

■ 青少年人工智能教育课堂 ■

C 语言 程序设计 EASY C PROGRAMMING 第一课

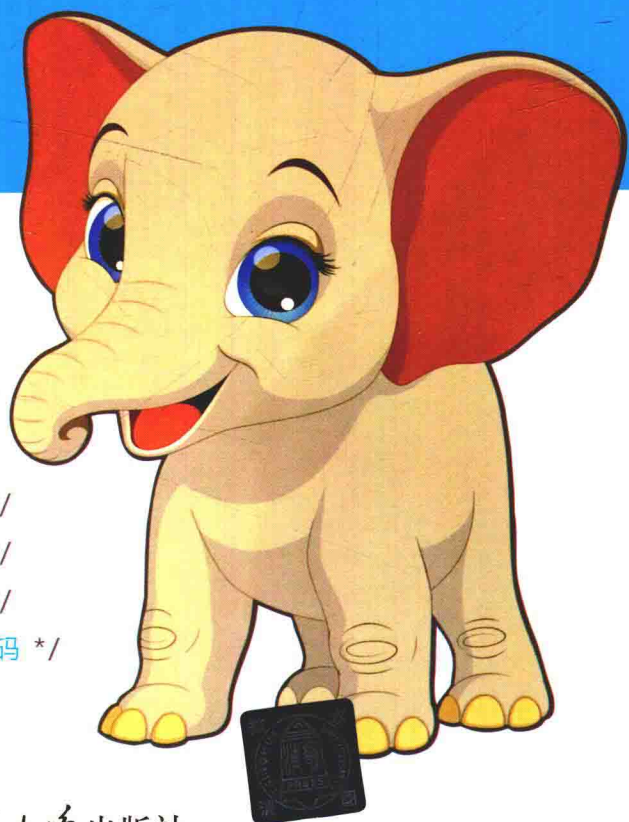
赵 军 编著



多媒体视频教学
扫码阅读

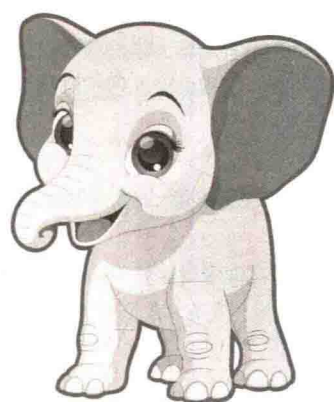
图文并茂讲解C语言基本概念

- 01 大量程序范例；/* 活用语法练习 */
- 02 综合实践范例；/* 强化设计思想 */
- 03 课后学习园地；/* 检验学习成果 */
- 04 轻松上手；/* 编写自己的第一行代码 */
- 05 return 0;



清华大学出版社

C 语言 程序设计 第一课



赵 军 编著

清华大学出版社
北 京

内 容 简 介

这是一本适合学习C语言编程的入门教材,全书通过丰富的范例对C语言的基础语法进行通俗明了的讲解,内容包括C语言的数据处理、表达式与运算符、选择性流程控制、循环流程控制、数组与字符串以及函数等的应用。

本书强调学用相结合,丰富的范例程序结合上机实践,教你领会C语言结构化编程的基本概念;综合范例练习帮助你强化语法的理解能力;各章的课后练习可马上检验你的学习效果;另外还有完整的教学视频可供下载,以辅助你更加高效地自学。

希望本书能降低中学生学习C语言编程的门槛,减少编程初学者自学的障碍,成为大家进入程序设计领域的第一课,同时为进一步学习人工智知识理论、应用拓展、创新设计等打下坚实的基础。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

