**小学生C语言编程**

**教 材**

目录

[第一单元 走进神奇的编程世界 - 2 -](#_Toc520046364)

[第1课 你“说话”的威力超出你的想象 ——认识人类与计算机沟通的语言 - 2 -](#_Toc520046365)

[第2课 送你一项指挥计算机“炒菜”的本领 ——小学生学习编程是为了技能？ - 3 -](#_Toc520046366)

[第二单元 1+1=2和计算器的故事 ——计算机帮你实现“四则运算” - 4 -](#_Toc520046367)

[第1课 了不起的“1+1=2” ——探索我们的大脑如何完成运算 - 4 -](#_Toc520046368)

[第2课 拆解计算器 ——探索计算器的计算奥妙 - 5 -](#_Toc520046369)

[第3课 创造一款可以做加法的计算器 ——从计算器计算到计算机编程计算 - 7 -](#_Toc520046370)

[第4课 准备好你的语言"翻译官" ——C语言编译工具介绍 - 9 -](#_Toc520046371)

[第5课 控制好计算机的“眼睛”和“嘴巴” ——编程术语之输入与输出 - 10 -](#_Toc520046372)

[第6课 探秘计算机的“记忆”能力 ——编程术语之变量 - 13 -](#_Toc520046373)

[第7课 准备好功能更多的“小箱子” ——编程术语之变量的类型 - 15 -](#_Toc520046374)

[第8课 综合设计 ——创造一款可以进行“四则运算”的计算器 - 18 -](#_Toc520046375)

**小学生C语言课程教学内容大纲**

# 第一单元 走进神奇的编程世界

## 第1课 你“说话”的威力超出你的想象 ——认识人类与计算机沟通的语言

同学们日常学习生活中经常要听老师讲课或者和其他同学聊天交流等等，是不是感觉很自然的事情，但有没有想过自己是怎么做到这一点的呢？和其他人交流真的像我们一直认为的那样容易么？如果同学们去和其他国家的同学们去沟通交流还会像我们现在这样容易么？可能有的同学想到了，我会英语，我可以和英国的同学们去沟通交流！是的，我们可能注意到了，现在“语言”出现了，因为语言的存在，所以掌握相同语言的人类就可以将内心的想法和思想无障碍的表达给对方，“语言”这项神奇的技能，虽然在我们的成长过程中发展的很自然，然而不知不觉中还是经过了从胎教就开始的启蒙和训练，以至于我们几乎快忘记了，一年级拼音识字时遇到的困难。

所以，当小明想让小李帮他拿一件东西给自己的时候，也同样需要共同的语言来传递这样的想法，这个过程不只是纯粹的聊天交流，而是一个指挥动作的执行过程，小明说“小李，请帮我拿一下角落里的书包”，这句话本身是语言表达即——汉语表达，而这句“话”其实是一条指令。什么是指令呢，就是指示完成某一动作的“话”，话的背后即是语言。所以，当我们需要让计算机帮助我们更好的完成工作时，我们需要找到一种能和它沟通交流，便于发布指令的共同语言。这就是我们说话背后的威力，远远超出我们自己习以为常的能力。

既然，我们和计算机沟通也需要尝试使用一种语言来进行，那么，用这种语言和计算机如何交流呢？大家首先想到的肯定是通过我们的嘴巴“说出来”。但目前的技术，还不能做到我们说出来一种共同的语言，计算机就可以理解并做出相应的反馈。既然嘴巴“说出来”不行，那我们还有什么方式进行交流呢？有的同学想到了，那就是写出来。不错，我们可以通过写出来这样的语言交给计算机去理解，但我们需要使用一种叫做“输入设备”的东西，把这样的语言存到计算机的大脑里，计算机才可以理解并做出相应的反应。那常见的输入设备，写东西到计算机的大脑里，使用什么呢？答案是：键盘，没错，键盘是我们以后经常要用到的和计算机交流的输入设备。

那么，这种和计算机沟通交流的语言是怎么发明出来的呢？谁来教我们这种语言呢？下节课，我们继续学习。

【拓展阅读】

动物的世界，可不可以像我们人类世界一样用共同的“语言”交流呢？比如小蚂蚁，我们经常可以看到，它们总是一起排队搬运东西，那么它们的语言是什么呢？它们有自己独特的“语言”，只是这种语言没有像我们人类这样是一种智能行为，它们的“语言”是通过一些身体本能的信息素散发来相互感知交流的。同学们，还能不能找出生活中用到的“语言”的例子呢？

