

C++语言 (中学版)

主 编 张文双
副主编 杨印国

前 言

为深入贯彻落实《国家教育事业发展“十三五”规划》，进一步推进《教育信息化2.0行动计划》，加强对学生科学素质、信息素养和创新能力的培养，信息学奥林匹克竞赛给有才华的学生提供了相互交流和学习的机会，培养和选拔出一批优秀的计算机人才。

近年来，C++语言已成为我国青少年信息学奥林匹克竞赛（NOI）和分区联赛（NOIP）的主导竞赛语言。为了适应竞赛的需要，同时针对中学生的数学基础和认知能力水平，我们编写了这本中学版的C++语言教程，书中所有的例题和习题均能在Dev-C++ 5.11环境中运行。

这本书的内容共分15章，主要内容包括：C++语言基础知识、C++语言的顺序结构、选择结构、循环结构、数组、函数、字符串、指针、结构体与联合体、文件、数据结构、算法、编译宏指令及编程小技巧、NOI Linux操作系统等内容。其中，第1~3章由杨印国编写，第4章由冯新晖编写，第5章由李秀丽编写，第6章由王金艳编写，第7和第8章由康健编写，第9章由陈进编写，第10章由刘树明编写，第11章由张文双编写，第12和第13章由杜柏林编写，第14章由侯启明编写，第15章由蔡英奇编写。全书由张文双、杨印国统稿审定。本书第14章作者侯启明作为2002年第十四届国际青少年信息学奥林匹克竞赛金奖获得者，将个人的编程技巧奉献给读者，希望能为您带来帮助。

由于编者的水平有限，书中若有疏漏之处，恳请各位读者指正。

编 者
2019年5月

目 录

第 1 章 初识 C++语言	1
1.1 集成环境及菜单的使用	2
1.2 程序的输入和调试	5
习题 1	9
第 2 章 C++程序设计语言基础	11
2.1 C++语言简介	11
2.2 数据类型	17
2.3 常量和变量	18
2.4 函数与表达式	22
2.5 数制	31
2.6 ASCII 码	34
习题 2	35
第 3 章 顺序结构程序设计	37
3.1 赋值语句	38
3.2 输入语句	39
3.3 输出语句	39
3.4 应用实例	41
习题 3	44
第 4 章 选择结构程序设计	46
4.1 逻辑运算及布尔表达式	46
4.2 条件 (if) 语句	48
4.3 if 语句的嵌套	53
4.4 switch 语句	55
习题 4	60
第 5 章 循环语句	63
5.1 while 循环语句	63
5.2 do-while 循环语句	68



5.3 for 循环语句	73
5.4 多重循环	78
习题 5	83
第 6 章 数组	87
6.1 认识数组	87
6.2 一维数组	88
6.3 二维数组	96
6.4 多维数组	100
6.5 字符数组	101
6.6 数组的综合应用实例	104
习题 6	107
第 7 章 函数	113
7.1 函数的定义	113
7.2 函数的类型和返回值	115
7.3 函数的调用与参数的传递	116
7.4 变量及其作用域	120
7.5 函数的应用	122
7.6 递归函数	124
习题 7	129
第 8 章 字符串	132
8.1 string 类型的初始化	132
8.2 string 类型的基本操作	134
8.3 应用实例	138
习题 8	140
第 9 章 指针	143
9.1 指针变量的定义	143
9.2 指针变量的基本操作	145
9.3 指针的应用	150
习题 9	152
第 10 章 结构与联合体	154
10.1 结构的定义	154
10.2 访问结构成员	157

10.3 结构应用举例	160
10.4 结构与链表	164
10.5 联合体	171
习题 10	173
第 11 章 文件	176
11.1 文件的类型	176
11.2 文本文件的概念	177
11.3 文件类型变量的定义及引用	177
11.4 文件的重定向	183
习题 11	187
第 12 章 数据结构	189
12.1 线性表	189
12.2 栈	192
12.3 队列	196
12.4 树	197
12.5 二分法	201
习题 12	204
第 13 章 常用算法	207
13.1 穷举法	207
13.2 排序算法	209
13.3 回溯算法	213
13.4 递推法	215
13.5 高精度数值处理	219
13.6 动态规划	224
习题 13	227
第 14 章 编译宏指令及编程小技巧	228
14.1 编译宏指令	228
14.2 编程小技巧	240
习题 14	247
第 15 章 NOI Linux 操作系统	249
15.1 NOI Linux 系统的介绍和安装	249
15.2 NOI Linux 的编程操作	254



