

最全的儿童编程软件介绍+汇总，值得收藏！

玩转单片机 6月9日

立创商城 WWW.SZLCSC.COM

一站式电子元器件采购自营商城

- 100%原装正品
- 支持小批量和零售
- 10000多平方米元器件仓库
- 现货库存超100000种

注册立享
15元优惠券
无门槛使用

大家在给孩子选编程软件之前，一定要知道教育性质的编程语言如Logo、Scratch、Etoys、Alice等和专业的编程语言如vc、Pascal、VB、Java、.net等有什么不同。

一个教育性质的程序设计语言，是一种主要设计于用来学习的工具，是与用于编写真实环境下的应用程序编程语言不大相同的一种编程语言。

许多教育性质的程序设计语言都提供建议的学习顺序，即是学习一组编程语言的顺序，每种语言都建立在其他语言的基础上，使得学生能够完成从简单又易于理解的娱乐性编程环境到专业开发环境的转变。今天超人来给大家分析分析几款流星的编程软件。

| Logo

Logo是特别为孩子们设计的入门程序设计语言。学习Logo的第一步是认识“turtle graphics”（龟状图形，最早可以追溯到1969年，proto-Lisp中的“turtle robots”）。



在现代的Lisp实现中，有一个叫“turtle”的抽象绘图设备，专门用来绘制龟状图形，是为了吸引孩子们进行编程。

Seymour Papert，Logo语言的发明者，是构造论的支持者，构造论是多种构成主义的学习理论。Papert认为像写作之类的活动，会在早期接受计算文化过程中自然而然的学会。所以Logo设计的初衷不仅仅是为了教会孩子们编程和计算文化中的一些概念，而是为了增强孩子们对日益以科技为主导的文化的适应。



“比开始较早进行智力教育更为重要的是，这能够使孩子们避免学习中形成的长期依赖，即学习的东西都是由所谓权威的人提出的...而这样的小孩无法给自己准确的定位，或者让社会觉得他们是在智力上是无助的”。

通过在3岁左右的孩子进行实验，对他们30年的跟踪记录发现，大多数都取得了教育上的成功。Logo实际上是一个Lisp的线性版本，对于高年级的学生而言，Logo能够用来解释计算机甚至是人工智能中的一些概念。

Brian Harvey写过一系列的文章“Computer Science Logo Style”，对那些通过Logo学习计算机科学的学生而言是很有帮助的。Logo目前已广泛应用于各种平台，同时提供了免费版和商业版。

Scratch

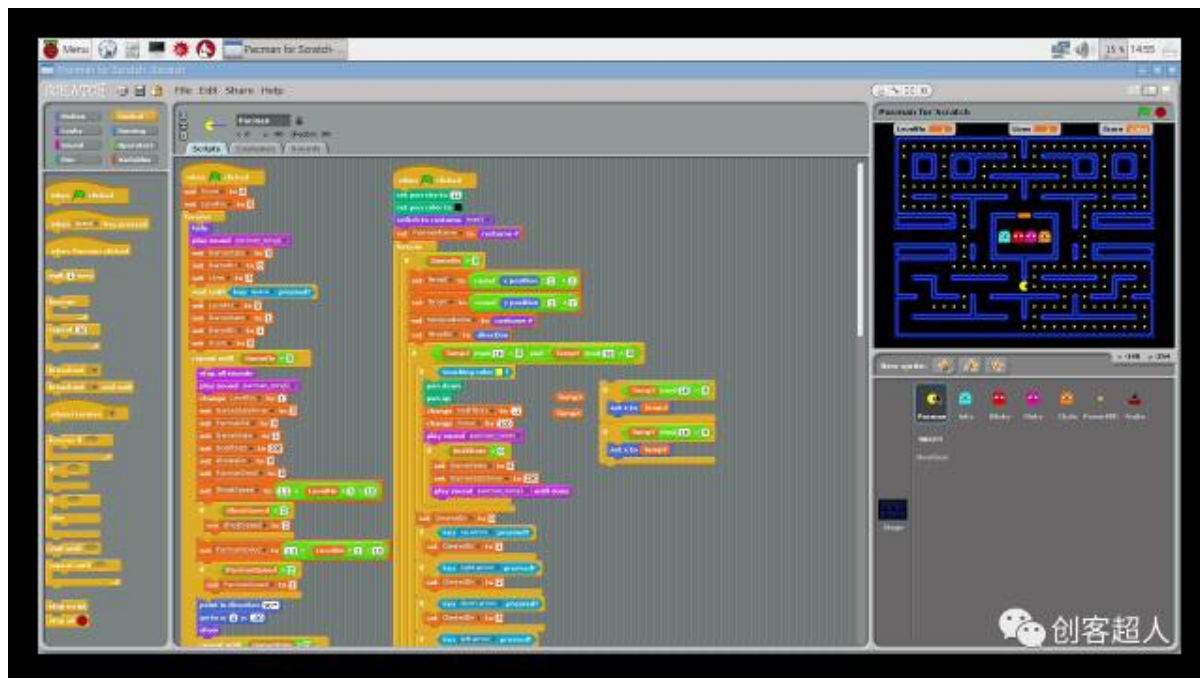
Scratch由麻省理工学院的 Dr. Mitchel Resnich 和他的“终身幼儿园（MIT's Lifelong Kindergarten research group）”研究组成员共同完成。