## 第2课 送你一项指挥计算机“炒菜”的本领 ——小学生学习编程是为了技能？

什么是编程，编程有那些工具，这些工具支持什么编程语言，小学生学习编程应该如何选择？会带来怎样的思维变化。为什么要选用C语言?参考下面链接

https://mp.weixin.qq.com/s/oKVnlylOGjt0fuKzfemlTg

# 第二单元 1+1=2和计算器的故事 ——计算机帮你实现“四则运算”

## 第1课 了不起的“1+1=2” ——探索我们的大脑如何完成运算

同学们，还记得一年级数学课上，老师是怎么帮我们理解1+1=2这个计算规则么？或许是这样：老师先拿出一个苹果摆在桌上，当计算加法时，就将另一个苹果拿出来也放到桌上，然后让大家数出来（假设认识数字和数数字的基础已经具备了），并把结果告诉老师。当计算1-1的结果时，或许就是这样：老师先拿出一个苹果摆在桌上，当得知计算减法时，就将这个苹果拿走或者吃掉，然后让大家数出来是不是还有苹果，并把结果告诉老师。从此以后，数学老师用这样摆实物或者掰手指的方式，让我们逐渐掌握了更多的数学运算规则。那后来我们是如何计算更大数的运算呢？这里以两个数的运算为例，老师让小明计算3+2=？：

已经是二年级的小明，终于可以摆脱掰指头辅助计算的尴尬，有了心算或草稿计算的能力，小明计算过程是这样的：首先先在心里记住或者叫存储这两个要做计算的数字，再记住计算的是加法还是减法，然后进行“心算”或者列出草稿算式辅助计算，虽然叫心算，其实我们分别用到了自己大脑的记忆和运算功能（而不是心脏的什么功能），最后得出结果展示出来。那么展示结果我们基本有两种方式：一种是写在黑板上给同学或者老师看，另一种是自己将结果说出来给大家。好的，到这里同学们一定回忆起了这个计算过程，我们用插图表示一下：

记忆运算数

“2”

记忆运算数

“3

记忆并识别运算符号

“+”

大脑根据一年级

数苹果的规则

计算出结果：

“5”

（在黑板上）

写出结果“5”

（通过“嘴”讲话）

告诉大家结果“5”

## 第2课 拆解计算器 ——探索计算器的计算奥妙

同学们上节课探讨了1+1=2的计算过程，并且通过分析，知道了我们大脑在计算3+2=5的时候，进行了怎样的思考过程，那么大家有没有想过计算器（注意，不是计算机）是怎样实现加、减、乘、除运算的呢？这节课，我们以经常使用到的普通计算器（如下图）作为研究对象，来探究一下它能进行基本运算的奥秘！学习完本节课内容后，自行拆解计算器，找出对应单元。掌握键盘是用来做输入功能的，显示器是用来输出结果展示的。

假设我们现在要用计算器计算3+2=？这个简单的问题，应该分几步呢？如下图所示：

计算器操作过程 解密计算器计算过程 人脑计算过程

输出结果到显示屏：

“5”

（在黑板上）

写出结果“5”

（通过“嘴”讲话）

告诉大家结果“5”

人体大脑根据一年级数苹果的规则

计算出结果：

“5”

计算器“大脑”根据“加法”运算规则，将存储的数字3和2做加法运算

记忆运算数“2”

记忆运算符号“+”

眼睛观察算式

记忆运算数“3”

识别键盘按键

将数字“3”

存储到计算器

识别键盘“+”号

将运算规则设定为“加法”

识别键盘按键

将数字“2”

存储到计算器

键盘输入运算符号

“=”

键盘输入运算符号

“+”

键盘输入

数字“2”

键盘输入

数字“3”

通过上面的计算器计算过程分析，我们进一步和人工计算过程进行了对比，同学们是否能找到他们的相同和不同之处呢？

相同之处：

都需要“输入”，什么是“输入”呢？对于人工计算过程就是通过眼睛将准备计算的数字读入到大脑中，对于计算器计算则通过被敲击的键盘将数字输送到计算器的记忆单元（称为存储单元）；

都需要存储（记忆），二者都需要将被运算（计算）的数字进行存储，人类进行“心算”时，将数字存储到了大脑记忆模块，计算器则将数字存储到了一个设计好的存储器中（称为存储单元）；

