++ / 黄品梅编著. —北京:机械工业出版社, 2011.3

(21世纪重点大学规划教材)

ISBN 978-7-111-33365-4

I. ①C… II. ①黄… III. ①C语言-程序设计-高等学校-教材
IV. ①TP312

中国版本图书馆 CIP 数据核字(2011)第 020283 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑:陈皓

责任印制:杨曦

北京双青印刷厂印刷

2011年3月第1版·第1次印刷

184mm×260mm·20.5印张·507千字

0001—3000册

标准书号:ISBN 978-7-111-33365-4

定价:35.00元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

社服务中心:(010) 88361066

门户网:<http://www.cmpbook.com>

销售一部:(010) 68326294

教材网:<http://www.cmpedu.com>

销售二部:(010) 88379649

读者服务部:(010) 68993821

封面无防伪标均为盗版

前 言

C++语言是一种令人向往的计算机软件语言，它的功能十分强大，它的应用无所不在，从计算机的操作系统、大型应用软件到日常生活中的电饭锅、洗衣机、空调等家用电器的控制软件、应用软件等有很多是用 C++语言编写的。另外，利用 C++语言的编程思路、学习方法和技巧，还很容易学习其他程序设计语言。所以很多本专科院校都将 C++语言作为计算机基础课程来开设。

读者可能会发现，怎么这本书这么怪，将例题、习题和思考题安排在知识点之前，与其他书的编排结构完全不同，之所以这样怪，是为了能使读者化难为易地学习 C++，下面具体介绍实现这一目的的方法及通过这些方法获得的效果，这些方法凝聚了作者多年来的教学实践和教研科研经验，获得的效果则是方法的有效验证和胜利果实。

1. 全新的教学方法

(1) 寓德于教（学），以德促教（学）

以道德品质为引领，树立为祖国、为人民多做贡献的思想和目标，不畏艰险，认真施教，勤学苦练。

(2) 人定胜机，机随人行

牢固树立“人脑必定胜于电脑”、“我一定能成为 C++高手”的坚定信念。运用“人是怎么做的，就让计算机跟着怎么做”的方法。

(3) 执行三大设计步骤

1) 用心分析人想通，机随人行写提纲。

2) 对号入座写文章，头脑运行改文章。

3) C++帮助且裁判，人定胜机凯歌扬。

每道题目均遵循此步骤编写。

(4) 纸上谈兵，机上打仗

编写提纲、流程图和程序均在纸上谈兵，输入和调试程序时才进入机上打仗。

(5) 跟踪追击，人脑运行

在纸上谈兵和机上打仗的过程中，均要结合此方法进行。

2. 全新的教材结构

让读者在全新的教材结构中达到如下效果。

(1) 创新习题，利于提高

例题、习题、思考题与以往 C++书籍的题目基本不同，注重联系生活实际，注重编程设计，更有利于读者掌握和运用 C++知识。

(2) 亲临其境，易学易懂

每章均通过对实例的详细讲解引出知识点和学习方法，并给出运行结果，有些例题还附上了调试记录，让读者兴趣盎然，易学易懂。

(3) 亲身体验，学用结合

每一章学完了第 1 节的例题后，马上进入第 2 节亲身体验阶段——独立编程应用。应用中遇到问题时，再带着问题参考第 1 节或第 3 节的知识，做到学用结合。

(4) 有的放矢，边用边学

本书将 C++ 的各项知识点和难点安排在每一章的第 3 节，目的是与第 2 节的题目配合起来，让读者有的放矢地学习 C++ 知识，即用一点，学一点，再用一点，再学一点，而不是一次性地学完某部分 C++ 知识再去做题。当读者能够编写第 2 节习题的程序时，就已经将该部分的 C++ 知识理解消化了。