Scratch 的设计灵感来源于教小朋友们编程时遇到的种种困难。它的名称来源于DJ混录唱片的技术。（#大概是觉得DJ混唱片、搭乐高积木，用Scratch编程某种程度很相似。）

Scratch：让编程变得像搭积木一样。过去，很多教程里总是喜欢拿方形、圆形等来帮助我们来

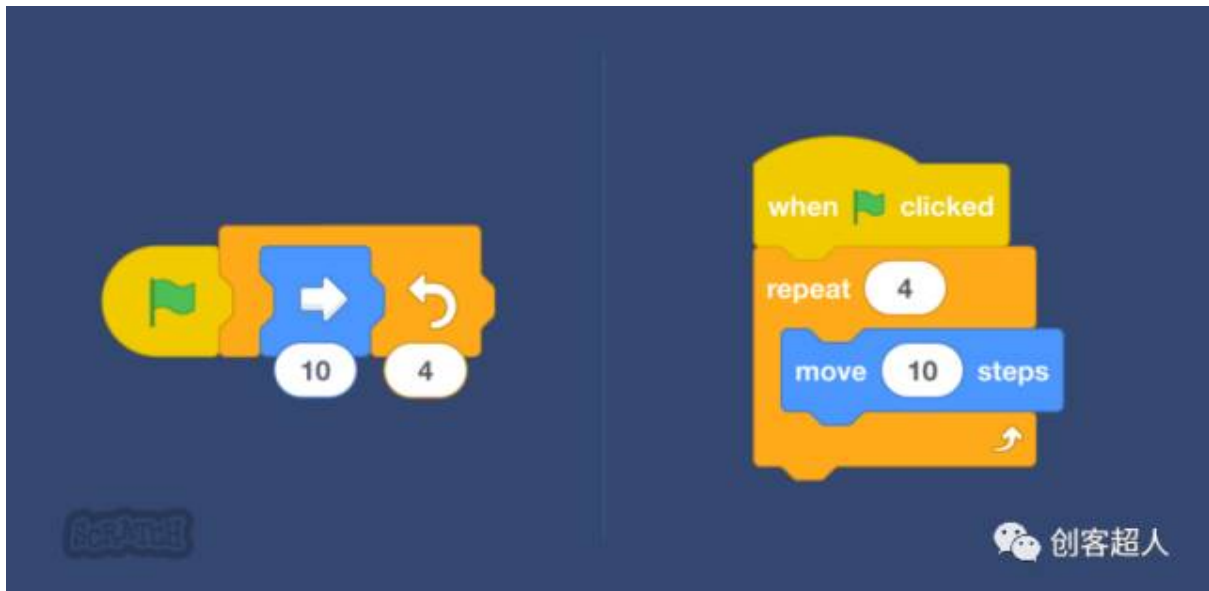
解高深的对象、实例等等编程理论。但总是越说越让人糊涂。



现在，Scratch出现了，一切都将不同！编程从此将会和搭乐高积木一样简单！Scratch现已提供基于Windows以及MAC版的下载，Linux版的也正在研发当中。编程界面分为操作符、变量、代码、图片、声音、程序等部分。

编程命令按照动作、感觉等形象的进行分类，并且这些编程命令均可支持拖拽。成套的逻辑语句利用突出或特殊形状等形式来辅助编程。

例如：“IF...THEN...ELSE...”是一个黄色的呈“E”状的图形块，其中有一个菱形的洞供填写条件，在“E”的三横中间可以加上不同的操作。编程过程中，用户可以添加来自本地或者Web的图片和声音，当然，添加的方式一样是拖拽。



Scratch 可将项目文件存为本地的“.sb”文件格式，并通过一个按钮将其上传到Scratch主页上。其它用户借此可对该项目进行评论。此外，也可从Scratch网站上下载一个叫做“.SB file directly”的 Java applet，借其可将Scratch项目植入任一站点。

Scratch 是软件基础教学的一次革命。让那些厌烦编程的人眼前一亮。Scratch使编程容易上手、充满乐趣，但却不失编写复杂应用程序的能力。

| Small Basic

微软Small Basic作为一个入门程序设计语言，是Visual Basic的简化版本。“旨在把‘乐趣’带回编程中”。Small Basic实在是小，只有15个关键字，并且每个都自然好懂。



创客超人

Small Basic提供特别的对象函数库，包括使人普遍感兴趣和孩子们感兴趣的东西（例如 Flickr）。孩子们能够在线或在本地创建有趣的交互程序。Small Basic利用了微软的Visual Studio IDE提供的自动完成和上下文相关的帮助。

| Scheme

Scheme最初是在1975年出于教学目的而设计的语言（参加[9] LISP of the day used non-recursive control structures to implement lambda calculus以非递归的控制结构实施λ演算），这主要是因为Lisp仍然在硬件实现具有高效率。

