

# 零基础学 编程

树莓派和Python

金学林 著



中国工信出版集团



电子工业出版社  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY  
http://www.phei.com.cn



# 零基础学 编程

树莓派和Python

金学林 著

电子工业出版社  
Publishing House of Electronics Industry  
北京·BEIJING

## 内 容 简 介

未来是计算机和机器人的世界。软、硬件编程将成为未来世界重要的一项技能。

通往山巅的技术之路有无数条，每条路的方式和难度都不一样。本书给零基础的读者指明了一条清晰适合的路径，采用通俗易懂的讲解方式，将软、硬件结合，并利用新奇、有趣的案例来激发读者的兴趣，让读者跨过进入编程世界的第一道门槛。

本书适合零基础而又想学习编程的任何人，可作为小学高年级，以及初、高中学生课外编程或机器人兴趣班的学习辅导书，也可作为树莓派电脑编程学习的入门指导书籍。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。  
版权所有，侵权必究。

### 图书在版编目 (CIP) 数据

零基础学编程：树莓派和 Python / 金学林著. —北京：电子工业出版社，2018.7  
ISBN 978-7-121-34344-5

I. ①零… II. ①金… III. ①软件工具—程序设计 IV. ①TP311.561

中国版本图书馆 CIP 数据核字(2018)第 115585 号

责任编辑：石 倩

印 刷：三河市鑫金马印装有限公司

装 订：三河市鑫金马印装有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编：100036

开 本：787×980 1/16 印张：13.75 字数：270 千字

版 次：2018 年 7 月第 1 版

印 次：2018 年 7 月第 1 次印刷

定 价：49.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010) 88254888, 88258888。

质量投诉请发邮件至 [zlts@phei.com.cn](mailto:zlts@phei.com.cn)，盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

本书咨询联系方式：010-51260888-819, [faq@phei.com.cn](mailto:faq@phei.com.cn)。



# 前言

万事有果必有因。

这本书能够和大家见面，也是因为数个偶然。

最早的起因，是笔者想写一个培训教程，帮助小学生来学习编程。在给他们做培训的过程中发现，从零基础开始学编程缺少一个合理而清晰的学习路径，因此笔者就在思索，如何能够更有效地开始学习编程。

计算机技术发展到现在，可以说学习资料和教材浩如烟海。一个零基础的学生，该如何选择一条适合自己的学习路径，从而轻松跨过进入编程世界的第一道门槛。这是一件非常困难的事情。

选择太多，对于初学者来说并不是一件好事。一条清晰的学习路径，相对平缓的学习难度曲线，每一个步骤都有详细说明，丰富的程序例子，尽量多的截图，这些都是笔者所设想的教程特色。

真正促使笔者将这些想法落实到行动上的触发点，是笔者有一个上幼儿园的儿子，姑且不管这些教程对别人有没有作用，至少可以作为培养儿子的资料，就当是教育投资也好啊。

笔者写完大概 10 小节的内容之后，就尝试着将这些教程发布到多个自媒体平台，没想到收获了很多粉丝的赞同和认可，有了读者们的鼓励和支持，笔者就更有动力继续编写下去了。后续又发布了更多的教程，没想到受到了电子工业出版社石倩老师的认可并向笔者约稿，因此才有了此书的面世。

不忘初心，方得始终。笔者的初心很简单，希望能够给零基础的初学者一些建议和指导，



能够跨过学习编程的第一道门槛，站上编程世界的第一级台阶。如果有读者觉得达到了这个目的，笔者就觉得不虚此行了。

### 本书主要内容：

- 从编程环境入手，通过安装树莓派电脑掌握基础知识。
- 通过安装 Python 以及编写第一个 Hello World 程序来学会如何编写代码和运行程序。
- 学习 Python 的基础语法和编程思路。
- 引入一些可以用于树莓派 GPIO 接口控制的传感器零件，学会组装连接、控制运行。
- 将多个零件组合，代码融会贯通，完成一个基本的功能。

不矫情，但还是要说很多感谢的话。

感谢爱人的支持，将家里安排得很好，让我没有后顾之忧。感谢老朋友老赵的支持，他给了很多建议。感谢石倩编辑的认可和鼓励，多次耐心的沟通和讨论，负责认真的审稿。感谢出版社的各位编辑老师，你们的辛勤工作值得更多的感谢！