都需要计算，二者都需要根据运算符号代表的运算法则进行运算，这个过程人工计算是通过人脑完成，计算器则是通过设计好的计算电路（暂称为运算单元）完成；

都可以“输出”结果，什么是“输出”呢？就是将计算结果展示出来。那么对于人类来说，更多是通过将结果写在黑板上或者通过发言说出来展示结果，计算器则更多是通过将结果放到显示屏上显示展示结果。

那么有没有同学思考过，当我们通过列算式的方式进行复杂的大数计算时是不是依然和上面的步骤一样呢，答案是一样的，请同学们自己思考这个过程！

不同之处：

结果展示的方式有区别，我们人类可以通过将结果写出来或者说出来，也可以通过肢体语言比划出来，来完成结果的展示，而计算器则大多数是通过显示屏显示出来。（当然，有些计算器也可以通过语音播报的方式说出来）。也就是说，计算器作为人类发明的工具进行结果展示时，目前的方式相对单一，同学们学了编程课程以后可以探索改进计算器的输出表达方式，比如让计算器通过舞蹈来表达计算结果。

总结，不论是人工计算，还是通过计算器辅助我们进行计算，都有下图所示的工作步骤：

键盘输入

（数字、符号）

输出结果“5”

（显示屏）

“5”

+

“2”

“3”

“3”

“2”

存储单元

（将输入的数字“2”存储在另一个地方）

存储单元

（将输入的数字“3”存储在一个地方）

按运算符号代表规则进行计算

## 第3课 创造一款可以做加法的计算器 ——从计算器计算到计算机编程计算

这一节课，我们将帮助大家一起写下人生中的第一个小程序（什么是程序，后面会讲到），它的功能是一个可以进行加法运算的计算器，有的同学会想，我已经买来了可以用的计算器，为什么还要再设计一个计算器呢？因为这一款计算器将是你自己生产制作的简单“计算器”，它会是你去探索宇宙奥妙的小梯子，等到你们去到更远的地方的时候，你会发现买来的计算器并不能帮助你更多。好了，那我们一起来看看你的这款计算器到底是什么模样呢？大家还记得我们拆解的计算器，从外面观察它是由显示器和键盘的东西组成，拆开看的时候，发现它里面还有绿色的小板子和很多黑黑的小方块，我们把绿色小板子和黑黑的小方块叫做电路，那我们看看我们的计算器从外面观察是什么模样呢？

（计算机图片）

大家看到了吧，原来我们准备设计制作的“计算器”从外面观察和叫做计算机或者“电脑”的机器长的一样呢，真是这样么？其实他们完全不一样，我们的计算机因为有了叫做软件的东西，变得像变形金刚一样，通过软件可以做很多事情，比如同学们特别喜欢的游戏软件，当你打游戏时，你感觉计算机就变成了一台“游戏机”，当你用视频软件播放动画片的时候，你感觉计算机就变成了一台“电视机”，但计算机只是计算机，只是因为它的脑袋里面装了很多软件，所以可以像变形金刚一样进行功能“变形”。那这些神奇的软件是怎么来的呢，它是通过一群叫做“程序员”的怪叔叔通过编写程序（编程）设计制造出来的。那编程到底又是什么呢？编程就是在计算机上用我们人类和计算机都可以理解的语言，给计算机写任务“小纸条”的过程，那我们为什么要写这些小纸条呢，因为计算机可以按照我们小纸条上写好的任务去做事情，这些在小纸条上写好的东西，我们就叫做程序，也就是那些我们安排给计算机的任务。所以我们平时经常说的编程、写代码、写程序可以简单理解为给计算机写一个任务小纸条让它去执行。

现在，因为我们是想让计算机帮我们进行加法计算这个任务，于是，我们在它的脑袋里面给它安装了一个我们自己写的程序（一个小纸条），而这个程序可以进行加法计算这个指定任务，所以我们把它叫做加法计算器。由于我们写在计算机脑袋里的这个程序没有自己单独的键盘、显示器和绿色的电路小板子，按键和显示加法计算结果都是通过电脑的键盘和显示器帮我们实现的，所以我们可以把它叫做软件“计算器”（生活中我们经常使用到的计算器叫做硬件计算器。）那么这个软件的心脏里装的是什么东西呢，也就是我们写给计算机的小纸条写的是什么东西呢？