(5) 归纳总结，巩固提高

有归纳思路才会清晰，有总结水平才会提高，建议读者在学习、实践之后进行总结评分。同时将 C++ 的知识亮点在每章的第 4 节中展示出来，供读者归纳总结，巩固提高。

3. 全新的教学语言

(1) 通俗易懂，生动活泼

运用通俗易懂、生动活泼的教学语言和 C++ 读者共同分享学习的快乐。例如，人定胜机、机随人行、顺瓜摸根、游览 C++ 乐园、分家原理、饭菜打包，一个萝卜一个坑、和 C++ 做朋友、起名字等说法。

(2) 画龙点睛，变难为易

运用浅显精练的语言道出 C++ 的知识点和亮点，使读者学习起来更容易。

4. 良好的教学效果

本书通过以上方法，揭开了 C++ 语言的神秘面纱，让读者化难为易，学以致用，自由欢快地遨游在 C++ 乐园中。作者在教学中通过以上方法取得了良好的效果。例如，很多学生认为这种 C++ 的教学方法好，易学、易懂，且经过一个学期 44 个学时理论课、12 个学时实验课的学习，就可以编写出较大型的 C++ 软件，如自动电镀生产线的简单生产工序操作控制软件、简单的图书馆借书还书管理软件、考勤打卡机管理软件、课程表软件等。许多学生还主动报考全国计算机等级考试二级，有的学生还考到三级、四级，他们满怀信心地说：“将继续将 C++ 自学下去”。还有部分学生在科研活动中多次获奖，用 C++ 语言编写出了一些较大型的组态软件，如虚拟示波器检测平台等，作者也多次获得校级教学优秀奖。

本书的阅读方法为：先学习每一章第 1 节的例题，然后进行第 2 节的勤奋实践，在实践中遇到问题时，再阅读第 3 节的知识，没用到的知识可先不看，留待之后学习，最后阅读第 4 节的内容，对该章知识进行归纳总结。

读者要注意的是：尽可能不要上网找程序，尽量自己编写程序。因为程序中要解决的问题基本都是生活中的问题，编写程序只不过是将从人平时做的事情转换成用 C++ 语言编写的程序，让计算机代替人去做事情而已，只要坚持独立自主、自力更生地编写程序，入了门后，就什么都不难了。

本书的完稿，首先要感谢本书参考文献中的所有作者！另外，还要向所有教育、培养、关心、支持和帮助过作者的人致以诚挚的感谢！

由于作者水平有限，书中不妥之处在所难免，恳请读者批评指正。

作者

目 录

前言

第 1 章 准备好学习 C++	1
1.1 C++是一种接近生活的高级算法语言	1
1.1.1 C++的特性	1
1.1.2 C++的优势	1
1.1.3 学习 C++的必要性和重要性	2
1.2 C++的难处	2
1.2.1 C++本身的难处	2
1.2.2 人为造成的学习 C++的难处	2
1.3 化难为易地学习 C++之精髓	4
1.3.1 道德品质好, 成才的基本条件	4
1.3.2 心有灵“C”一点通	4
1.3.3 人定胜机, 机随人行	5
1.4 思考题	6
第 2 章 一份令人欣喜且信心十足的 C++处女作及 C++基础知识 (一)	7
2.1 初步认识 C++的一个简例	7
2.2 勤奋实践, 总结评分	10
2.2.1 习题	10
2.2.2 思考题	11
2.2.3 总结评分, 继续提高	11
2.3 C++基础知识 (一)	11
2.3.1 一般的 C++程序设计开发流程	12
2.3.2 C++程序的流程图	14
2.3.3 C++的文章——一系列的 C++语句	15
2.3.4 输出语句 cout	15
2.3.5 Visual C++的集成开发环境	16
2.3.6 Visual C++ 6.0 的基本操作	22
2.4 C++基础知识 (一) 的亮点	40
第 3 章 按部就班的事件处理——顺序结构 程序及 C++基础知识 (二)	42
3.1 例题详解	42
3.1.1 显示自动控制煮饭情况	42
3.1.2 计算斜三角形的边长	44
3.2 勤奋实践, 总结评分	47
3.2.1 习题	47