C语言程序设计第一课/赵军编著. —北京:清华大学出版社, 2018
ISBN 978-7-302-50639-3

I. ①C… II. ①赵… III. ①C语言—程序设计 IV. ①TP312.8

中国版本图书馆CIP数据核字(2018)第156480号

责任编辑:夏毓彦

封面设计:王翔

责任校对:闫秀华

责任印制:董瑾

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦A座

邮 编:100084

社总机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印装者:北京嘉实印刷有限公司

经 销:全国新华书店

开 本:190mm×260mm

印 张:13.5

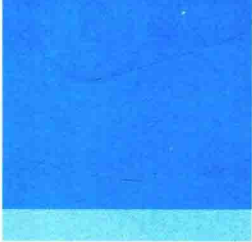
字 数:302千字

版 次:2018年10月第1版

印 次:2018年10月第1次印刷

定 价:49.00元

产品编号:079690-01



前言

人工智能技术的未来就是信息技术的未来，而“程序设计”或称为“编程”是学习人工智能技术的逻辑编程类和通过编程实践解决问题的基础课程之一，已经列入中学的信息技术课程，对于将来人才综合素质的评估，即便是非计算机或信息类专业的人才，程序设计也是必备的基础能力之一。

C 语言称得上是一门历史悠久的高级程序设计语言，也往往是现代程序设计初学者最先接触的程序设计语言，对近代的计算机科学领域有着非凡的贡献。C 语言持续屹立不倒已达 40 余年，无论是后来的 C++、Java、PHP，还是 .NET 中的 C#、VB.NET 等，都是以 C 语言作为参考发展起来的。因此，学会 C 语言往往是学习其他程序设计语言的基础。只有当我们具备了逻辑编程的坚实基础和通过编程实践来解决问题的能力，才能进一步学习人工智能的知识理论类的课程、应用拓展类课程、创新设计类的课程。

C 语言具备高级语言的结构化语法，有高度可移植性与强大的数据处理能力。绝大多数硬件驱动程序、网络协议都是 C 语言所编写的，特别是以 C 开发出来的程序，其执行效率相当高，也相当稳定，深受许多程序设计者的喜爱。

市面上关于 C 程序设计方面的书非常多，但编写的主要内容通常适用于大专院校的程序设计语言课程，这类书的定位较不适合中学生和初学者，在这种思路的指引下，希望可以编写一本适合中学生入门和初学者自学的教材。因此本书讲述的内容以基础语法为主，再导入一些简单的函数基本概念，希望学习者可以通过有趣且多样的简易范例程序，轻松学会 C 程序设计语言。

另外，本书在各章结束前除了综合程序范例外，还安排了课后习题，可用于检验学习成效。因此，本书非常适合作为 C 语言的入门教材。笔者通过大量的范例程序来帮助初学者学习，以便快速带领大家进入 C 语言程序设计领域。

本书由赵军主编，参与本书编写的人员还有张明、王国春、施妍然、王然等。由于编者水平和经验所限，书中难免存在疏漏和不足之处，希望得到大家的批评指正。

本书为读者特意录制了教学视频，希望能降低中学生的学习门槛，减少初学者的自学困惑。

读者可以从以下地址（注意区分数字与英文大小写）下载所有范例程序的源代码、教学 PPT 和全程视频文件：

https://pan.baidu.com/s/1cJ5-nuB4m8_c5Opul3KTHA

也可以扫描右方的二维码下载，如果下载有问题，或者对本书有任何疑问与建议，请联系 booksaga@126.com，邮件主题为“C 语言程序设计第一课”。



最后，为了便于读者在学习中进行讨论和交流，我们还建立了“程序设计第一课讨论群”（QQ 群），大家可以在群里讨论问题，笔者也会对重点问题进行解答。QQ 群号为 801630455，也可以直接扫描进群二维码：