#include<stdio.h>

int main()

{

int a, b, s;

scanf("%d,%d",&a,&b);

s=a+b;

printf("%d",s);

return(0);

}

哇哇，大家第一眼看过去的时候一定觉得这是什么乱七八糟的东东，好的，这就是可以和计算机进行沟通的C语言啦！每一句C语言也叫做一句代码，马上你就明白这些代码是什么东西啦，接下来我们用我们人类的语言来翻译一下：

|  |  |
| --- | --- |
| C语言表达（代码） | 人类语言表达（人话） |
| #include<stdio.h> | 告诉计算机：请你认真听，准备好，我准备让你做些事情。 |
| int main()  {  ......  } | 告诉计算机：我准备将我的主要讲话内容和让你做的任务放在这 个大括号“{ }”里面。 |
| **int** a, b, s; | 告诉计算机： a、b、s分别代表可以存放一个数字的存储(记忆)单元名称（就像我们前面讲到的记忆数字的地方），**int** 代表这些被起好名字的存储单元里只能存入整数。 |
| scanf("%d,%d",&a,&b); | 从键盘输入两个整数，分别存放到名字为a和b的存储单元中。 |
| s=a+b; | 将刚才存放到a和b中的两个整数相加，相加结果存放在s中。 |
| printf("%d",s); | 将名字为s的存储单元中存放的整数显示到电脑屏幕上。 |
| return(0); | 告诉计算机：全部讲话结束。 |

那么，现在我们交给计算机执行的任务“小纸条”就准备好了，但是我们怎么把这个“小纸条” 传递给计算机执行呢，这里我们就要用到一个超级无敌工具——翻译“软件”，它的正式名称其实叫“编译软件”，它叫什么不重要，同学们只要知道，编译软件有很多种，但我们这里选用非常适合传递或者翻译C语言的编译软件给你的计算机，它的名称叫DEV C++，下一节课我们将详细介绍DEV C++这个翻译工具。大家这里会有疑问，C语言不是已经可以和计算机沟通了，为什么还要再对C语言进行翻译给计算机呢？因为C语言是我们人类和计算机共同可以理解的语言，是为了方便我们人类书写小纸条任务的高级语言，而计算机能直接理解的语言其实叫做“机器语言”，所以我们需要一个翻译工具，也就是编译软件将我们写好的C语言程序翻译成机器语言，然后计算机就可以执行我们书写的任务了。具体过程示意图如下：

C语言程序

（任务小纸条）

机器语言

编程 编译软件翻译

## 第4课 准备好你的语言"翻译官" ——C语言编译工具介绍

## 第5课 控制好计算机的“眼睛”和“嘴巴” ——编程术语之输入与输出

同学们想一想目前计算机的“眼睛”和“嘴巴”是什么呢？你一定会首先想到：摄像头和音响！这节课则要告诉同学们，计算机的“眼睛”和“嘴巴”更像是它的键盘和显示器！你一定很惊讶，怎么会是它们呢？

首先我们来想一想，我们自己的眼睛和嘴巴主要作用是什么呢？我们眼睛和嘴巴的作用有很多，比如让自己显得更美丽，比如嘴巴可以吃东西等等。但是，其中眼睛很重要的作用是能够帮助我们看到这个世界的模样，感受到世界的美好，嘴巴则可以将我们的心灵感受表达出来，让更多的人知道我们是怎么思考的。再具体到我们做数学作业的场景中，眼睛可以帮助我们读题，将需要计算的数字通过眼睛“输入”到大脑记忆单元，再通过我们大脑的运算功能得到计算结果，最后我们可以选择将计算结果通过嘴巴“说”出来，也可以选择通过手写展示到作业本上。就这个场景中，想一想上节课，计算机是如何帮助我们做加法的呢，首先计算机也要读题，然后再将要计算的数字输入到计算机的“大脑记忆单元”，由于我们目前的技术手段原因，通过计算机的摄像头直接来读题并理解的方式还不够普及。所以，这个过程，是在我们人的帮助下理解题目，然后通过键盘将计算题目要用到的数字“输入”到计算机的“大脑记忆单元”（术语叫存储单元），再通过我们设定好的任务小纸条（即是编写好的程序）进行运算得到结果，最后通过计算机的显示器将结果展示给我们。所以到这里，我们发现计算机的键盘也很像是我们的“眼睛”，可以将数字“输入”到它的大脑记忆单元，那么是不是键盘只能输入数字呢，当然不是，键盘可以根据你的需求输入很多东西。同理，计算机的显示器就更像是计算机的“嘴巴”和结果展示本。现在我们很多的科学家正在研究如何让计算机的视觉系统和语言系统更像我们人类。同学们可以课后进行相关的探究。