3.2.2	思考题	48
3.2.3	总结评分, 继续提高	48
3.3	C++基础知识(二)	48
3.3.1	数据类型	48
3.3.2	C++如何给数据起名字	50
3.3.3	算术运算(一)和赋值	53
3.3.4	顺序结构程序	57
3.4	C++基础知识(二)的亮点	58
第4章	何去何从的情况判断——选择结构程序及C++基础知识(三)	59
4.1	查询高考成绩进入哪类学校分数线的例题详解	59
4.2	勤奋实践, 总结评分	61
4.2.1	习题	61
4.2.2	思考题	62
4.2.3	总结评分, 继续提高	62
4.3	C++基础知识(三)	63
4.3.1	输入语句cin	63
4.3.2	关系运算与逻辑运算	64
4.3.3	选择结构程序	65
4.4	C++基础知识(三)的亮点	69
第5章	简化循环往复的事情——循环结构程序及C++基础知识(四)	70
5.1	累计产品数量的例题详解	70
5.2	勤奋实践, 总结评分	72
5.2.1	习题	72
5.2.2	思考题	73
5.2.3	总结评分, 继续提高	73
5.3	C++基础知识(四)	73
5.3.1	算术运算(二)	73
5.3.2	几种循环结构程序	75
5.3.3	几种转向语句	79
5.4	C++基础知识(四)的亮点	86
第6章	同类数据的组合——数组	88
6.1	往大房间(数组)里放数据并显示的例题详解	88
6.2	勤奋实践, 总结评分	89
6.2.1	习题	89
6.2.2	思考题	90
6.2.3	总结评分, 继续提高	90
6.3	数组和字符串知识	90
6.3.1	数组知识	90
6.3.2	字符串知识	99

6.4	数组和字符串的亮点	101
第7章	数据的指南针——指针	103
7.1	例题详解	103
7.1.1	通过指针得到单个数据	103
7.1.2	通过指针得到一组数据	104
7.1.3	运用指针观察分析商场销售情况	105
7.2	勤奋实践, 总结评分	106
7.2.1	习题	106
7.2.2	思考题	106
7.2.3	总结评分, 继续提高	106
7.3	指针知识	107
7.3.1	在什么情况下用到指针	107
7.3.2	关于指针	107
7.3.3	如何运用指针	111
7.4	指针的亮点	115
第8章	多种数据信息的整体包装——结构	116
8.1	打印学生成绩单的例题详解	116
8.2	勤奋实践, 总结评分	119
8.2.1	习题	119
8.2.2	思考题	120
8.2.3	总结评分, 继续提高	120
8.3	结构知识	121
8.3.1	在什么情况下用到结构	121
8.3.2	关于结构	121
8.3.3	如何运用结构	123
8.3.4	结构的特殊运用——链表	130
8.3.5	控制输出格式法	142
8.4	结构的亮点	144
第9章	可灵活方便地互换套用的一段程序——函数	146
9.1	计算长方形面积和半圆形面积的多函数的例题详解	146
9.2	勤奋实践, 总结评分	149
9.2.1	习题	149
9.2.2	思考题	149
9.2.3	总结评分, 继续提高	149
9.3	函数知识	150
9.3.1	在什么情况下用到函数	150
9.3.2	关于函数	150
9.3.3	如何运用函数	159
9.4	函数的亮点	168