编 者

2018 年 7 月

目 录

第

1

章

C 语言初步体验

1.1 认识程序设计语言	2	1.4.1 程序代码的编写	12
1.1.1 机器语言	2	Hello World 程序	
1.1.2 汇编语言	3	范例程序: CH01_01.c	13
1.1.3 高级语言	3	1.4.2 程序代码的编译与运行 ...	14
1.1.4 非过程性语言	4	1.4.3 程序调试简介	15
1.1.5 人工智能语言	4	1.5 程序代码快速解析	16
1.2 C 语言的特色	4	1.5.1 头文件与 #include 指令 ...	17
1.2.1 硬件沟通能力	5	1.5.2 main() 函数	18
1.2.2 高效率的编译型语言	5	1.5.3 printf() 函数与注释	18
1.2.3 程序的可移植性高	6	1.5.4 return 语句	19
1.2.4 灵活的流程控制	6	1.6 综合范例程序	19
1.3 Dev-C++ 的下载与简介	6	学生学籍信息的输出程序	
1.3.1 Dev-C++ 下载过程	6	范例程序: CH01_02.c	20
1.3.2 认识 Dev-C++ 工作环境 ...	10	本章重点回顾	20
1.4 Hello ! 我的第一个 C 程序 ...	11	课后习题	21

第

2

章

C 语言的数据处理

2.1 变量	24	2.1.2 变量的命名规则	27
2.1.1 变量声明	24	2.1.3 sizeof 运算符	28
变量声明的示范		sizeof 运算符的应用	
范例程序: CH02_01.c	25	范例程序: CH02_02.c	29

2.1.4 常数	30	2.3.1 printf() 函数	45
计算圆面积		格式化输出的示范	
范例程序: CH02_03.c	31	范例程序: CH02_10.c	47
2.2 基本数据类型	32	八进制数与十六进制数表示法	
2.2.1 整数类型	32	范例程序: CH02_11.c	48
整数修饰词综合范例		2.3.2 格式化字符的高级设置 ...	49
范例程序: CH02_04.c	34	格式化字符的高级设置	
溢出输出效应		范例程序: CH02_12.c	50
范例程序: CH02_05.c	35	2.3.3 scanf() 函数	51
2.2.2 浮点数类型	36	scanf() 函数输入数据	
单精度与双精度浮点数		范例程序: CH02_13.c	52
范例程序: CH02_06.c	38	2.4 综合范例程序 1——	
浮点数科学记数表示法		成绩统计小帮手	53
范例程序: CH02_07.c	39	成绩统计小帮手	
2.2.3 字符类型	40	范例程序: CH02_14.c	54
字符声明的示范		2.5 综合范例程序 2——	
范例程序: CH02_08.c	41	转义字符序列的应用	55
2.2.4 转义序列	43	转义字符序列的应用	
转义字符的简单应用		范例程序: CH02_15.c	55
范例程序: CH02_09.c	44	本章重点回顾	56
2.3 格式化输入与输出功能	45	课后习题	57

第 3 章 活用表达式与运算符

3.1 运算符的简介	60	关系与逻辑运算符的求值	
3.1.1 赋值运算符	60	范例程序: CH03_03.c	66
3.1.2 算术运算符	61	3.1.5 递增与递减运算符	67
查询百位数		递增与递减运算符的实际应用范例	
范例程序: CH03_01.c	62	范例程序: CH03_04.c	69
3.1.3 关系运算符	63	3.1.6 位运算符	70
关系运算符运算的示范		位运算符的综合运用	
范例程序: CH03_02.c	63	范例程序: CH03_05.c	74
3.1.4 逻辑运算符	64	3.1.7 复合赋值运算符	75