15.3 NOI Linux 系统下评测及注意事项	262
15.4 对拍	269
习题 15	276
习题参考答案	278
第二十四届全国青少年信息学奥林匹克联赛初赛提高组 C++语言试题	308
附录	318
附录一 ASCII 码表	318
附录二 C++常见出错信息	319
附录三 C++常用词汇英汉对照	320
附录四 C++程序设计中的常用数学知识	323
参考文献	326

第 1 章

初识 C++ 语言

荷兰计算机学家德克斯特拉 (E. W. Dijkstra) 在 20 世纪 60 年代提出了结构化程序设计思想。如今被广泛应用的结构化程序设计语言 C 语言, 是 1972 年由 Dennis Ritchie (C 语言之父) 在贝尔实验室设计出来的, 由于它不适合大型程序开发等局限性, 1980 年, 贝尔实验室的 Bjarne Stroustrup 博士 (C++ 之父) 对 C 语言进行改进和扩充, 于 1983 年研制出 C++。

C++ 是一种面向对象的程序设计语言, 可运行在多种平台上, 如 Windows、Mac OS X (苹果机系统) 和 Linux 操作系统。

Dev-C++ 是一个 Windows 环境下的轻量级 C/C++ 集成开发环境 (IDE), 它使用信息学奥林匹克竞赛要求的编译器——g++。其最大的优点是功能简捷, 是学习 C++ 的首选开发工具, 很适合初学者使用。原开发公司 Bloodshed 在开发完 4.9.9.2 版本后停止开发, Orwell 公司继续更新开发, 目前的最新版本为 5.11。

本书将详细介绍 Dev-C++ 5.11 的使用方法。

在 Windows 7 系统里下载安装 Dev-C++ 5.11 (大小为 48MB)。

1. 启动方法

双击桌面上的 Dev-C++ 图标, 或者单击“开始”→“所有程序”→Bloodshed Dev-C++→Dev-C++ 命令。

2. Dev-C++ 的退出

退出 Dev-C++ 的方法有以下几种。

(1) 选择主菜单文件中的“退出”命令, 或者按快捷键 Alt+F4, 彻底退出 Dev-C++。若有未保存的程序, 系统会提示是否保存。

(2) 在输入程序代码后, 选择主菜单文件中的“关闭”命令, 只关闭当前编辑的程

序，然后还可以打开其他项目进行编辑。

1.1 集成环境及菜单的使用

Dev-C++集成开发环境的各个组成部分有：标题栏、主菜单栏、工具栏、代码输入与编辑区、位于窗口底部的编译信息显示区等，程序设计的所有工作基本上都要在该窗口中进行。界面如图 1-1 所示。