第 10 章	简明方便的别名代替——引用	170
10.1	张三（真名）和小三（别名）是同一个人的例题详解	170
10.2	勤奋实践，总结评分	171
10.2.1	习题	171
10.2.2	思考题	171
10.2.3	总结评分，继续提高	172
10.3	引用知识	172
10.3.1	在什么情况下用到引用	172
10.3.2	关于引用	172
10.3.3	如何运用引用	173
10.4	引用的亮点	176
第 11 章	方便调试移植的多个文件——多文件结构	177
11.1	求长方形面积和半圆形面积的例题详解	177
11.2	勤奋实践，总结评分	179
11.2.1	习题	179
11.2.2	思考题	179
11.2.3	总结评分，继续提高	179
11.3	多文件结构知识	180
11.3.1	在什么情况下用到多文件结构	180
11.3.2	关于多文件结构	180
11.3.3	如何运用多文件结构	182
11.4	多文件结构的亮点	182
第 12 章	面向对象的事物分类及其编程——类、对象和面向对象的程序设计初步	183
12.1	打印一个右上三角形图案的例题详解	183
12.2	勤奋实践，总结评分	185
12.2.1	习题	185
12.2.2	思考题	185
12.2.3	总结评分，继续提高	185
12.3	类、对象和面向对象的程序设计初步知识	185
12.3.1	在什么情况下用类、对象和面向对象的程序设计	185
12.3.2	关于类、对象和面向对象的程序设计初步	186
12.3.3	如何运用类、对象和面向对象的程序设计	193
12.4	类、对象和面向对象的程序设计初步的亮点	194
第 13 章	事物的继承和派生——继承、派生及面向对象的程序设计方法	195
13.1	计算平均分及打印学生成绩清单的例题详解	195
13.2	勤奋实践，总结评分	200
13.2.1	习题	200
13.2.2	思考题	201
13.2.3	总结评分，继续提高	201

13.3	继承、派生及面向对象的程序设计方法知识	201
13.3.1	在什么情况下用到继承、派生及面向对象的程序设计方法	202
13.3.2	关于继承、派生及面向对象的程序设计方法	202
13.3.3	如何运用继承、派生及面向对象的程序设计方法	206
13.4	继承、派生及面向对象的程序设计方法的亮点	207
第 14 章	在对象中使用运算符——运算符重载	209
14.1	计算全班所有科目的总平均分的例题详解	209
14.2	勤奋实践, 总结评分	212
14.2.1	习题	212
14.2.2	思考题	212
14.2.3	总结评分, 继续提高	212
14.3	运算符重载知识	213
14.3.1	在什么情况下用到运算符重载	213
14.3.2	关于运算符重载	213
14.3.3	如何运用运算符重载	216
14.4	运算符重载的亮点	217
第 15 章	程序运行中需要利用前段的数据——静态成员	218
15.1	电影院统计入场人数的例题详解	218
15.2	勤奋实践, 总结评分	220
15.2.1	习题	220
15.2.2	思考题	221
15.2.3	总结评分, 继续提高	221
15.3	静态成员知识	221
15.3.1	在什么情况下用到静态成员	221
15.3.2	关于静态成员	221
15.3.3	如何运用静态成员	223
15.4	静态成员的亮点	225
第 16 章	程序运行中不确定的存储容量大小——堆	226
16.1	运用堆存储和使用高考被录取者的成绩的例题详解	226
16.2	勤奋实践, 总结评分	228
16.2.1	习题	228
16.2.2	思考题	228
16.2.3	总结评分, 继续提高	228
16.3	堆知识	228
16.3.1	在什么情况下用到堆	228
16.3.2	关于堆	229
16.3.3	如何运用堆	231
16.4	堆的亮点	232

第 17 章 和谐友好、随意使用的成员——友元	234
17.1 运用友元计算全班所有科目的总平均分的例题详解	234
17.2 勤奋实践, 总结评分	236
17.2.1 习题	236
17.2.2 思考题	236
17.2.3 总结评分, 继续提高	236
17.3 友元知识	237
17.3.1 在什么情况下用到友元	237
17.3.2 关于友元	237
17.3.3 如何运用友元	238
17.4 友元的亮点	238
第 18 章 输入/输出的信息管理——输入/输出流	239
18.1 例题详解	239
18.1.1 运用 I/O 文本文件实现字符串的写入与读出	239
18.1.2 采用文件 I/O 流管理学生成绩档案	241
18.2 勤奋实践, 总结评分	244
18.2.1 习题	244
18.2.2 思考题	244
18.2.3 总结评分, 继续提高	244
18.3 输入/输出流知识	245
18.3.1 在什么情况下用到输入/输出流	245
18.3.2 关于输入/输出流	245
18.3.3 如何运用输入/输出流	252
18.4 输入/输出流的亮点	256
第 19 章 突发事件的救急——异常处理	258
19.1 错误输入了所规定字符的异常情况处理的例题详解	258
19.2 勤奋实践, 总结评分	260
19.2.1 习题	260
19.2.2 思考题	261
19.2.3 总结评分, 继续提高	261
19.3 异常处理知识	261
19.3.1 在什么情况下用到异常处理	261
19.3.2 关于异常处理	262
19.3.3 如何运用异常处理	265
19.4 异常处理的亮点	272
第 20 章 综合设计面向对象的程序	274
20.1 面向对象的综合设计范例——仓库材料管理软件	274
20.2 勤奋实践, 总结评分	279
20.2.1 习题	279

