



王瑞 王舵 编著

C++

教学做一体化教程

程序设计

清华大学出版社



The cover features a large, bold, black 'C++' logo. The 'C' is on the left, and the two '+' signs are to its right. A small black tag is attached to the bottom of the 'C'. The background is white, with black silhouettes of leaves in the top-left and bottom-right corners.

王瑞 王舵 编著

教学做一体化教程

程序设计

清华大学出版社
北京

内 容 简 介

C++ 不仅支持 C 语言的几乎全部功能,还提供了数据抽象和面向对象的技术,更提供模板技术来提高运行效率。通过学习 C++ 可以快速过渡到面向对象编程,能够进行真正的软件开发。

本教材采用教、学、做一体化模式,以核心知识、能力目标、任务驱动和实践环节为单元组织本教材的体系结构。每章都由核心知识、能力目标、任务驱动和实践环节 4 个模块所构成。在语法上严格遵守 C++ 2011 标准,在程序设计思想方面强调模块化思想,在克服难点方面注重结构清晰地安排内容,循序渐进地展开知识,特别强调知识点的能力目标,通过合理的任务驱动和实践环节提高程序的设计能力和综合运用知识的能力。全书分为 10 章,分别讲解了初识 C++,数据类型,表达式与语句,函数,类,类与对象,单继承与组合,多继承与多态,运算符重载,模板。

本书不仅适合作为高等院校理工类学生学习 C++ 程序设计的教材,而且特别适合作为教、学、做一体化的教材。

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++ 程序设计教学做一体化教程/王瑞,王舵编著. —北京:清华大学出版社,2013.4

ISBN 978-7-302-31085-3

I. ①C… II. ①王… ②王… III. ①C 语言—程序设计—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 303599 号

责任编辑:田在儒 张 弛

封面设计:李 丹

责任校对:袁 芳

责任印制:杨 艳

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

课件下载: <http://www.tup.com.cn>, 010-62795764

印 装 者:北京国马印刷厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:14.75

字 数:356 千字

版 次:2013 年 4 月第 1 版

印 次:2013 年 4 月第 1 次印刷

印 数:1~5000

定 价:29.00 元

产品编号:045587-01

前言 FOREWORD

本教材是作者多年讲授 C++ 语言的结晶,每章的核心知识点模块强调在编程中最重要和实用的知识点,其中的简单示例起着帮助读者理解和掌握核心知识的作用;能力目标强调使用核心知识进行编程的能力;需要完成的任务中的任务模板起着训练编程能力的作用,其中的任务小结主要总结任务中涉及的重要技巧、注意事项以及扩展知识,通过任务模板的训练,读者有能力完成后续的实践环节。第 1 章主要讲解开发 C++ 程序的基本步骤,读者可以迅速开发出第一个简单的 C++ 程序,并充分认识到 C++ 语言的特点。第 2 章是数据类型,在核心知识和任务的安排方面特别注重训练读者应当掌握和理解基础知识。第 3 章是表达式与语句,在任务安排上注重结合实际问题,训练读者熟练地计算各种表达式、识别各种语句和它们的作用。第 4 章介绍函数,C++ 几乎保持 C 语言的全部面向过程编程技术,只有些微改变,在任务安排上注重结合实际问题训练读者熟练使用函数。第 5 章是本书的重点内容之一,讲述类的基本知识以及成员的声明和定义,强调特殊类型函数成员的使用,为了实现能力目标,特别注重选择有启发的例子和任务,以此训练读者使用所学知识解决实际问题的能力。第 6 章讲述对象的创建和使用,注重强调类中构造函数的调用。第 7 章是本书的难点之一,继承是面向对象语言最强大的特性,本章特别训练不同类型继承的特点、如何使用继承和组合。第 8 章讲述继承中的多态特性,强调如何根据对象的不同调用不同的函数体。第 9 章讲述运算符作为普通函数进行重载,强调运算符重载时要注意的各种问题。第 10 章讲述模板,它是通用代码不受数据类型的影响,并且可以根据要处理的对象自动适应数据类型的变化。