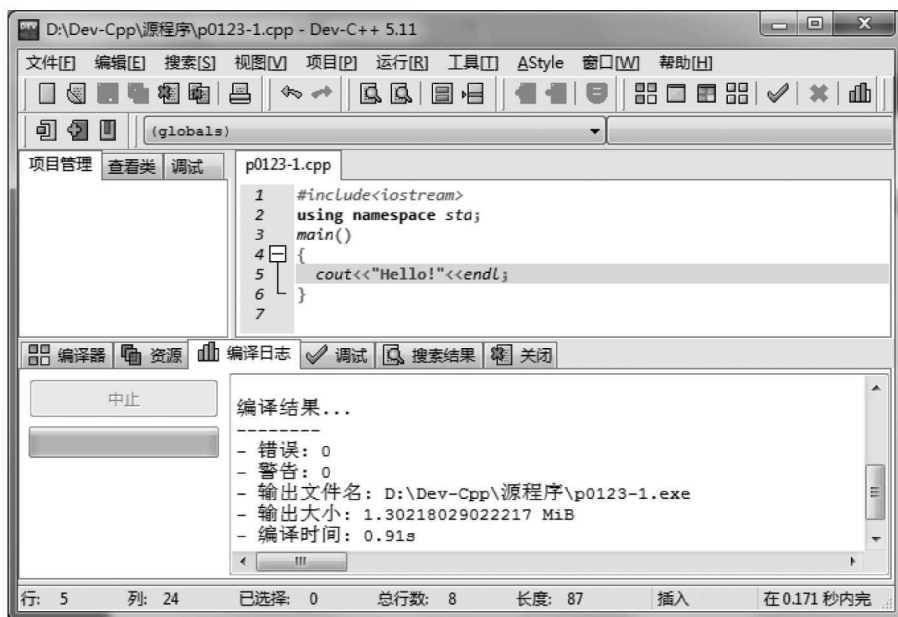


图 1-1 Dev-C++集成开发环境

1. 主菜单栏

主菜单栏位于窗口上部，共有 10 个菜单项：文件、编辑、搜索、视图、项目、运行、工具、AStyle、窗口、帮助。各个菜单又包含若干个下拉子菜单。这里给大家介绍常用的操作。

1) 文件菜单

文件菜单是最重要的菜单，主要用于对文件的管理，如新建文件、打开文件、保存文件等，具体功能见表 1-1。

表 1-1 文件菜单中各子菜单的功能

子菜单	快捷键	功能描述
新建	Ctrl+N	新建源代码，也可以创建新的项目
打开项目或文件		显示“打开文件”对话框，供选择打开已有的文件
保存		将当前的文件存盘
另存为		以用户指定的路径和文件名将当前的文件存盘
全部保存		对所有类型的文件存盘
关闭		关闭当前窗口

续表

子菜单	快捷键	功能描述
关闭项目		关闭当前正在操作的项目
导入		导入 Visual 中的代码
导出		将当前代码以 html、rtf、text 文件格式导出
打印		打印当前窗口中的内容
打印设置		对打印机进行简单的设置
退出	Alt+F4	退出 Dev-C++ 集成开发环境

2) 编辑菜单

编辑菜单用于对当前编辑窗口中的内容进行编辑，如代码的输入、修改、复制、粘贴、删除等，具体功能见表 1-2。

表 1-2 编辑菜单中各子菜单的功能

子菜单	快捷键	功能描述
恢复	Ctrl+Z	撤消最近的一次操作
重做	Ctrl+Y	撤消恢复操作
剪切	Ctrl+X	删除选定的文本，并放入剪贴板
拷贝	Ctrl+C	将选定的文本复制到剪贴板中
粘贴	Ctrl+V	将剪贴板中的内容粘贴到当前光标处
选择全部	Ctrl+A	选择全部内容
插入		插入当前日期、时间、常用的程序结构等，如 for()
注释	Ctrl+.	可以将当前行设定为注释内容，不再编译
缩进	Tab	将光标向右移动，代码实现缩进，增强程序阅读能力
取消缩进	Shift+Tab	将光标向左移动，以实现同级代码对齐，便于阅读

3) 运行菜单

运行菜单用来运行 C++ 程序，如编译程序、运行程序、编译并运行程序、语法检查等，具体功能见表 1-3。

表 1-3 运行菜单中各子菜单的功能

子菜单	快捷键	功能描述
编译	F9	当前程序
运行	F10	运行当前程序
编译运行	F11	编译并运行当前程序
全部重新编译	F12	运行程序到光标所在行
检查语法		检查源代码中的语法错误，光标停在错误处
性能分析		可以对项目进行分析
切换断点		在程序中设置断点
调试		可以对项目进行调试



4) 工具菜单

工具菜单用来选择编译器、文件默认存放路径、设置编辑器的工作模式、自定义快捷键等,具体功能见表 1-4。

表 1-4 工具菜单中各子菜单的功能

子菜单	快捷键	功能描述
编译选项		用来选择编译器
环境选项		文件默认存放路径
编辑器选项		设置编辑器的工作模式,比如输入代码时自动缩进等
快捷键选项		可以自定义快捷键
配置工具		可以选择编辑器