20.2.2	思考题	283
20.2.3	总结评分, 继续提高	284
20.3	面向对象的综合设计知识	285
20.3.1	在什么情况下用到面向对象的综合设计	285
20.3.2	关于面向对象的综合设计	285
20.3.3	如何运用面向对象的综合设计	287
20.4	面向对象的综合设计的亮点	307
附录	308
附录 A	常用的 C/C++ 标准库函数	308
附录 B	ASCII 码表	313
参考文献	316

第1章 准备好学习 C++

1.1 C++是一种接近生活的高级算法语言

学习程序设计语言的目的是要编写计算机软件程序，然后把软件程序输入给计算机，让计算机运行软件程序去解决生产、生活、工作等方面的实际问题。C++语言也不例外，它是多种程序设计语言中的一种，并且是一种高级程序设计语言，所谓高级，是指它接近于人类生活中的语言。人类用 C++语言去表达、描述和处理事情与人本身的做法是很接近的，所以，人类运用 C++语言是比较直接、方便和容易的。那么，C++语言的特性是什么呢？它有什么优势呢？为什么要学习 C++呢？本节会一一进行介绍。

1.1.1 C++的特性

C++的主要特性是：**用 C++语言编写程序，就像写文章一样**，文章是由一个个字、词组成一个句子，又由若干个句子组成一篇文章，C++程序也是一样，由一个个关键字、名字、代号等组成一条语句，又由若干条 C++语句组成一篇 C++文章——C++程序；另外，文章有议论文、应用文、散文、诗歌等体裁，C++程序也有多种程序结构。总之，编写 C++程序与写文章是相近的。

此外，编写 C++程序的思路与人想事情、做事情的思路是很接近的。

例如，计算一个式子 $t=5+6$ ，并且写出 t 的答案。

C++的核心程序部分是这样表达的：

```
t=5+6,  
cout<<"t"<<t,
```

程序运行结果是：

```
t=11,
```

可以看出，这与人的想法、做法很接近， $t=5+6$ 是完全一样的，`cout` 对应于“答”，计算机会把双引号引起的字符信息原样显示出来，即 `t=`，接着又把答案显示在等号后，所以运行结果就是 `t=11`。其实，只要能将人的思路对应到 C++的字、语句、结构（这个例子采用的是顺序结构，即先计算，再写答案）上，C++程序就编写出来了。

C++还有一些技术上的特性，例如，支持数据封装、类的继承、函数的多态等，这些都暂时不用知道得那么清楚，因为要编写成功的 C++程序，这些技术特性自然会体现在其中。

1.1.2 C++的优势

从上述 C++的特性可以看出，C++有如下一些优势：

1) 编写 C++程序十分方便、容易，只要将人做事的思路、步骤进行适当的转换，就能形成 C++程序。

2) C++语言简洁、结构紧凑、运用灵活。

3) C++语言易学、易懂。

4) C++语言应用广泛。

1.1.3 学习 C++的必要性和重要性

学习 C++语言，是为了用这种功能强大的语言创造计算机的脑袋——计算机软件程序，要是计算机没有了这个脑袋，想象一下计算机会变成什么——一台死机器，同样，全自动电饭锅、全自动洗衣机、自动化装置等，也都是这样的结局。