本教材特别注重引导学生参与课堂教学活动,适合高等院校相关专业作为教、学、做一体化的教材。

本教材的示例和任务模板的源程序以及电子教案可以在清华大学出版社网站上免费下载,以供读者和教学使用。

编者

2013 年 3 月

目 录

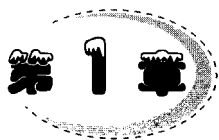
CONTENTS

第 1 章 初识 C++	1
1.1 引言	1
1.2 开发环境	1
本章小结	4
习题 1	4
第 2 章 数据类型	6
2.1 整数类型	6
2.2 浮点类型	9
2.3 布尔型	11
2.4 枚举类型	13
2.5 指针	15
2.6 数组	17
2.7 结构体	20
2.8 引用类型	23
2.9 const 修饰符	26
2.10 字符数组	28
2.11 string 类型之一	31
2.12 string 类型之二	34
本章小结	37
习题 2	38
第 3 章 表达式与语句	40
3.1 算术运算符	40
3.2 关系和逻辑运算符	43
3.3 位运算	44
3.4 逗号运算符和条件运算符	46
3.5 类型转换	48
3.6 条件语句	50

3.7 循环语句.....	54
3.8 转移语句.....	58
本章小结	61
习题 3	62
第 4 章 函数	64
4.1 main 函数	64
4.2 函数原型.....	65
4.3 函数定义.....	67
4.4 函数传值调用.....	68
4.5 函数传址调用.....	71
4.6 递归调用.....	74
4.7 函数重载.....	75
4.8 变量.....	78
本章小结	81
习题 4	82
第 5 章 类	84
5.1 类的定义.....	84
5.2 控制访问成员.....	87
5.3 函数成员的特性	91
5.4 特殊函数成员.....	95
5.5 const 修饰数据成员	97
5.6 const 修饰函数成员	99
5.7 this 指针	103
5.8 static 修饰数据成员	104
5.9 static 修饰函数成员	107
5.10 友元.....	109
本章小结.....	111
习题 5	112
第 6 章 类与对象	115
6.1 对象	115
6.2 对象的创建和撤销	118
6.3 默认构造函数	120
6.4 复制构造函数	121
6.5 构造函数重载	125

6.6 对象数组	127
6.7 new 和 delete	130
本章小结	133
习题 6	135
第 7 章 单继承与组合	140
7.1 基类和派生类	140
7.2 继承	143
7.3 组合	148
7.4 protected 成员	150
7.5 派生类对象的构造	152
7.6 函数覆盖	156
7.7 向上类型转换	158
本章小结	161
习题 7	162
第 8 章 多继承与多态	168
8.1 多继承	168
8.2 虚基类	172
8.3 多态与虚函数	174
8.4 纯虚函数	179
8.5 虚析构函数	182
本章小结	185
习题 8	185
第 9 章 运算符重载	190
9.1 运算符函数	190
9.2 重载运算符=	193
9.3 重载运算符+和-	197
9.4 重载关系运算符	199
9.5 重载运算符++和--	201
9.6 重载运算符[]	204
9.7 重载运算符()	206
本章小结	209
习题 9	209
第 10 章 模板	212
10.1 模板函数	212

10.2 函数模板重载.....	215
10.3 函数模板举例.....	217
10.4 类模板.....	219
10.5 类模板举例.....	222
本章小结.....	224
习题 10	225



初识 C++

1.1 引 言

C 语言由贝尔实验室的 Dennis Ritchie 设计,于 1972 年实现。1983 年美国国家标准化委员会和信息处理分会创立了 X3J11 技术委员会,以提供该语言与机器无关的明确含义。1989 年,该委员会制定了 C 语言的标准(ANSI C),并获得批准。1999 年,该标准被更新。

贝尔实验室的 Bjarne Stroustrup 对 C 语言做了扩充,于 1980 年设计并开发出 C++ 语言。C++ 语言提供许多新功能,使 C 语言更加有条理,更重要的是,它提供了面向对象编程的功能。1985 年后,C++ 语言开始成为行业和大学占统治地位的一种语言并逐渐形成多种风格。1998 年国际标准化组织(International Standards Organization,ISO)完成对 C++ 语言的标准化后,所有的编译器都与标准兼容,并形成统一的 C++ 标准库,自此,该语言正式成为一门成熟的面向对象编程语言。