接下来，我们首先来认识并学会控制好计算机的“嘴巴”，也就是计算机编程的输出功能，如果现在你想让计算机“说”（输出）一句话出来，你该怎么办呢？

（1）思考我准备让计算机对我“说”什么或者“输出”什么。

（2）给计算机准备好这项“输出”任务。

（3）将这项任务用C语言写在“任务小纸条”上。（这就是编程过程，小纸条内容就是程序）。

（4）将“任务小纸条”交给你的“翻译官”翻译成机器语言。

（5）计算机理解了你交给的“输出”任务，将这句话展示到计算机显示器上。

（6）“说”一句话的任务完成。

按照以上步骤来编写程序，也就是准备你的任务小纸条。

第一步，假设你想好了准备让计算机对你说：欢迎进入编程世界！

第二步，用C语言来描述这项任务：

C语言用“printf”这个单词来表示输出一些内容到计算机显示器上，而这个单词本意也就是“打印”“输出”的意思。

【题外话】那这项任务为何要用英文描述而不是中文“输出”这个词来直接描述呢？因为C语言这套和计算机进行交流的语言规则，是一个外国人发明创造的。如果同学们学了本课程以后，想设计创造一个和计算机直接进行中文交流的语法规则，那将会是一个非常不错的想法，你将会因为第一个完成这样伟大的创造而被大家铭记。目前来看，C语言就是由这类英文单词和一些字母、数字、符号等构成的，从本节课开始，同学们也就要陆续记住一些类似这样的英文单词来进行C语言的描述，最后翻译给计算机去执行我们安排的任务。

现在我们知道了，printf这个单词就是告诉计算机要输出一句话到显示器上，那同学们猜一下，这个输出任务的完整C语言描述语句会是怎么样的呢？

这样可以么：printf：欢迎进入编程世界！;

这样可以么：printf "欢迎进入编程世界！";

这样可以么：printf（欢迎进入编程世界！）;

这样可以么：printf【欢迎进入编程世界！】;

同学们有没有发现，如果C语言只规定了一个单词来表示输出的话，是不够的，这样会导致每个同学交给计算机“翻译官”的语句写法都不一样，计算机就无法理解我们的任务。就像我们每个中国人虽然都在用汉字来表达和讲话，但如果词句的先后顺序规则没有统一，大家最后就没办法理解对方的语言。

所以，一定要有一套统一的规则，规定完整的计算机“输出”任务语句，而C语言其实已经规定好了，这就是C语言的语法规则，用来统一标准，防止引起误解。我们现在来学习一下printf语句的完整语法规则：

printf( "欢迎进入编程世界！" );

同学们看到了吧，只有你将输出任务用这样一句C语言来描述，计算机才能通过翻译官正确的交给计算机执行。这个语句由以下几个部分组成：

（1）关键词：printf

（2）规则符号：括号+双引号 (" ")

（3）语句结束符号：分号 ;

（4）文字内容：欢迎进入编程世界！（这句话必须放在（2）的双引号中）

以上的规则术语叫做C语言语法规则，也就是大家共同约定好的统一格式，只有按这个格式书写C语言，“翻译官”才能正确翻译。

如果你想让计算机显示的不是单纯的一句话，而是键盘上的一些符号，符号与符号之间要进行换行或者空格隔开怎么办呢？符号可以直接像文字一样放到双引号当中就可以了，换行则要通过"\n"来表示。比如显示如下图案：

\*

\* \* \*

该语句要这要写：

printf( " \*\n\* \* \*" );

也可以写成这样两句：

printf( " \*\n" );

printf( "\* \* \*" );

同学们，发现换行要用一个斜杠 \ 和字母 n 的组合“\n”来代替，空格则可以直接在双引号当中用敲击键盘空格来代替。所以双引号当中的功能有两个，一个是显示输出的文字，另一个是进行输出格式的控制。输出格式控制就是你想换行或者空格等这样的控制操作。到这里，同学们就初步掌握了输出内容和格式的控制，接下来学习一下输出的另一种语法格式。