学习 C++的必要性和重要性就在这里了。

作者了解过一些学生，他们迷上了 C++就像迷上了电脑游戏，其乐趣还胜于玩电脑游戏。在一道道 C++编程题的创作过程中，你主动指挥计算机动作，而计算机在你的指挥下，乖乖地按你的命令运行，得到了你心想事成的结果。这些学生感受到了真正的乐趣，从与他们的交谈中，看得出他们内心的那份喜悦，那份向往，那份追求，在他们的作业反馈、教学反馈中，也颇有同感。所以，C++的伟大之处，在于你可以通过它指挥计算机动作，即做到机随人行，这在后面的学习中会有越来越深的体会。

1.2 C++的难处

不是说 C++易学、易懂、易运用吗？怎么又说学习 C++有难处了呢？是的，任何事情都有它的难、易两面性。同样一件事情，不同的人去做，其难易程度就不同。下面来分析 C++究竟难在哪里，为什么大部分学习者都认为 C++难学。

1.2.1 C++本身的难处

C++语言具有众多的优点和技术性能，伴随而来的则是复杂的知识，包括丰富的运算符、数据类型，众多的名词术语和规定，许多已编好的 C++标准函数、模板，可以直接访问内存地址，进位位操作，指针，面向对象的程序设计等，这往往使得初学者望而却步。

1.2.2 人为造成的学习 C++的难处

社会上都说 C++难学，一传十，十传百，不难也认为难了，特别是对于初学者，人云亦云，乍一看，那么多名词术语，确实难，这种先入为主的观念是造成 C++难学的原因之一。

作者通过多年来在教学实践中与学生的交流，以及在网络上获取的信息，总结出了 C++难学的**最主要原因：学习者的学习目的不正确，学习意志不坚强**。例如，一些学习者是为考试、为学分、为文凭而学，即不是真的想学，而是应付着学，这是学不好的。另外，在学习 C++的过程中碰到问题、困难时，部分学习者还会表现出学习意志上的不坚强，甚至打退堂鼓，这当然不可能学好了。

还有就是教的方法不当，例如，有些教师没有充分备好课，没有很好地引导学生去学，等等，也会造成 C++ 难学。

此外，一些学习者的学习方法不当也会造成 C++ 难学。例如，**一开始就直接用 C++ 写语句，而不是先把人的做法想通**，人都想不通，机怎么随人行呢？还有就是**不愿意写提纲、画流程图，不愿意纸上谈兵，一开始就在电脑上编程**，想到一句就敲入一句，这也不可能很好地掌握 C++。另有些学习者**学了不记，不实践，不独立编程，而是依赖于从书上、网上找程序**，没有充分调动大脑的主动性去学，没有亲自从头到尾去编写程序，这也等于没有真正学习 C++，结果肯定是没法掌握 C++。

一般 C++ 的课程学习是安排在大学一年级或大学二年级，这一阶段的大学生大多数还是以高中时期的机械记忆方式为主，而不是以理解记忆为主，再加上现在很多 C++ 题目的答案在网上比比皆是，所以学生只要找到相应的题解，基本上不用怎么理解就可以应付学习了，但换了药（实际问题）之后，平时只会处理换汤不换药问题的学习者就无从下手了。所以学习 C++ 最重要的是自主解题，勤学多练，学用结合，巩固提高。此外，有的学习者还是立足于学例题、用例题的方式，热衷于搞题海战术，但例题弄得再明白，有可能还是不会自己编写程序，因为你只会跟着别人的思路走，而没有培养好自己的编程设计思路，所以，就很难做到独立编程。

在网上了解到，在些学习者以为数学和英语要学得好才能学好 C++。这基本没有直接的关系，虽说 C++ 的关键字、语句等都是用英语写的，但和学英语至少要记 3000 个以上的单词相比就少得多，因为 C++ 中常用的字（关键字和保留字）就那么几十个，至于 C++ 标准函数名和类名等，在使用时查表就知道了，不需要记。关于数学方面，一般都知道或可以查到数学公式、算法及解题过程，只要学习者把人的解题过程用 C++ 语言来实现就行了。