软件行业中,面向对象编程是一项技术革命。据研究发现,使用模块化面向对象的设计和实现方法,使得软件开发人员工作效率比采用结构化编程技术快 10~100 倍。

就程序员而言,C 程序员将注意力集中在编写函数上。执行某一任务的一组操作将构成函数,函数又构成程序。数据在 C 语言中主要是为了支持函数。C++ 程序员的主要工作在于创建用户自定义的数据类型——类。类的数据部分叫做数据成员,函数部分叫做成员函数。

面向对象技术的核心就是把数据(属性)和函数(行为)封装到程序包(类)中。类就像建造房屋的蓝图,利用它,建筑者可以建起许多房屋;利用类的定义,也可以建立相同类的许多变量(对象)。程序员只要将注意力集中在设计类上,然后在合理位置应用对象,就可以极大提高效率。例如,组装电脑时,用户并不需要知道声卡(一个类)内部的工作原理(类内部实现细节),更不会用原始的集成芯片和材料去制作声卡(类中数据和函数具体的实现方法),只需将一块声卡安装到合适的位置就可以了(使用声卡类的一个对象)。

1.2 开发环境

核心知识

C++ 程序中包含变量和函数。函数由变量声明和执行语句组成。下面是一个简单的 C++ 程序,仅用来输出一个字符串。

```
#include <iostream.h>
int main()
{
    cout<<"这是我的第一个 C++ 程序"<<endl;
    //在屏幕上输出“这是我的第一个 C++ 程序”
    return 0;
}
```

上面程序的运行结果是：

这是我的第一个c++程序

- `#include <iostream.h>` 该语句的功能是进行编译预处理。`<iostream.h>` 是 C++ 提供的标准库头文件,其中包含了输入输出的类、函数和全局变量等。`#include` 指令后面的文件名使用尖括号“`<>`”括起来,表示该文件由编译系统提供;文件名如果使用双引号括起来,表明用户想使用自己的头文件。
- `int main()` 该语句表明 `main` 函数没有返回类型。每个 C++ 程序都有一个主函数 `main`,它是程序的入口和出口。C++ 程序从 `main` 函数的第一条语句开始执行,顺序执行完所有语句之后,程序结束。
- `cout<<"这是我的第一个 C++ 程序"<<endl;` 该语句告诉编译器在屏幕上显示字符串。
- `//在屏幕上输出“这是我的第一个 C++ 程序”` 注释就是对程序的一个说明和解释,本身对程序不产生任何影响。C++ 语言的注释以“`//`”开始,直到该行结束,当然,它还同时支持 C 语言的注释风格: `/* 注释 */`。

能力目标

能够声明变量;
能够使用 `cin` 和 `cout` 进行输入和输出;
理解函数的作用。

任务驱动

(1) 任务的主要内容

编写一个 C++ 程序,在 `main` 函数中进行如下操作。

- 声明 `int` 类型的变量 `c`, `thisvariable`, `q789123` 和 `number`。
- 提示用户输入一个整数。
- 把键盘输入的整数为变量 `number` 赋值。
- 判断 `number` 是否等于 20: 如果等于 20,输出“`number` 等于 20”;如果不等于 20,则输出“`number` 不等于 20”。
- 在程序末尾输出“这是我参与的第一个 C++ 程序!”。

(2) 任务的代码

根据任务内容的提示完成下面的程序,程序的运行结果如下:

```
从键盘输入一个整数,给number赋值
234
number不等于20
这是我参与的第一个C++程序:
```

```
#include <iostream.h>
//定义 main 函数
int main()
{
    【代码 1】:    //声明 int 类型的变量 c,thisvariable,q789123 和 number
    cout<<"从键盘输入一个整数,给 number 赋值"<<endl;
    【代码 2】:    //从键盘输入一个整数,为变量 number 赋值
    if(number==20) //判断 number 是否等于 20
        cout<<"number 等于 20"<<endl;    //如果 number 等于 20 执行这条语句
    else          //如果 number 不等于 20 时
        cout<<"number 不等于 20"<<endl;
    【代码 3】:    // 输出“这是我参与的第一个 C++ 程序!”
    return 0;
}/* main 函数定义结束 */
```