复合赋值运算符的实际应用

范例程序: CH03_06.c 76

3.1.8 条件运算符 77

条件运算符的使用

范例程序: CH03_07.c 78

3.2 运算符的优先级 79

运算符优先权的实际运用

范例程序: CH03_08.c 80

类型转换的范例

范例程序: CH03_09.c 82

3.3 综合范例程序 1——

钞票兑换机 84

钞票兑换机

范例程序: CH03_10.c 84

3.4 综合范例程序 2——

温度转换器 85

温度转换器

范例程序: CH03_11.c 85

本章重点回顾 86

课后习题 87

第 4 章 选择性流程控制

4.1 结构化程序设计 90

4.2 选择结构 92

4.2.1 if 条件指令 93

消费满额赠送来店礼品

范例程序: CH04_01.c 94

4.2.2 if else 条件语句 96

奇偶数判断器

范例程序: CH04_02.c 97

4.2.3 if else if 条件语句 99

消费金折扣回馈

范例程序: CH04_03.c 100

阶梯电价查询程序

范例程序: CH04_04.c 102

4.2.4 switch 选择语句 104

快餐店点餐程序

范例程序: CH04_05.c 106

分数段判断

范例程序: CH04_06.c 108

4.3 综合范例程序 1——

闰年计算器 110

闰年计算器

范例程序: CH04_07.c 110

4.4 综合范例程序 2——

简易计算器的制作 111

简易计算器的制作

范例程序: CH04_08.c 111

本章重点回顾 113

课后习题 113

第 5 章 循环流程控制

5.1 for 循环	116	5.3.2 continue 指令	129
数字累加计算		continue 指令的应用	
范例程序: CH05_01.c	118	范例程序: CH05_07.c	130
九九乘法表		5.3.3 goto 指令	132
范例程序: CH05_02.c	120	goto 指令的应用	
5.2 while 循环	121	范例程序: CH05_08.c	132
正因数分解		5.4 综合范例程序 1——	
范例程序: CH05_03.c	122	求解最大公约数	134
判断循环执行次数		求解最大公约数	
范例程序: CH05_04.c	124	范例程序: CH05_09.c	134
数字反向输出		5.5 综合范例程序 2——	
范例程序: CH05_05.c	126	密码验证器	135
5.3 流程跳离指令	127	密码验证器	
5.3.1 break 指令	128	范例程序: CH05_10.c	136
break 指令的应用		本章重点回顾	137
范例程序: CH05_06.c	128	课后习题	138

第 6 章 数组与字符串

6.1 数组简介	141	二阶行列式	
6.1.1 一维数组	141	范例程序: CH06_04.c	150
累加数组元素		6.1.3 多维数组	152
范例程序: CH06_01.c	144	三维数组的应用范例	
学生成绩的输入与计算程序		范例程序: CH06_05.c	153
范例程序: CH06_02.c	145	6.2 字符串	154
6.1.2 二维数组	147	6.2.1 字符串声明	155
二维数组的应用		字符与字符串的比较	
范例程序: CH06_03.c	148	范例程序: CH06_06.c	156

6.2.2 字符串数组 157

字符串数组的应用

范例程序: CH06_07.c 158

字符串数组与学生成绩的计算

范例程序: CH06_08.c 160

6.3 综合范例程序 1——

冒泡排序法 161

冒泡排序法

范例程序: CH06_09.c 164

6.4 综合范例程序 2——

字母大小写转换器 165

字母大小写转换器

范例程序: CH06_10.c 165

本章重点回顾 166

课后习题 167

第 7 章 函数

7.1 函数简介 170

“Hello! 我是函数!”

范例程序: CH07_01.c 170

7.1.1 函数的原型声明 172

省略函数原型声明的示范

范例程序: CH07_02.c 172

7.1.2 函数主体的定义 174

7.1.3 函数调用 174

数字比大小

范例程序: CH07_03.c 175

7.2 参数的传递方式 176

7.2.1 传值调用 177

传值调用的范例

范例程序: CH07_04.c 177

7.2.2 传址调用 179

传址调用的应用

范例程序: CH07_05.c 180

7.3 数组参数的传递 181

7.3.1 一维数组传递 182

一维数组与参数传递

范例程序: CH07_06.c 183

7.3.2 多维数组传递 184

二维数组与参数传递

范例程序: CH07_07.c 185

求三维数组中的最小值

范例程序: CH07_08.c 187

7.4 递归函数 189

求解 n 阶乘的函数

范例程序: CH07_09.c 190

7.5 综合范例程序 1——

汉诺塔游戏 192

汉诺塔游戏

范例程序: CH07_10.c 192

7.6 综合范例程序 2——万年历 ... 194

万年历

范例程序: CH07_11.c 194

本章重点回顾 196

课后习题 197

附录 习题答案 199

第 1 章

C 语言初步体验

本章重点

- | | |
|-------------------|--------------------|
| 1. 程序设计语言发展的历史 | 4. 认识 Dev-C++ 工作环境 |
| 2. C 语言的特色 | 5. 编写第一个 C 程序 |
| 3. Dev-C++ 的下载与简介 | |