我们现在来看想要将键盘输入的数字放到计算机的“大脑记忆单元”中该怎么办？

输入的关键词是“scanf”，功能是从计算机键盘接收输入的数字或字符存放到指定的记忆存储单元，具体语法格式如下：

scanf ("%d,%d", &a, &b);

这句输入功能C语言，分解为以下几个组成部分：

（1）关键词：scanf

（2）规则符号1：括号+双引号+ 逗号 (" ", )

（3）规则符号2：记忆单元地址符号 &

（4）规则符号3：输入内容类型控制符号 %d，准备输入几个**整数**就用几个%d来表示

（5）记忆单元名称：a,b等字母命名的名称，一个字母或者一个自命名单词代表一个记忆单元

（6）语句结束符号：分号 ;

以上格式，用第一个逗号“,”将括号内的控制分成两部分，在双引号内进行输入类型和数量的格式控制；双引号后面的部分用来告诉计算机，键盘中输入的数字或字符放到那几个指定的记忆单元中。

例如，输入3个整数存放到名称为a,b,c的三个记忆单元中，C语言功能语句如下：

scanf ("%d,%d,%d ", &a, &b, &c);

那么这句C语言代码写成这样行不行呢：

scanf ("%d%d%d ", &a, &b, &c);

同学们可以发现，这一句的区别是%d之间缺少了逗号（“,”），这样虽然是可以的，但最好是输入数字的格式和格式控制部分写出的格式一致，比如%d之间以逗号隔开，在我们输入数字的时候就以逗号隔开，当%d之间没有任何符号隔开时，我们该怎么判断一个数是否输入完毕了呢？答案是输入一个数字完成后，以空格，回车等都可以使计算机认为这个数字输入完毕（这个操作只对数字类型成立）。

那么这句C语言代码再改写成这样行不行呢，即去掉双引号后面的所有逗号：

scanf ("%d,%d,%d " &a &b &c);

不行！只有双引号内的逗号可以根据你的喜好省略，其他的逗号是不能省略的，就像我们用汉语说话一样，标点符号不能省略，否则就会引起歧义。神奇吧？同学们知道了格式控制的重要性了么？

我们再回头看一看输出功能语句的另一种形式，将某记忆单元中的内容输出到显示器怎么办呢？例如，将上例中存储到a,b,c三个记忆单元中的三个数字输出：

printf ("%d,%d,%d " ,a , b, c);

同学们，是不是发现输出功能printf和输入功能scanf语法规则很相近，只是printf语句在记忆单元名称（a,b,s）前不需要使用地址符号&。同样的，printf需要输出几个记忆单元，双引号中的%d就要相应的有几个。

## 第6课 探秘计算机的“记忆”能力 ——编程术语之变量

变量？什么意思？其实同学们已经和这个东西认识很久了，只是我们以前一只在喊它的小名“记忆单元”“存储单元”，也就是字母“a”“b”“s”，怎么样，有印象吧！本节课为了让同学们更全面地了解计算机“记忆”能力，全面对计算机的“记忆”能力代表——变量进行详细介绍。

同学们回忆一下之前的课程，数学加法作业的场景，我们人类记数字是通过大脑的记忆单元完成的，计算器记住数字是通过黑色小方块的电路单元完成记忆的，那么计算机是怎么样完成记忆的呢？虽然计算机是比计算器复杂的多的机器，但是他们都是我们人类设计制造出来帮助我们的机器工具，所以在记忆能力这方面，他们有些相似的地方，就是都通过一些更大的黑色小方块电路来存放东西的，这些电路（我们称为硬件内存）就和我们人类大脑的记忆单元有一样的作用。那我们是不是需要直接把计算机拆开，然后找到这些黑色小方块的硬件，对他进行操作呢?显然不是，C语言的发明者就创造了一个叫变量的东西，让我们可以直接使用变量来对计算机的记忆进行控制，我们只需要对变量进行控制，“翻译官”就会自动帮我们翻译成计算机的任务，由计算机自己去对这些黑色小方块进行操作。