(3) 任务小结或知识扩展

- C++ 程序的开发过程一般经过:编辑、编译、连接和运行四个步骤。编辑是指用户在开发环境中输入源代码,一般保存为 *.cpp 文件。该文件通过编译,将源代码生成机器语言指令构成的目标文件。由于目标文件只是相对独立的程序块,需要通过连接将其转化为可执行程序。
- 一个 C++ 程序由一个主函数和若干个函数组成。一个函数必须在声明之后才能使用(被调用)。
- C++ 中函数的声明原则是:函数定义在先,调用在后,调用前不必声明;函数调用在先,定义在后,调用前需要声明。也就是说函数定义已经包含函数声明。
- C++ 程序中唯一不能定义函数的地方是另一个函数定义体内。
- C++ 中任何函数都可以被包括自己的函数调用。

(4) 代码模板的参考答案

```
【代码 1】: int c,thisvariable,q789123,number;
【代码 2】: cin>>number;
【代码 3】: cout<<"这是我参与的第一个 C++ 程序!"<<endl;
```

实践环节

写一个 C++ 程序,计算从 $1+2+3+4+\dots+100$ 的值。请根据注释补充程序。该程序的运行结果如下。

```
1+2+3+...+100=5050
```

```
#include <iostream.h>
//定义 main 函数
int main()
{
```

```

【代码 1】:      //声明变量 x 和 sum 为 int 型
【代码 2】:      //将变量 sum 初始化为 0
                x=100;
                for(int i=1;i<=x;i++)
【代码 3】:      //将变量 i 和变量 sum 相加,并把结果赋值给 sum
                cout<<"1+2+3+...+100="<<sum<<endl;
                return 0;
}/* main 函数定义结束 */

```

本章小结

- C++ 程序一般经过编辑、编译、连接以及运行四个阶段。
- C++ 程序的文件扩展名一般是 .cpp 或 .c。
- C++ 程序中通过 # include <iostream.h> 告诉编译器把输入和输出流包括到文件中。其中 cin 称为标准输入流,一般从键盘接收数据;cout 称为标准输出流,通常是向屏幕输出。
- C++ 程序中首先执行的是 main 函数。
- 以双斜杠(//)开始注释,然后描述代码的功能,使用户能够更好地理解这些代码。

习 题 1

1. 每个 C++ 程序都是从_____函数处开始执行。
2. 每个 C++ 语句都是以_____结束。
3. _____用于说明程序,并提高程序的可读性。
4. 将 $y=ax^3+7$ 用 C++ 语句表达,下面哪个选项是正确答案? ()
 - A. $y=axxx+7$;
 - B. $y=a * x * x * (x+7)$;
 - C. $y= a * x * (x * x+7)$;
 - D. 不能用 C++ 语句表达
5. 编写一个程序,完成两个整数的加法。具体要求如下:
 - 声明三个整型变量 a、b 和 c;
 - 提示用户输入整数;
 - 使用键盘输入为变量 a、b 赋值;
 - 用表达式 $a+b$ 的结果为变量 c 赋值;
 - 输出 c 的值。

```

#include <iostream.h>
int main()
{

```

```

    【代码 1】:      //声明三个整型变量 a、b 和 c
    cout<<"请输入两个整数"<<endl;

```

```

    【代码 2】:      //使用键盘输入的数为变量 a 赋值

```

```
cin>>b;           //使用键盘输入的数为变量 b 赋值
【代码 3】:       //用表达式 a+b 的结果为变量 c 赋值
cout<<"a+b="<<c<<endl; //输出 c 的值
}
```

6. 编写一个 C++ 程序,在 main 函数中提示用户输入圆的半径,调用子函数 CircleArea 计算圆的面积并返回给 main 函数。

```
#include <iostream.h>
#define pi 3.1415           //使用宏替换
double CircleArea(double); //CircleArea 函数圆形说明
/* 定义 main 函数 */
int main()
{
    【代码 1】:           //声明 double 类型变量 r 和 area
    cout<<"请输入圆的半径"<<endl; //提示用户输入
    【代码 2】:           //接收键盘输入并为 r 赋值
    area=CircleArea(r);     //调用子函数 CircleArea
    cout<<"圆的面积为"<<area<<endl;
    return 0;
}/* main 函数定义结束 */

/* 定义子函数 CircleArea */
double CircleArea(double r)
{
    return pi * r * r;
}/* 子函数 CircleArea 定义结束 */
```