C 语言称得上是一种历史悠久的高级程序设计语言，也往往是现代程序设计初学者最先接触的程序设计语言之一。C 语言对近代的计算机科学领域有着非凡的贡献。C 语言的前身是 B 语言，最早是在 1972 年时由贝尔实验室的 Dennis Ritchie 博士在 PDP-11 计算机的 UNIX 操作系统上发展出来的。

设计 C 语言的最初目的主要是作为开发 UNIX 操作系统的工具，由于 C 语言推广相当成功，使得 UNIX 操作系统开发难度降低且进行顺利，同时加入了数据类型概念以及其他函数功能，因而广泛应用于其他的程序设计领域。

1.1 认识程序设计语言



程序设计语言发展的历史已有半世纪之久，种类还不少，如果包括实验、教学或科学研究的相关用途，问世的程序设计语言可能有上百种之多，由最早期的机器语言发展至今，已经迈入第五代自然语言，不过每种程序设计语言都有其发展的背景及目的。

程序设计语言就是一种人类用来和计算机沟通的语言，也是用来指挥计算机运算或工作的指令集合，可以将人类的思考逻辑转换成计算机能够了解的语言。每一代的语言都有其特色，无论任何一种语言都有其专用的语法、特性、优点及相关应用的领域。按照其发展演变过程，分类如图 1-1 所示。



图 1-1 程序设计语言的发展史

1.1.1 机器语言

机器语言 (Machine Language) 是最早期的程序设计语言，任何程序在执行前都必须被转换为机器语言，由 1 和 0 两种符号构成。机器语言写法如下：

```
10111001 (设置变量 A)  
00000010 (将 A 设置为数值 2)
```


不过每一家计算机制造商，往往因为计算机硬件设计的不同而开发不同的机器语言。这样不但使用不方便，可读性低，也不易于维护，并且不同的机器平台，编码方式也不尽相同。

1.1.2 汇编语言

汇编语言（Assembly Language）指令比机器码指令看起来稍有“意义”一些，但仍然与机器语言是一一对应的对应关系，因此与机器语言一样被归类为低级语言，只是它在编写上比机器语言容易多了。

每一种系统的汇编语言都不一样，就PC而言，用的是80×86的汇编语言。例如，MOV指令代表设置变量内容、ADD指令代表加法运算、SUB指令代表减法运算，汇编语言写法范例如下：

```
MOV  A , 2    ( 变量 A 的数值内容为 2 )  
ADD  A , 2    ( 将变量 A 加上 2 后，将结果再存回变量 A 中，如 A=A+2 )  
SUB  A , 2    ( 将变量 A 减掉 2 后，将结果再存回变量 A 中，如 A=A-2 )
```

1.1.3 高级语言

对一般人来说，纯粹用汇编语言完成一个程序仍然是一件相当困难的事情。所谓高级语言，就是比汇编语言的语句更容易看懂的程序设计语言。高级语言的指令和语句都更接近日常生活中常使用的文字或符号，我们编程时所需要做的就是变量声明，以及程序流程的控制。例如，Fortran语言是世界上第一个开发成功的高级语言，更是历久弥新，现在仍有许多研究机构用来解决工程与科学上的问题。或者早期非常流行的BASIC语言，不但易学易懂，非常适合于初学者了解程序语言的工作方式，甚至间接带动了当年PC使用的风潮，目前最为流行的高级语言有C、C++、Java、Visual Basic或Python语言，尤其是堪称常青树的C语言，算得上是近三十年来最为出色的高级语言。

1.1.4 非过程性语言

“非过程性语言”（Non-procedural Language）也称为第四代语言（Fourth Generation Language, 4GL），英文简称为 4GLS，特点是编程者不必描述数据存储的细节，只需要将步骤写出来，且不必管计算机要如何去执行，也不需要去理解计算机的执行过程，这种语言减轻了用户设计程序的负担，例如数据库的结构化查询语言（Structural Query Language, SQL）就是一种第四代语言，SQL 语言写法范例如下：

```
DELETE FROM employees  
WHERE employee_id = 'C800312' AND dept_id = 'R01';
```