因本书中有需要读者动手组装的课程，只看文字并不容易理解，后续笔者会陆续将其制作成视频上传至个人网站 ([www.code66.cn](http://www.code66.cn))。如果读者有任何意见或建议，欢迎加入交流 QQ 群 (603559164)。

作者

---

轻松注册成为博文视点社区用户 ([www.broadview.com.cn](http://www.broadview.com.cn))，扫码直达本书页面。

- 下载资源：本书如提供示例代码及资源文件，均可在 [下载资源](#) 处下载。
- 提交勘误：您对书中内容的修改意见可在 [提交勘误](#) 处提交，若被采纳，将获赠博文视点社区积分（在您购买电子书时，积分可用来抵扣相应金额）。
- 交流互动：在页面下方 [读者评论](#) 处留下您的疑问或观点，与我们和其他读者一同学习交流。

页面入口：<http://www.broadview.com.cn/34344>





# 目 录

第 1 章 编程基础知识和环境准备 .....	1
1.1 零基础的小白能学会编程吗 .....	2
1.1.1 为什么要学编程 .....	2
1.1.2 兴趣是最好的老师 .....	3
1.1.3 为什么零基础的自学编程者，大多半途而废 .....	3
1.1.4 有趣好玩很重要 .....	4
1.1.5 家长带着孩子一起学是最好不过的 .....	5
1.1.6 当你决定出发的时候，最困难的一部分已经完成了 .....	5
1.1.7 计划的重要性 .....	6
1.2 700 元的电脑真的可以学会编程 .....	6
1.2.1 计算机（电脑）的历史 .....	6
1.2.2 硬件和软件 .....	7
1.2.3 操作系统 .....	7
1.2.4 硬件选择 .....	8
1.2.5 准备材料 .....	11
1.2.6 如果已经有一台电脑了，怎么办 .....	12
1.3 十分钟组装一台可编程电脑 .....	12
1.3.1 给树莓派 3 主板安装散热片 .....	12
1.3.2 将树莓派 3 主板安装到亚克力外壳中 .....	15
1.3.3 接好移动电源 .....	18
1.3.4 接好显示器 .....	19
1.3.5 接好键盘和鼠标 .....	20
1.3.6 进入系统 .....	21



1.3.7	关机，分解装箱.....	22
1.4	如何编写人生的第一行代码：Hello World.....	23
1.4.1	如何开机.....	23
1.4.2	认识 LX 终端.....	23
1.4.3	编写第一个 Python 程序.....	28
1.4.4	设置网络.....	30
1.4.5	访问互联网.....	31
1.4.6	使用 Text Editor 文本编辑器输入英文.....	32
1.4.7	学会输入中文.....	34
1.4.8	关机.....	36
1.4.9	如果已经有一台电脑了，怎么办.....	37
<b>第 2 章 Python 编程语言基础.....</b>		<b>40</b>
2.1	加法计算器.....	41
2.1.1	直接在 Python 环境输入代码并运行.....	41
2.1.2	变量、赋值操作符、输入函数、参数、字符串、输出函数.....	42
2.1.3	了解加法计算器代码.....	44
2.1.4	字符串和整数是不同的数据类型.....	45
2.1.5	将 Python 代码放到文件中.....	46
2.1.6	执行 Python 程序文件.....	47
2.1.7	参考加法计算器的代码创建类似程序——减法计算器.....	48
2.2	四则运算器.....	50
2.2.1	0 和 1 的世界.....	51
2.2.2	布尔类型——True 和 False.....	52
2.2.3	逻辑运算符——and、or 和 not.....	52
2.2.4	if 判断语句.....	53
2.2.5	用 if 语句判断输入的符号.....	55
2.2.6	测试程序.....	57
2.3	功能更丰富的四则运算器程序.....	59
2.3.1	增加注释行.....	59
2.3.2	让程序来判断输入的结果是否正确.....	61
2.3.3	让程序来出计算题目吧.....	62
2.4	计算日期所属星座.....	64