7. 根据下面的要求编写 C++ 语句。

- 声明 int 类型的四个变量 x、y 和 z 以及 result;
- 提示用户输入 3 个整数;
- 接收键盘输入的 3 个数,分别为 x、y 和 z 赋值;
- 计算 x、y 和 z 的乘积,并赋值给变量 result;
- 输出“x、y 和 z 的乘积结果是”,后面输出 result 的值。

数据是程序操作的对象。它们具有一定的数据名称、数据类型等。其中数据名称是为某一数据指定的标识符,数据类型确定该数据操作时具体的范围。

C++ 程序中最基本的元素是数据类型。数据类型通常具有下面的三种作用:限定数据的取值范围;定义数据类型的运算集合以及规定该数据类型变量占用的内存大小。

2.1 整数类型

核心知识

字符型(char)、短整型(short int)、整型(int)和长整型(long int)称为整数类型。整数类型可以带符号,也可以不带符号,通常默认类型是带符号。带符号的整数类型最左边的位是符号位,其他的是数值;无符号类型中,所有的位都是数值。例如,带符号 short,表示数的范围是-32 768 到 32 767;如果是无符号 short,范围是 0 到 65 535。

(1) 字符型(char)

- 字符常量是用单引号引起来的字符。常量: 'a', 123, '\n'(转义字符), '\x3f'。
- char 定义的变量只能存储单个字符,而不能存储字符串。变量: char ch=12;。
- 字符型数据使用 8 位 ASCII 码表示。
- 有符号的 char 代表从-128 到 127 之间的数值;无符号 char 表示从 0 到 255 之间的数值。

(2) short 型

short 型变量在内存中占 2 个字节。

(3) int 型

- 常量: 15(十进制), 076(八进制), 0x17fc(十六进制)。
- 变量: 变量必须先声明后使用。

```
int x;  
x=1786;
```

- int 型变量在内存中占 4 个字节。

(4) long 型

- 常量: 5 238 898, 0x4fea2。

- 变量: long x=6 553 554;。
- long 型变量在内存中占 4 个字节。

能力目标

能够声明整型变量,并赋予初值。

了解 char、short、int 和 long 型变量的取值范围。

任务驱动

任务一

(1) 任务的主要内容

编写一个程序,完成下面的功能要求。

- 声明变量 c1、c2 和 c3 以及 TAB。c1 的初始值为'A'(其 ASCII 码值为 65);
- 输出 c2 的值为'a'(其 ASCII 码值为 97),用 c1 的表达式给 c2 赋值;
- TAB 的初始值是水平制表符;
- 输出 c3 时,会有一声铃响,给 c3 赋初始值。

(2) 任务的代码

根据“任务内容”的提示完成下面的程序,程序的运行结果如下:

```
#include <iostream.h>
int main()
{
    【代码 1】: //声明变量 c1、c2、c3 和 TAB。c1 的初始值为'A' (其 ASCII 码值为 65)
    【代码 2】: //c2 的是'a' (其 ASCII 码值为 97)
    cout<<c2<<endl;
    【代码 3】: //TAB 的初始值是水平制表符
    cout<<"哦"<<TAB<<"你也在这里吗?"<<endl;
    cout<<"想做的事情总找得出时间和机会;不想做的事情总找得出借口。"<<"\n";
    //'\n'是字符常量,表示回车换行
    【代码 4】: //输出 c3 时,会有一声铃响,给 c3 赋初始值
    cout<<"程序就要结束了,会有一声铃响,听听看"<<c3<<endl;
    return 0;
}
```

(3) 任务小结或知识扩展

- 字符型数据使用 8 位 ASCII 码表示。字符型变量在内存中以整数存储,能够参与整数类型的操作。
- 在 C++ 中定义了一些以“\”开头的字符序列,作为特殊字符常量,例如\n表示换行,\a表示响铃。
- short int 类型数据的最大表示数为 32767,当该类型变量的赋值超过这个范围,称为溢出,得到的将是一个负数。