还在网上了解到，有的学习者介绍他们的入门方法是这样的：通过学习 C++ 的知识和读懂 C++ 程序的案例，然后转换成日常人能接受的自然语言去学习。这是弄反了思维过程。应该是人先有想法，产生了日常自然语言，再去编写 C++ 程序，这样，才符合人的思维逻辑。而不是反过来，先有了结果（程序），再去思考人的想法。

在网上，又有学习者说：在编写程序时，先记录自己想的自然语言，然后写程序代码，再将程序代码转变为流程图。通过这样的过程提高自己的抽象逻辑思维能力。这就绕了个大圈子，有些步骤也弄反了。实际上，大可不必这么麻烦，按照本教材中介绍的“人定胜机，机随人行”法则，先把人解决问题、处理问题的想法和做法想通，然后通过提纲或简易流程图或流程图记录下来（即自然语言），再用 C++ 语言对号入座填写到流程图中的相应位置，则程序就出来了。接下来，用人脑跟踪运行，再进行上机调试、运行，则计算机就可以帮助读者完成任务了。该过程相当于写一篇文章，即先写提纲（或流程图），再写文章（程序），对于提纲大家是会写的，就是将人做事的过程和步骤用文字进行描述，剩下的问题是怎样将其转换成 C++ 程序，只要读者认真学习，用心记住 C++ 语句的基本功能，并学会按功能转换使用 C++ 语句即可。另外，在程序编写中，并没有太多的抽象，而主要是分解和细化，所以，读者不要把 C++ 想得那么高深。

也就是说，难处大多是人的学习态度、学习思路造成的，或是由于学习方法不当造成的。事实上，学习 C++ 并不难。

有的学习者一开始学 C++，就想用 C++的工具编写程序，但由于工具较多、书写较长、代号也较多，实现起来相当不容易。因此，要踏实学好 C++语言的知识 and 编程思想，打好软件设计基础，并用 C++知识和编程思想去融会贯通、举一反三，这样，即使再复杂的程序、再多 C++工具的理解和应用都不成问题，加之努力学习和勤奋实践，学习者就会熟能生巧、运用自如了。

1.3 化难为易地学习 C++之精髓

1.2 节讲了 C++本身存在的难处和人为造成的学习 C++的难处，本节将讲解如何把 C++的难化为易，并道出学习 C++的精髓——“**寓教于教（学），以德促教（学），人定胜机，机随人行**”的十八字方针。教师应遵循“**寓教于教，以德促教，人定胜机，机随人行**”的十六字方针教学，学习者应遵循“**寓教于学，以德促学，人定胜机，机随人行**”的十六字方针学习 C++。

1.3.1 道德品质好，成才的基本条件

从古至今，大凡是有成就、成大器的人士，无一不是德高望重的人，所以人的道德品质决定了人的思想意识，决定了人的一言一行，决定了人的善恶好坏，决定了人的素质高低，也决定了人的事业成败。

学习者应该具备良好的道德品质，因为只有具备了良好的道德品质，才会在思想上树立起远大的目标、信念，才会不畏艰险、努力学习、勤于思索、勇于创新、积极进取、淡泊名利、团结协作，积极参加各种创新项目和社会实践活动，学到较多的知识，无私地为人类、为社会做贡献，从而成为高素质的创新人才，成为社会的有用之才，成为祖国合格的接班人。

归根到底，培养高素质创新人才的关键在于培养人的道德品质，只有这个基础打好了，高楼大厦的根基稳固了，才能通过日后的添砖加瓦，将整栋高楼大厦建成。所以，以道德品质为引领去培养高素质的创新人才是极其必要和重要的，以道德品质为引领去学好各种知识和本领同样是极其必要和重要的。

总之，**要把学习 C++当做是日后为祖国、为人民多做贡献而积累知识和本领**，要树立这样的远大目标和理想去学习，这样才能在心中有一盏明灯，才能在思想上把 C++化难为易。在学习的过程中，无论大家碰到什么困难，都不要气馁，要勤学苦练、勇攀高峰，这样才能学好 C++。