2.4.1	输入月份和日期.....	64
2.4.2	检查月份和日期是否正确.....	65
2.4.3	如何判断所输入的日期对应哪个星座.....	66
2.4.4	如何测试到每一种情况.....	69
2.5	Python 循环语句.....	69
2.5.1	什么是循环.....	69
2.5.2	while 循环的语法.....	70
2.5.3	从 1 加到 100 求和.....	71
2.5.4	从 1 开始, 连续 100 个奇数相加的结果是多少.....	72
2.5.5	从 2 开始, 连续 50 个偶数相加的结果是多少.....	73
2.5.6	从 1 月 1 日到 12 月 31 日的每一天分别是什么星座.....	73
2.6	循环的更多用法——斐波拉契数列.....	77
2.7	练习使用循环和判断语句.....	79
2.7.1	已知 2017/1/1 是星期天, 输出 2017 年每一天是星期几.....	79
2.7.2	输出 2016 年的每一天是星期几.....	82
2.7.3	输入一个年份, 判断是闰年还是平年.....	83
<b>第 3 章 Python 编程语言进阶.....</b>		<b>87</b>
3.1	列表类型.....	88
3.1.1	认识列表类型.....	88
3.1.2	访问列表中的值.....	88
3.1.3	更新列表.....	89
3.1.4	追加列表元素.....	89
3.1.5	删除列表元素.....	89
3.1.6	如何遍历列表.....	89
3.1.7	使用更简单的方法实现“输入数字 1~7, 判断是星期几”.....	91
3.1.8	改造星座判断程序.....	91
3.2	数据类型转换.....	95
3.2.1	统计包含“2”的数字总个数.....	95
3.2.2	标准数据类型.....	96
3.2.3	数据类型转换.....	96
3.2.4	函数 range.....	97
3.2.5	统计代码.....	98



3.2.6	二维列表.....	98
3.3	字典数据类型.....	101
3.3.1	认识字典数据类型.....	101
3.3.2	访问字典里的值.....	102
3.3.3	修改字典里的值.....	102
3.3.4	删除字典元素.....	102
3.3.5	判断是否存在键.....	103
3.3.6	如何遍历字典.....	103
3.3.7	改造“最多邮编省份名称统计”程序.....	103
3.3.8	输入一行字符串，打印出其中每个字符出现的次数.....	104
3.4	Python 函数.....	115
3.4.1	输入参数求三角形、圆形或长方形的面积.....	115
3.4.2	认识函数.....	116
3.4.3	函数的定义.....	117
3.4.4	函数的使用.....	118
3.4.5	按值传递参数和按引用传递参数.....	119
3.4.6	参数的几种形式.....	120
3.4.7	常用的系统内建函数.....	121
3.4.8	递归函数.....	121
3.4.9	改造“四则计算器程序”.....	124
3.4.10	改造面积计算程序.....	125
3.4.11	关于函数和模块设计定义的一些经验.....	126
3.5	模块和进程.....	130
3.5.1	认识模块.....	130
3.5.2	在另一个文件里导入模块.....	131
3.5.3	日期和时间模块.....	132
3.6	字符串操作和读写文件.....	133
3.6.1	认识字符串.....	133
3.6.2	访问字符串中的值.....	133
3.6.3	转义字符.....	133
3.6.4	字符串运算符.....	134
3.6.5	字符串格式化.....	135
3.6.6	常用的字符串内建函数.....	135



3.6.7	文件读写.....	136
3.6.8	统计文章中出现次数最多的 10 个字.....	136
<b>第 4 章</b>	<b>使用树莓派电脑控制各种硬件.....</b>	<b>140</b>
4.1	让 LED 灯亮起来.....	141
4.1.1	购买硬件.....	141
4.1.2	GPIO 介绍.....	141
4.1.3	LED 灯电路原理.....	144
4.1.4	硬件连接.....	144
4.1.5	编写程序.....	146
4.1.6	执行程序.....	146
4.1.7	程序中每行代码的说明.....	147
4.2	使用笔记本电脑远程控制树莓派电脑.....	151
4.2.1	需要网络支持.....	151
4.2.2	如何查看网络 IP 地址.....	151
4.2.3	如何远程登录.....	156
4.2.4	如何上传文件.....	159
4.2.5	如何执行树莓派电脑上的程序.....	162
4.2.6	如何通过图形界面访问树莓派电脑.....	162
4.2.7	摆脱线的束缚.....	166
4.3	发出蜂鸣声音.....	167
4.3.1	蜂鸣器.....	167
4.3.2	持续鸣叫.....	168
4.3.3	有节奏地鸣叫.....	169
4.4	控制温湿度传感器.....	170
4.4.1	温湿度传感器.....	170
4.4.2	硬件连接.....	172
4.4.3	编写程序.....	174
4.5	制作温度报警器.....	176
4.5.1	硬件连接.....	176
4.5.2	编写程序.....	177
4.6	控制单位数码管显示数字.....	180
4.6.1	电路原理.....	181