5) 窗口菜单

窗口菜单用来管理开发环境窗口,如关闭所有程序设计窗口、切换各个窗口、最大化窗口等,具体功能见表 1-5。

表 1-5 窗口菜单中各子菜单的功能

子菜单	快捷键	功能描述
全部关闭		关闭所有打开的程序设计窗口
全屏	Ctrl+F11	最大化程序设计窗口
后继		切换到后一个设计窗口
前驱		切换到前一个设计窗口
列表		列出打开的所有设计窗口

6) 其他菜单

其他的菜单使用较少,这里只做简单介绍。

- (1) 搜索菜单:提供了查找字符串的功能。
- (2) 替换菜单:提供了替换找到的字符串的功能。
- (3) 项目管理:提供了打开或关闭项目管理窗口的功能。
- (4) 状态条菜单:提供了显示光标所在行、代码行数和总的字符数的功能。
- (5) 工具条菜单:提供了显示各种工具栏和所采用的编译器的功能。
- (6) 格式化选项:提供了设定窗口代码格式化的样式的功能。
- (7) 格式化当前文件:按格式化选项的设定,整理窗口代码格式,如缩进。

上面介绍的是菜单的功能,有关菜单的操作还需说明几点:

- (1) 菜单操作可以使用鼠标,也可以使用键盘。
- (2) 按 Alt 键,光标移到主菜单,移动“→”或“←”键选择主菜单,按 Enter 键打开相应子菜单;再按“↑”或“↓”键选择子菜单,按 Enter 键确认。
- (3) 同时按下 Alt 键和所需主菜单项括号中的字母(一般为第一个字母),可打开其子菜单,如按快捷键 Alt+F 可打开文件菜单;再按子菜单中所需项括号中的字母,可执行相应

操作，如按 S 键可保存文件。

(4) 用快捷键直接操作子菜单，如按 F9 键可以编译源文件。

(5) 子菜单中有“▶”，说明还有下级菜单，单击带“...”的选项会打开对话框。

2. 编辑窗口

1) 窗口的组成

编辑窗口主要用于输入和编辑 C++ 源程序。编辑窗口由标题选项卡（上面有标题）、代码行号、代码折叠/展开按钮、窗口编辑区、水平滚动条和垂直滚动条、光标位置显示（如 5 行 9 列）、总行数和总字符数等组成。

C++ 可以同时打开多个窗口，但任一时刻只有一个窗口处于活动状态，称为活动窗口，也称为当前窗口，即当前正在工作的窗口。活动窗口的特点是标签突出显示，而非活动窗口的标签下沉。单击某一非活动窗口，就能使其成为活动窗口。代码编辑等所有操作都要在活动窗口中进行。

2) 窗口的操作

窗口可以被打开、移动、全屏/常规、关闭。

窗口的新建：通过主菜单文件中的“新建”命令，可以新建一个窗口。

窗口的打开：通过主菜单文件中的“打开项目或文件”命令，可以打开一个已有的文件。

窗口的移动：通过鼠标拖动标题标签，可以移动活动窗口。

窗口的全屏/常规：选择主菜单窗口中的全屏命令后，可以最大化编辑窗口。可以利用快捷键 Ctrl+F11 实现窗口的全屏和常规切换。

窗口的关闭：通过主菜单文件中的“关闭”命令，或者右击标题，选择“关闭”，也可以关闭活动窗口。通过主菜单文件中的“全部关闭”命令，可以关闭所有窗口。

3. 状态栏

通过位于窗口底部的状态栏，可以查看光标所在的行列、选择的字符数量、代码总行数、代码的字符总数。

4. 编译信息区

对代码编译后，可以给出如下样式的编译信息，便于程序的调试。

错误：0

警告：0

输出文件名：D:\Dev-Cpp\p0123-1.exe

输出大小：1.30218029022217 MiB

编译时间：1.20s

1.2 程序的输入和调试

下面通过一个简单的例子来说明 C++ 程序的输入、编辑、保存、编译、运行等操作。



例 1-1 字符串的输出。

程序如下：



```
#include<iostream>
using namespace std;
main ()
{
    cout<<"Hello!"<<endl;
    cout<<"This is a simple C++ program!"<<endl;
}
```