(4) 代码模板的参考答案

【代码 1】: `char c1='A',c2,c3,TAB;`

【代码 2】: `c2=c1+32;`

【代码 3】: `TAB='\t';`

【代码 4】: `c3='\a';`

任务二

(1) 任务的主要内容

下面是一个简单的加密解密程序。用户输入需要加密的数值 `number`,称为“明文”。程序将数值乘以 7 加 5 进行加密得到“密文”并显示;最后将密文进行解密。具体要求如下:

- 声明 `int` 类型的变量 `number`、`secret` 和 `encrypt`;
- 提示用户从键盘输入一个数值,并为 `number` 赋值;
- 计算加密之后的值,并赋值给变量 `secret`;
- 输出密文;
- 对密文进行解密(算法是 `secret` 的值减去 5 再除以 7),并将结果赋值给变量 `encrypt`;
- 输出 `encrypt` 的值。

(2) 任务的代码

根据“任务内容”的提示完成下面的程序,程序的运行结果如下:

```
输入一个整数为变量number赋值: 1298
number的密文是9891
对密文进行解密, 结果是:1298
```

```
#include<iostream.h>
int main()
{
    【代码 1】:    //声明 int 类型的变量 number、secret 和 encrypt
    cout<<"输入一个整数为变量 number 赋值: ";
    【代码 2】:    //使用键盘输入为变量 number 赋值
    【代码 3】:    //对 number 进行加密
    cout<<"number 的密文是"<<secret<<endl;
    【代码 4】:    //将密文 secret 进行解密
    cout<<"对密文进行解密,结果是:"<<encrypt<<endl;
    return 0;
}
```

(3) 任务小结或知识扩展

C++ 程序中的数据以变量或常量的形式描述。声明变量时,不仅要说明其数据类型,还要用标识符代表其占用的内存空间。C++ 的标识符遵循以下几条规则。

- 标识符只能由字母、数字以及下划线组成。
- 必须以字母开头,如果第一个字符是下划线,则被视为系统自定义的标识符。
- 严格区分字母的大小写,例如, `string` 和 `String` 是两个不同的标识符。
- 标识符可以任意长。
- 标识符不能是 C++ 保留的关键字。

附：C++ 的关键字如下。

ASM CONTINUE FLOAT NEW SIGNED TRY AUTO DEFAULT FOR
OPERATOR SIZEOF TYPEDEF CASE DE GOTO FRIEND PRIVATE
STATIC UNION BREAK DELETE PROTECTED STRUCT UNSIGNED
CATCH DOUBLE IF PUBLIC SWITCH VIRTUAL CHAR ELSE INLINE
REGISTER TEMPLATE VOLATILE CONST CLASS ENUM RETURN
THIS EXTERN LONG SHORT THROW WHILE

(4) 代码模板的参考答案

【代码 1】: `int number, secret, encrypt;`

【代码 2】: `cin>>number;`

【代码 3】: `secret=number*7+5;`

【代码 4】: `encrypt=(secret-5)/7;`

实践环节

完成一个程序, 将从键盘接收的 3 个整数中选出最大数并输出。程序的运行结果如下:



```
#include <iostream.h>
```

```
int main()
```

```
{
```

```
    【代码 1】:      //声明字符型变量 c
```

```
    【代码 2】:      //声明 short 型变量 n
```

```
    【代码 3】:      //声明 int 型变量 a
```

```
    cout<<"请输入三个整数"<<endl;
```

```
    【代码 4】:      //从键盘接收三个整数, 分别赋值给变量 c,n,a
```

```
    【代码 5】:      //声明 int 型变量 max, 其初始值为 c
```

```
    if(n>max)        //判断 max 和 n 的大小
```

```
    【代码 6】:      //将 c 和 n 中的较大值赋值给 max
```

```
    【代码 7】:      //判断 max 和 a 的大小
```

```
        max=a;      //将 max 和 a 中较大值赋值给 max
```

```
    cout<<"c、n、a 中最大值为: "<<max<<endl;
```

```
    return 0;
```

```
}
```

2.2 浮点类型

核心知识

C++ 中按照精度的不同划分了三种不同的浮点类型, 分别是单精度(float)、双精度(double)和长双精度(long double)。