1.1.5 人工智能语言

人工智能语言称为第五代语言，或称为自然语言（Natural Language），它是程序设计语言发展的终极目标，为用户提供以一般人类语言的语句直接和计算机进行对话，向计算机发出问题，而不必考虑程序的语法与规则，所以自然语言必须有人工智能（Artificial Intelligence, AI）技术的发展作为保障。

1.2 C 语言的特色



C 语言是一种相当灵活、轻盈且历史悠久的程序设计语言，不但是现代程序设计领域初学者最先接触的程序设计语言之一，也深受全世界专业程序设计人员所喜爱。由于许多平台上都可以运行 C 的程序，各家厂商所出品的 C 开发工具时常融入不同的特性与特殊语法，而这往往会增添程序设计人员在开发上的困扰。

因此在 20 世纪 80 年代初，美国国家标准协会（American National Standard Institution, ANSI）特别为 C 制定了一套完整的国际标准，称为 ANSI C，作为 C 语言的标准，即标准 C。因此，我们目前只要使用符合

ANSI C 格式的标准 C 的语法，即可在各个平台上通行无阻了。为什么 C 能有如此屹立不动的地位呢？我们可以简单归纳出以下四项特点。

1.2.1 硬件沟通能力

C 语言经常被程序员们称为“中级语言”，原因是 C 语言不但具有高级语言的亲和力，容易开发、阅读、调试与维护，而且在 C 的程序代码中允许开发者加入低级的汇编程序，使得 C 程序更能够直接控制与存取硬件系统，而且一直都是开发操作系统的主力底层语言，例如连单芯片（如 8051）、嵌入式系统或硬件驱动程序的开发，也都可以使用 C 语言来设计。

1.2.2 高效率的编译型语言

任何程序编写的目的，都是为了执行的结果，因此都必须转换成机器语言才能正确执行，从程序设计语言转换的方式来看，可以分成编译型语言与解释型语言两种。

1. 编译型语言

编译型语言在程序开始执行前，必须使用编译器（compiler）来将源代码程序转换为机器可读的可执行文件或目标程序，不过编译器必须先把源代码程序读入主存储器后才可以开始编译，编译后的目标程序（object file）可直接对应成机器码，故可在计算机上直接执行，不需要每次执行都重新“翻译”，执行速度自然较快。例如 C、C++、PASCAL、FORTRAN、Java 语言都属于编译型语言。

2. 解释型语言

解释型语言在程序开始执行前，源代码程序可以通过解释器（Interpreter）将程序一行接一行地读入，逐行“解释翻译”并交由计算机执行，解释的过程中如果发生错误，就会立刻停止，不会产生目标文件或可执行文件。由于每次执行时都必须再“解释”一次，因此执行速度较慢，效率也较低，例如 Basic、LISP、Prolog、Python 等语言都采用解释执行的方法。

1.2.3 程序的可移植性高

C 语言具备了相当强的可移植性（portability），就好比硬件的高兼容性，也就是使用 ANSI C 函数库所编写的程序，只要程序代码稍作修改就能立刻搬到其他操作系统上执行，许多计算机及操作系统平台上基本都开发和提供了 C 的编译器，例如 MS-DOS、Windows 系列、UNIX/Linux 操作系统，甚至 Mac 系统等。

1.2.4 灵活的流程控制

C 的语法不但严谨简洁，而且在设计上具有高级语言的结构化流程控制与模块化特性，更可以使用函数（function）与运算符（operator）来增加程序代码的可读性，还具有功能强大的函数库（library），从而节省了程序设计人员重新编写程序代码的重复工作，同时也让程序代码较容易调试和维护。