运行结果为:

```
Hello!
This is a simple C++ program!
```

1. 输入源程序

Dev-C++集成开发环境启动后,单击“文件”→“新建”→“源代码”命令就可以在窗口中编写代码了,用C++语言编写的程序叫C++源程序。

在输入过程中,每输完一行按一下Enter键。可随时使用编辑命令快捷键进行编辑操作,常用的编辑命令见表1-6。

表 1-6 常用的编辑命令

命令	编辑操作
←	光标左移一个字符
→	光标右移一个字符
↑	光标上移一行
↓	光标下移一行
Home	光标移至行首
End	光标移至行尾
Page Up	光标上移一页
Page Down	光标下移一页
Backspace	删除光标左边的字符
Delete	删除光标所在处的字符

如果窗口中已有程序,那么要输入新的程序,就要单击菜单“文件”→“新建”→“源代码”命令,打开一个新窗口,再输入新程序。当然也可以直接按快捷键Ctrl+N。

2. 调试源程序

程序代码输入完毕,在编译前要进行以下的调试工作。

- (1) 静态查错:为了减少无谓的调试,这一步很重要。
- (2) 很小的数据可以手工运算,不必什么都要靠调试来完成。
- (3) 把数组定义得小一点,可以减少系统资源占用,从而优化程序。
- (4) 在编程过程中要注意存盘,以防止程序意外丢失。
- (5) 模块调试。这种分治策略,可以降低调试难度。

(6) 使用“运行”菜单中的“语法检查”或“检查当前文件语法”命令，可以自动检查代码中的语法错误。存在错误的行，在行号处会出现一个×。

3. 保存程序

选择主菜单“文件”→“保存”命令，或按快捷键 Ctrl+S，保存当前程序。

如果想另存，可以选择菜单“文件”→“另存为”命令，会出现“另存为”对话框，在其中输入文件名（例如：ytr），单击“保存”按钮，则程序会以“ytr.cpp”为文件名保存在默认目录中。若要保存到其他地方，在输入文件名时指明路径就可以了。

4. 编译源程序

程序调试完毕，即可开始编译，编译就是把源程序一次性翻译成目标程序，以便计算机执行。选择菜单“运行”→“编译”命令，或者按快捷键 F9，就可以对程序进行编译，此时屏幕下方会出现编译信息。如果程序中存在语法错误，在编译时，系统会给出错误信息代码，以便于我们进行修改。如，expected ';' 提示缺少分号；expected ')' 提示缺少右括号。同时，也会指出错误所在的行列，如行 9，列 11。此时可利用表 1-6 中的编辑命令快捷键修改，再进行编译，直到成功。

在文件编译时，会自动在当前目录中产生一个与源文件同名的可执行文件，如 ytr.exe。该文件不需要在 C++ 环境中运行，而是要在操作系统环境下运行，如在 Windows 中双击文件名图标即可运行。

注意：编译前，在窗口右上角的编译器要选择“TDM-GCC 4.9.2 32-bit Release”。

5. 运行程序

程序编译通过后，就可以运行了。步骤如下。

(1) 运行程序：选择菜单“运行”→“运行”命令，或按快捷键 F10，实现程序的运行。

(2) 查看结果：运行完毕，自动产生一个窗口，显示运行结果。

注意：程序修改后，必须重新编译，再运行。否则运行结果还是未修改时的结果。

6. 常见错误信息

expected '('	缺少左括号
expected ')'	缺少右括号
expected ';'.	缺少分号
expected ';' before 'cout'	cout 前面的语句缺少分号
expected ','	缺少逗号
expected '='	缺少赋值号
missing terminating " character	缺少终止字符"
'p' was not declared in this scope	标识符未定义
invalid operands of types 'int'	int 类型变量的操作数无效
expression syntax error	表达式语法错误