为了便于理解，同学们可以将变量看作是我们生活中用到的小纸箱，这些纸箱有很多，有的小纸箱用来放衣服，有的小纸箱用来放玩具，而且为了我们使用方便，我们会给每个纸箱起一个名字，比如玩具熊纸箱、积木纸箱等等。具体到C语言当中，变量用来存放数字、字符等东西，不能存放玩具或者衣服。那怎么样用C语言来申请一个纸箱来存放我们的数字呢？

int m;

分析这句代码，由两部分组成，开头是int，这表示你申请的箱子的类型，int代表整数，也就是说你的这个类型后面的箱子只能用来存放整数；int后面空格再书写字母m，m代表箱子的名称。那你觉得一个箱子不够用，需要多几个箱子怎么办？比如要用4个箱子：

int m, n, p, q;

或者

int m;

int n, p, q;

上面的这个过程叫做变量声明，相当于告诉计算机准备好一些特定类型和数量的箱子，我后面会用到。

声明变量的语句由两部分组成：变量类型 变量名称;两者中间要有空格。

变量类型如int，代表准备好存放整数的变量；变量名称如m,n,p,q表示后面会分别用到。

给你的小纸箱也就是变量起名字是有规则的，不能随意命名，就像我们的名字一样有一些基本规则。命名规则就是每个变量名可以由字母、数字和下划线组成，开头必须是字母或者下划线，比如“int a\_1;”比如“int sum;”等等，所以变量名不一定是一个字母，只要符合命名规则，又方便自己记忆就可以。

【练一练】现在设计一款小程序（软件），实现计算长方形周长的功能。要求根据数学作业长和宽的要求，从键盘输入这两个数字（整数），最后将周长计算结果显示到计算机显示器上。

五步分析法：

（1）程序要求输入的数字有几个，是什么类型：

输入长和宽两个数字，类型为整数

（2）输入的数字放到什么地方：

存放到2个变量中

（3）程序要求输出的数字有几个，是什么类型：

输出一个周长，类型为整数

（4）基于以上过程，总共需要使用到几个变量，如何命名方便记忆：

3个变量，分别是长、宽和周长，用汉语拼音首字母起名，c、k、zc

（5）输入和输出之间有什么公式关系：（输出）周长=（输入）长\*宽\*2

好的，分析完成，写出完整代码

六步代码法，写好任务小纸条，即编程或者写代码：

第一步：

告诉计算机准备一下器材，我要写程序了，相应C语言：

#include<stdio.h>

第二步：

告诉计算机，你的任务小纸条主要内容放在什么地方，就是下面的大括号里：

int main()

{

......

}

第三步：

首先声明变量，准备好你将要用到的变量或者小纸箱，明确名称、个数和类型：

int c, k, zc;

注意这里是刚才五步分析法分析的结果，c代表长，k代表宽，zc代表周长，都为整数

第四步：

确定输入变量是那几个，将键盘输入的数字存放在这些变量中，即输入

scanf ("%d,%d", &c, &k);

记住在使用输入功能时，地址符号&不能丢，现在即表示第一个键盘输入的数字放到了c变量中，第二个数字放到了k变量中。

第五步：

找出输出和输入的关系公式，这里，需要输出的周长=长\*宽\*2，\*号代表数学乘法

C语言表示以上公式：

zc=c\*k\*2;

注意分号“;”不能丢，每一句代码结束都要加上分号“;”

第六步：

输出结果：printf("%d",zc);注意这里输出不需要使用地址符&

## 第7课 准备好功能更多的“小箱子” ——编程术语之变量的类型

上节课，我们学习了整数类型的变量，也就是只能用来存放整数的小箱子，这节课会准备更多类型的箱子给大家“玩耍”。

思考这样的问题：

现在设计一款小程序（软件），实现计算长方形周长的功能。要求根据数学作业长和宽的要求，从键盘输入这两个数字（小数类型），最后将周长计算结果显示到计算机显示器上。

功能和上节课一样，但是要求输入的数字类型为小数，那么这样该怎么办呢。我们用六步代码法来完成这项任务：

跳过前两步，直接从第三步开始：声明变量

小数用float这个关键词来表示，同样的声明对比看：

上节课整数声明：int c, k, zc;

本节课小数声明：float c, k, zc;

第四步：输入

上节课用%d来进行格式控制，表示输入的是整数类型

scanf ("%d,%d", &c, &k);

本节课用%f来进行格式控制，表示输入的是小数类型,变量命名和个数没有变

scanf ("%f,%f", &c, &k);