4.6.2	一个灯 A 管接线.....	182
4.6.3	程序解释说明.....	184
4.6.4	将全部灯管接线.....	184
4.6.5	显示数字 1.....	184
4.6.6	显示所有数字.....	186
4.7	控制双位数码管显示时间秒数.....	189
4.7.1	电路原理.....	189
4.7.2	刷新机制.....	190
4.7.3	全部灯管接线.....	191
4.7.4	显示数字 01.....	191
4.7.5	显示当前时间秒数.....	194
4.8	将测量温度显示到数码管并同时示警.....	199
4.8.1	电路原理.....	199
4.8.2	硬件连接.....	200
4.8.3	编写程序.....	201



# 第 1 章

## 第 1 章

### 编程基础知识和环境准备



## 1.1 零基础的小白能学会编程吗

有朋友问，编程难吗？一点都不懂的人能学会吗？

也有朋友问，我们家小孩很喜欢打电脑游戏，能让他学习编程吗？要多大才可以学啊？

还有朋友问，很想学习编程，但就是坚持不了，不知道怎样才能学会编程，有没有快速又简单的办法？

笔者的回答是：能，每个人都能学会编程，越早学越好。方法也很简单，即“兴趣+方法+坚持”。

### 1.1.1 为什么要学编程

Facebook 创始人扎克伯格说：“编程已成为一项基本技能，每个人都该会。”

仅在美国，每年就有 50 万个计算机相关的工作岗位，但是每年只有 5 万名计算机科学的学生毕业。

编程显然已成为一项基本技能，是每个人都应该做的事情，就像阅读一样。它是每个学校都应该传授的技能。

苹果公司创始人史蒂夫·乔布斯说：“人人都应该学会编程，因为它会教你如何思考。”

编程主要是人脑思维方式的映射。解决问题的时候，人的思维方式是需要完整性和逻辑性的，而通过编程训练，可以不断培养良好的思维方式，帮助人掌握逻辑思考的能力。

有人说，我会开车，但我并不需要知道车是怎么造出来的或是怎么修理的，同理，我会使用软件就够了，没有必要学编程。没错，我们并不是为了编程而学编程，就像我们学修车并不是为了去修车，而是通过学习简单的修车知识帮助我们更好更安全地开车和保养车。

从心理学上来讲，全程专注于一个目标，能够享受到做事情带来的满足和激情，会让人感受到做事情的乐趣，而编程，就是完全具备这种乐趣的事情，当你茶饭不思，努力思考一段代码为什么没有成功运行时，当你经过多次思考和尝试，最后把问题解决时，你的内心会有极强的满足感，这是一种非常棒的体验。



编程还是一个非常高效的、用于实现想法的工具，对于小孩子来说，乐高是他们塑造世界的玩具；对于成人来说，编程其实就是一个更加具备拓展性的“乐高”。

### 1.1.2 兴趣是最好的老师

想想看，你是否有过这样的经历：

喜欢打游戏的——凌晨三点了，一点都不困啊，再来一局 LOL！

喜欢看小说的——时间过得好快啊，都五点了，再看一章《雪鹰领主》就睡！

喜欢看视频的——今天晚上通宵也要把《仙剑奇侠传》看完！

为什么我们在做这些事时一点也不觉得累，而是感觉时间过得飞快呢？

我想是因为做这些事情的时候，你是放松的、是消遣的，没有目标或者指标要求，没有压力，所以才会乐此不疲、废寝忘食。

那么如果是学习编程呢？你需要学习、需要思考、需要开动脑筋、需要练习、需要总结，这都是有力度的。

如何能够化解或者避免这些压力或者疲倦呢，我想只有自己真正地喜欢它、爱好它，才能不觉得累，才能坚持下去。

学习知识最重要的是培养学习的兴趣，俗话说：“兴趣是最好的老师”。对知识的学习感兴趣，就会变被动为主动，以学习为乐事，在快乐中学习，既能提高学习的效率，还能够加深对知识的理解，这样学到的知识才能够灵活地运用。

学习编程，一定要从兴趣出发，给自己定一些确定的目标，比如说：发布一个自己的个人网站、做一辆遥控的小车、将家里的门锁换成由手机控制的、给家里的鱼缸做一个远程喂食系统，等等。