操作技巧

在输入 C++ 源代码时，先要输入头文件、名字空间和函数名这三行，是不是很麻烦？



下面介绍一个工具，它能帮助我们简化这个操作。步骤如下。

(1) 单击“工具”菜单的“编辑器选项”命令，打开“编辑器属性”窗口，如图 1-2 所示。

(2) 单击“代码”按钮，在第 2 行标签中出现“缺省源”标签，如图 1-3 所示。



图 1-2 编辑器属性界面



图 1-3 代码—缺省源界面

(3) 单击“缺省源”标签，显示如图 1-4 所示界面。



图 1-4 缺省源界面

(4) 在窗口的编辑区里输入 C++ 程序的头文件、命名空间和函数名。选中“向项目初始源文件插入代码”选项。然后单击“确定”按钮，完成缺省源代码的设置。

当再新建源代码文件时，C++ 窗口如图 1-5 所示。

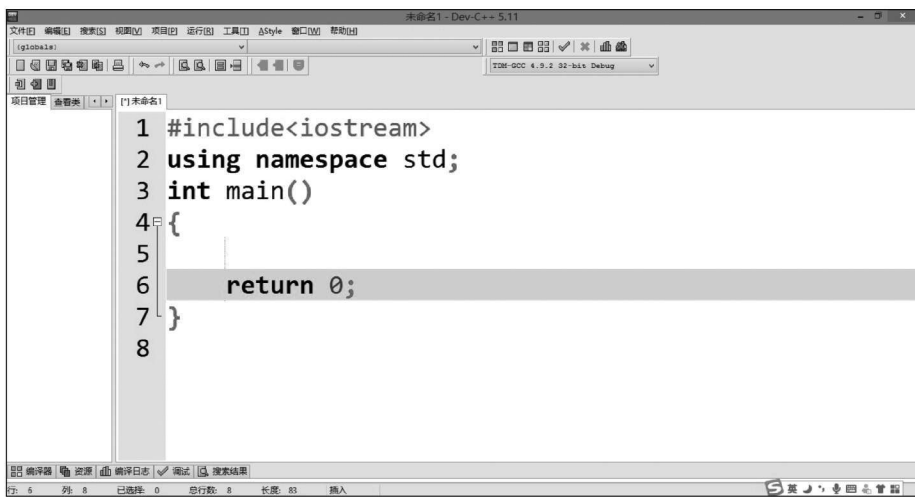


图 1-5 设置缺省源后，新建源代码窗口

窗口的编辑区里果然出现了 C++ 程序的头文件、命名空间和主函数的框架，输入程序代码时节省了输入的工作量，很方便。

习题 1

一、简答题

1. C++ 语言是谁研究出来的？
2. 如何编译 C++ 源程序？要注意什么问题？
3. 在 C++ 中，如何打开多个窗口？它们之间怎么切换？
4. C++ 程序需要编译，请说明编译的含义。
5. 试说明怎样由键盘来操作主菜单？

二、分析题

完成下面程序的输入、保存、编译、运行，分析并观察运行结果。
程序如下：

```
#include<iostream>
using namespace std;
main()
{
    int x;
    x=1+2;
    cout<<"the sum is";
    cout<<x;
    cout<<endl;
    cout<<"all done!"<<endl;
}
```



```
}
```

三、释义题

指出下列错误信息的含义:

1. expected ')'
2. expected ';'
3. 'p' was not declared in this scope
4. expected ';' before 'cout'
5. missing terminating " character

第 2 章

C++程序设计语言基础

C++语言属于编译型的高级语言，是在 C 语言的基础上发展起来的，是一种面向对象的程序设计语言。C++是“C Plus Plus”的简写。

2.1 C++语言简介

2.1.1 C++语言的特点

编写一个程序，就是以一种程序设计语言所要求的规范在表达思想，这和人与人之间使用自然语言交流一样，程序员和计算机使用程序设计语言交流。用 C++编写的程序叫源程序，并以“.cpp”文件的形式保存在计算机中。C++源程序必须经过编译器这个软件进行编译，翻译成目标文件，格式为“.obj”。目标文件也不能直接执行，必须通过连接器与C++系统库连接在一起，形成一个“.exe”格式的可执行文件，才能执行。

C++语言具有如下特点。

- (1) 全面兼容 C 语言，绝大多数 C 程序可以在 C++中直接运行。
- (2) 程序结构紧凑、语句灵活、运行可靠。语法要求严格，不易出现编写错误。
- (3) 运行效率比其他任何高级语言都要高，仅次于汇编语言（低级语言）。
- (4) 支持面向对象程序设计。
- (5) 可扩充性、可维护性强，适合开发大中型软件，可以编写系统软件，驱动程序和