1.3 Dev-C++ 的下载与简介



早期程序设计人员如果要着手开始设计 C 程序，则首先必须找一种文本编辑器（例如 Windows 系统下的记事本）来编辑程序代码，接着选择一种 C 的编译器（如 Turbo C/C++、gcc 等）来编译并执行 C 程序。不过，现在不用这么麻烦了，只要找个可将程序的编辑、编译、执行与调试等功能集于一体的同一操作环境，即所谓的“集成开发环境”（Integrated Development Environment, IDE）。由于 C 语言受到各界的欢迎，市场上有许多家厂商陆续开发了许多 C 语言的 IDE，如果大家是 C 的初学者，又想学好 C 语言，那么免费的 Dev-C++ 肯定是一个不错的选择。

1.3.1 Dev-C++ 下载过程

原本的 Dev-C++ 已停止开发，改为发行非官方版，Owll Dev-C++ 是一个功能完整的集成开发环境（含编译器），是开放源码（open-source code）

的软件，专为 C/C++ 语言所设计。在这个环境中，我们可以轻松编写、编辑、调试和执行 C 语言的种种功能。要安装 Dev-C++ 软件，可从如下网址自行下载（参见图 1-2）最新的版本：

http://sourceforge.net/projects/orwelldevcpp/?source=typ_redirect

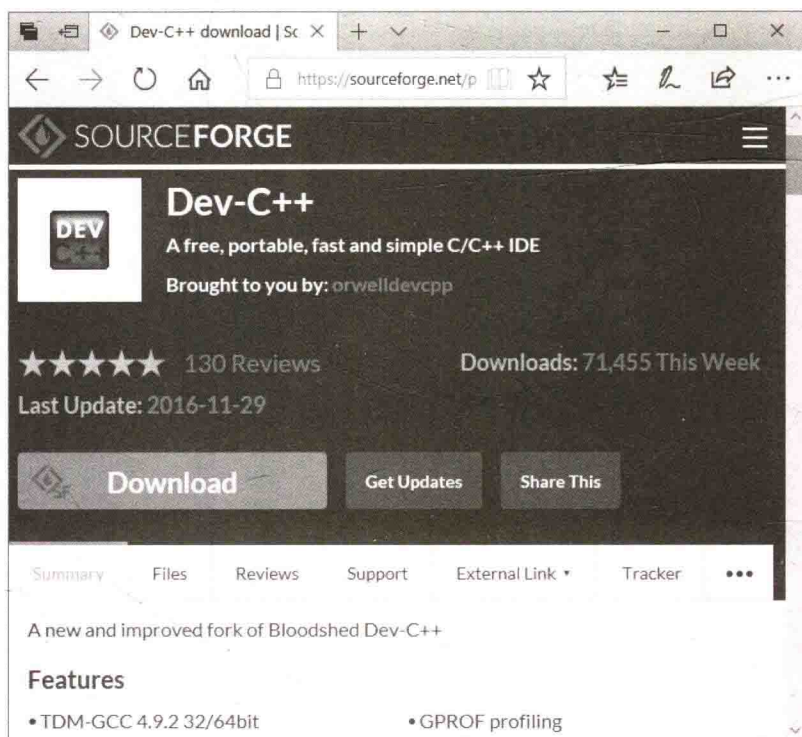


图 1-2

在我们下载好了“Dev-Cpp 5.11 TDM-GCC 4.9.2 Setup.exe”安装程序之后，就可以在所下载的文件夹中用鼠标左键双击以启动这个安装程序。安装程序启动后，首先会要求选择语言，此处先选择“English”，如图 1-3 所示。

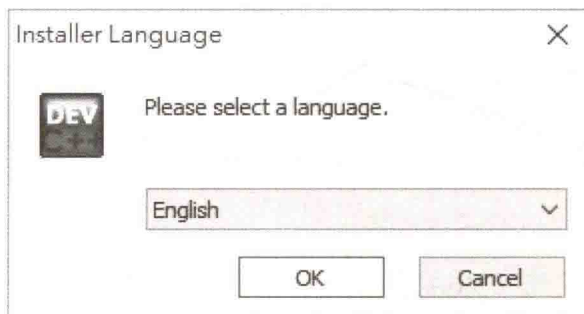


图 1-3