这样带着目标去学习，带着实际问题去学习，会比毫无目标的学习要更有动力，更有效果。

### 1.1.3 为什么零基础的自学编程者，大多半途而废

笔者有一个认识很多年的好友，老赵。有一天打电话给笔者说，他女儿在上高中，有兴趣学编程，但不知道从哪里着手，希望笔者能从技术方向上提供一些学习路径建议。然后笔者就



刷刷刷列了提纲：

- 技术路径：Html > CSS > JavaScript > PHP > MySQL
- 学习内容：w3school.com.cn 网站
- 学习方法：跟随网站教程逐步学习，掌握基础知识后练习一些模拟项目

从一个做技术的程序员角度来看，这个路径是比较适合零基础的初学者的，但是过了一段时间之后，老赵又来电话了：女儿按照这个方法学习了一段时间之后，发现学不下去了，就好像知道机器的每一个零件，但就是组装不出机器来。

仔细分析之后，有点明白了：学习网站的内容大而全，不容易分清主次重点；不知道如何融会贯通多个技术点完成一个实际项目；缺少专业辅导碰到问题不知道如何解决。

因此，笔者一直在思考，有没有一种更好的方式来学习编程呢？有没有一种更有效的路径来学习编程的入门知识呢？

#### 1.1.4 有趣好玩很重要

最近在给一所小学的四五年级的学生做公益编程兴趣课，看到这些孩子渴望的眼神，开心的笑容，真是很受感染（见图 1-1-1）

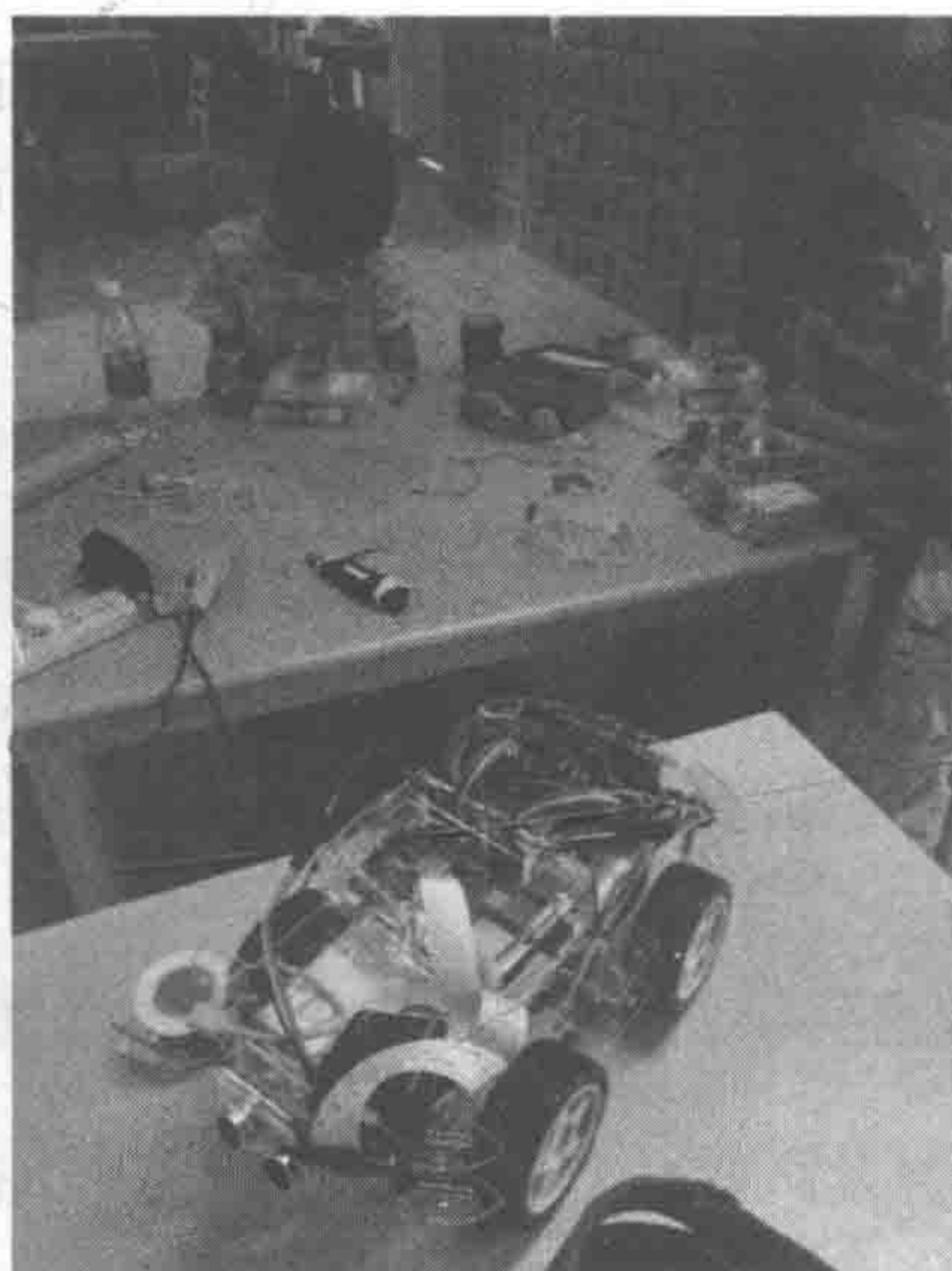


图 1-1-1.



这些小朋友只有 10 岁左右，但从上课的效果来看，学会编程是完全没有问题的。

感受最深的一点是：理论知识一定要讲得有趣，结合生活中的例子效果会更好，结合硬件多动手效果最好。

### 1.1.5 家长带着孩子一起学是最好不过的

中国的很多家长，望子成龙、望女成凤，会花很多钱给孩子报很多培训班，却很少花时间陪孩子做他们喜欢做的事情，但其实陪伴比学习对他们的影响更大。

编程，就是家长可以和孩子一起学习一起成长的一种方式。想想看，当家长和孩子一起时，每完成一段代码，每解决一个 bug，每一次让小车跑起来，都会是满满的开心和收获！

考虑到这样的目的，笔者在设计教程时，就是按照 10 岁孩子的理解能力作为基准，尽量将高深晦涩难懂的名词以及技术，通过浅显易懂的语句，使其变得更容易理解。