所以说，**道德品质好，是学好 C++的关键，是把 C++化难为易的关键，是成才的基本条件。**

1.3.2 心有灵“C”一点通

除了以高尚的道德品质引领 C++的学习外，在具体学习 C++知识时，还要做到老老实实、谦虚谨慎、勤学苦练，因为科学知识是来不得半点虚伪的。

学习者在学习 C++时要摒弃那种只用 C++解决数学题和只研究 C++语法题的观念，而应把学习 C++的视野扩大，将 C++与日常生活中的问题切实地联系起来。

国有国法，家有家规，学习 C++就必须按照 C++的语法、规则、要求去做，如若逆之，C++这个裁判就会出示黄牌或红牌警告，程序就会调试不通。所以掌握好 C++的语法、规则、要求是势在必行的，即必须要在心中装有 C++这个灵丹妙药才行得通，否则，就寸步难行了。

怎样才能达到心有灵“C”一点通呢？应该把 C++的关键字、语句及语法规则、程序结构、数据结构牢牢记住，要记住它们的功能、用法，并能熟练运用。当然，不是一下子就能把 C++的所有知识学会、记住的，不能有急功近利、急于求成的心态，饭要一口一口地吃，仗要一个一个地打，学习 C++必须要从容镇静、戒骄戒躁、严谨细致、前后联系、相互借鉴地学，要学一点、记一点、用一点、消化体会一点、联系一点、总结一点、提高一点，再学一点、再记一点、再用一点、再消化体会一点、再联系一点、再总结一点、再提高一点，依此循环往复，日积月累，C++这个知识球就会像滚雪球一样，越滚越大，心中的灵“C”就会越来越多，最终能达到融会贯通、运用自如的境界。

往往许多学习者学了不记或记不牢 C++的关键字、语句等知识，即脑袋空空，当然在编写程序时就困难重重，不知道该用什么词来造句，也不知道要造哪些句，不知道文章的总体结构应怎样设置等，结果就是编不出程序。就像小学生学中文一样，先学文字，再学词组，进而是学语法、造句，这些基本知识学到一定程度了，才有能力写文章，掌握 C++这门高级算法语言自然也应该遵循学中文的规律。

所以，**一定要边学边用、边用边记、认真体会、总结提高**，方能达到心有灵“C”一点通。

不但要学懂 C++知识，**还要学会学习、运用 C++ 的方法**，不是单纯地为了学习而学习，而是要达到举一反三、事半功倍的效果。

1.3.3 人定胜机，机随人行

人脑聪明还是电脑聪明？自然是人脑聪明，计算机的硬件和软件均是人设计制造出来的，人肯定比电脑聪明，肯定胜于电脑。所以应**坚定不移地树立起“我一定能学好 C++，我一定会成为 C++高手”这个信念**，这个信念会使你的精神堡垒更加坚固，当学习中遇到困难时，这个信念会站出来说：“**不用怕，人定胜机**”，会给你信心，给你勇气，助你成功；当你编写程序时，这个信念会支撑你：“**人定胜机，我比电脑强，我一定能编写出程序，我一定会成功的！**”。

“人定胜机”体现了人自信、自立、自强的精神，体现了不达目的誓不罢休的锐意进取精神。

电脑能运行是由于它既有硬件又有软件，而软件是人编写出来的，电脑的运行完全按照人的旨意运作，所以，机器是随人行的。如果要机器正确运行，则首先人设计的程序要正确；如果要使程序的逻辑正确，则人的设计思路得正确；如果要做到设计思路正确，则要求人要想通自己的做法。反过来说就是，编写程序时，**人要先想通自己是怎么做的，不要先想 C++，就只想人是怎么做的，然后就让机器跟着怎么做，即将人的做法转换成 C++程序去处理事情**，这就是“**机随人行**”了。

此外，还要求人要胆大心细，不要前怕狼，后怕虎，要大胆尝试，勇于创新，因为机器始终要乖乖地服从人的命令和指挥的，所以，不要怕电脑，而要信人脑！