第五步：找出输入输出关系公式

需要输出的周长=长\*宽\*2，\*号代表数学乘法,C语言表示以上公式：

zc=c\*k\*2;这个没有发生变化

第六步：输出结果

上节课用%d来控制输出语句，本节课用%f来进行格式控制，表示输出的是小数类型,变量命名和个数没有变

上节课printf("%d",zc);注意这里输出不需要使用地址符&

本节课printf("%f",zc);

完整代码如下：

#include<stdio.h>

int main()

{

float c, k, zc;

scanf ("%d,%d", &c, &k);

zc=c\*k\*2;

printf("%f",zc);

}

同学们一定发现了变量类型更换以后，输入和输出功能的类型控制符号也由%d变成了%f，这是配套一致的，因为存放类型决定了输入和输出的类型格式。

除了这些类型，还有什么类型呢。有时候我们会输入字母、单词或者是某种符号，这就是字符或字符串类型，字符串我们暂时不考虑。

字符变量声明关键词为：char，输入输出功能格式控制符号为%c

字符变量命名规则和其他变量一样。

【练一练】

英语老师有时候会给我们布置作业，要求把大写字母对应的小写字母写出来，这个作业计算机可以辅助你完成，深不神奇，期不期待？我们就设计这样一个程序，功能是：输入一个大写字母，将它对应的小写字母输出。

在正式开始程序设计前，我们首先来补充一些关于计算机中字符的一些知识：

在计算机中英文的26个字母是存储中一个叫ASCII字符集的东西里，这是一个什么东西呢，这是一个每个字母都可以用一个数字来表示的集合，当你给出这个数字时，你的计算机翻译官会根据你的需求自动翻译为字符，比如大写字母'A'在这个ASCII字符集中用65这个数字来表示，那怎么输出这个字母A的整数表示呢：

char m;

m='A'

printf("%d",m);

结果为：65

你一定感到很奇怪，不是字符么，怎么就变成了数字了呢，这就是格式控制的原因，因为你指定输出一个%d格式的东西出来，也就是代表A的整数，如果你换成下面这样：

char m;

m='A'

printf("%c" , m);

此时会将字母A输出，因为你指定输出一个%c格式的东西出来，这也进一步说明了格式控制的重要性。

所以，现在同学们知道了字母背后其实是数字，只是因为“翻译官”的存在，才会将这些数字按指定格式输出。ASCII字符集中，任意一个英文小写字母都比大写字母大32，所以有了这个基础，我们就可以设计解决刚才的问题了。

五步分析法：

（1）程序要求输入的内容有几个，是什么类型：

输入一个大写英文字母，字符型（char）

（2）输入的数字放到什么地方：

存放到1个字符变量中

（3）程序要求输出的内容有几个，是什么类型：

输出一个小写英文字母，字符型（char）

（4）基于以上过程，总共需要使用到几个变量，如何命名方便记忆：

2个变量，输入用c1来命名，输出用c2来命名

（5）输入和输出之间有什么公式关系：

c2=c1+32;

六步代码法：

第一步：

告诉计算机准备一下器材，我要写程序了，相应C语言：

#include<stdio.h>

第二步：

告诉计算机，你的任务小纸条主要内容放在什么地方，就是下面的大括号里：

int main()

{

}

第三步：

首先声明变量，准备好你将要用到的变量或者小纸箱，明确名称、个数和类型：

char c1, c2;

注意这里是刚才五步分析法分析的结果，c1代表输入的大写字母，c2代表相应的小写字母

第四步：

确定输入变量是那几个，将键盘输入的数字存放在这些变量中，即输入

scanf ("%c ", &c1);

记住在使用输入功能时，地址符号&不能丢，现在即表示键盘输入的字母放到了c1变量中。

第五步：

找出输出和输入的关系公式，这里，需要输出的变量c2比c1多32.

C语言表示以上公式：

c2=c1+32;

注意分号“;”不能丢，每一句代码结束都要加上分号“;”

第六步：

输出结果：printf("%c",c2);注意这里输出不需要使用地址符&

完整代码如下：

#include<stdio.h>

int main()

{

float c1, c2;

scanf ("%c ", &c1);

c2= c1+32;

printf("%c", c2);

printf("%d\n", c2);

}

试试你的程序：

输入：A

输出：a

97

想一想为什么是这样的结果？

## 第8课 综合设计 ——创造一款可以进行“四则运算”的计算器