同时，通过软硬件结合控制显示效果的方式，让编程显得更有趣，并且尽量将软件工程的观念贯穿整个教程中，让大家逐步具备软件工程的思维方式。

另外，在学习过程中，交流和沟通是非常重要的的一环，有兴趣的读者可以加入 QQ 群（603559164）进行交流沟通。

### 1.1.6 当你决定出发的时候，最困难的一部分已经完成了

知乎上有个问题的答案很有意思。

问：为什么零基础自学编程者，大多半途而废？

答：大多数人的努力程度，自学不了任何东西。

一万小时定律，相信大家都听说过，学会编程入门知识和成为 IT 专家，是不一样的目标，所以我们不需要一万小时，但这并不是说学会编程就很轻松，学习本身就不是轻松的事情，不付出努力，任何事情都不会成功。所以，坚持是一件非常重要的事情。



### 1.1.7 计划的重要性

“一年之计在于春，一日之计在于晨，一生之计在于勤。”

做任何事，最好都要做计划。计划按照时间维度可以分长期、中期、短期。

笔者个人的体会，就是将大的目标通过计划逐渐分解为细微的目标，每完成一个细微的目标，都会给自己一点成就感，给自己不停地刺激，获得愉悦感、成就感，从而逐渐完成目标。

喜欢编程的朋友，先设立一个小目标吧：入门。

最后，转载一句雾老师的话：

我们读书，我们接受教育，我们向有智慧的朋友求教，一切的目的，都是为了赋予自己这样一种能力——无论时局如何变化，无论命运把我们丢到何等陌生的环境，我们仍然能够活下去，爬起来，站直了。

## 1.2 700 元的电脑真的可以学会编程

“工欲善其事，必先利其器。”本节来说明初学编程应该选择什么样的电脑，以及如何购买。

### 1.2.1 计算机（电脑）的历史

先来看看计算机的历史：

1946 年，美国军方定制了第一台计算机，占地 170 平方米，重量达 30 多吨。

1946—1957 年，第一代，电子管计算机。

1957—1964 年，第二代，晶体管计算机。

1964—1971 年，第三代，中小规模集成电路计算机。

1971—2015 年，第四代，大规模和超大规模集成电路计算机。

2015 年—？，第五代，具有人工智能的新一代计算机，它具有推理、联想、判断、决策、学习等功能。