The Scheme Programming Language

Greg J. Badros

`badros@cs.washington.edu`

University of Washington, Seattle

CSE-341, Summer 1999

(C) 1999, Greg J. Badros—All Rights Reserved

23-June-1999

G. Badros - CSE-341, Scheme

0

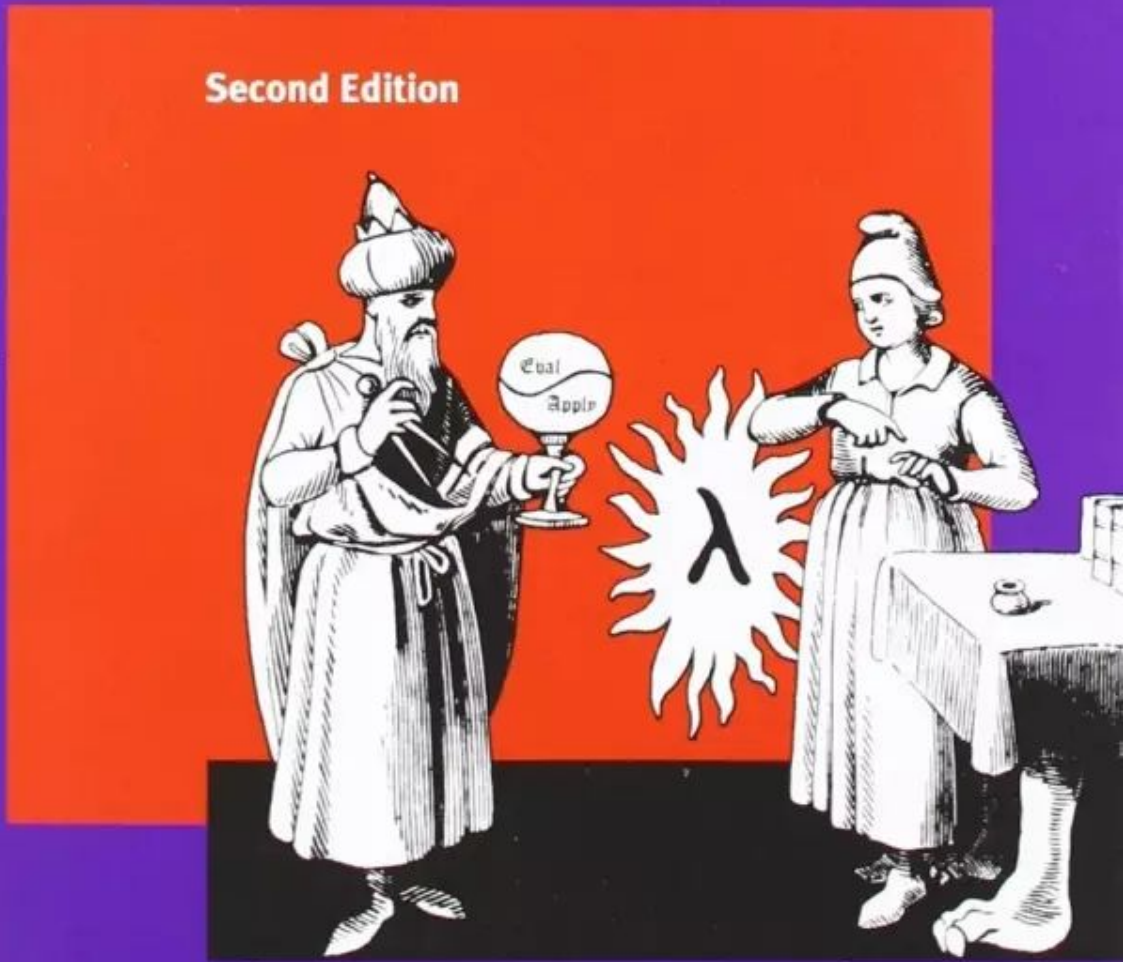
Guy L. Steele, Jr和Gerald Jay Sussman创造出了Scheme，通过使用简洁高效的Lisp过程调用，使得Scheme成为最快的解释型语言之一。

从语言教学上来讲，可以通过以 domain-specific languages (DSL, 特定领域语言) 和 Metacircular Evaluator(元循环计算) 的方式进行。

1984年出版的《Structure and Interpretation of Computer Programs》（《计算机程序的构造和解释》）将这种教育理念纳入了经典的教学课本中，从而改变了Pascal在大学程序设计语言中的优势地位。

Structure and Interpretation of Computer Programs

Second Edition



Harold Abelson and
Gerald Jay Sussman
with Julie Sussman

创客超人

“Abelson 和 Sussman 在 1984 年出版的《Structure and Interpretation of Computer Programs》(SICP)，给上世纪八十年代的计算机课程设计带来了彻底性的改变。

更为重要的是，SICP打破了语法入门课程的垄断，不再是将课程的组织仅仅围绕某个当前流行的程序设计语言的语法，而是将学习计算科学的重点放在了：

functional abstraction(函数抽象)，data abstraction (数据抽象)，streams (流)，data-directed programming (数据导向编程)，implementation of message-passing objects (消息传递对象的实现)，interpreters (解释器)，compilers (编译器)，and register machines (寄存器机器)。”

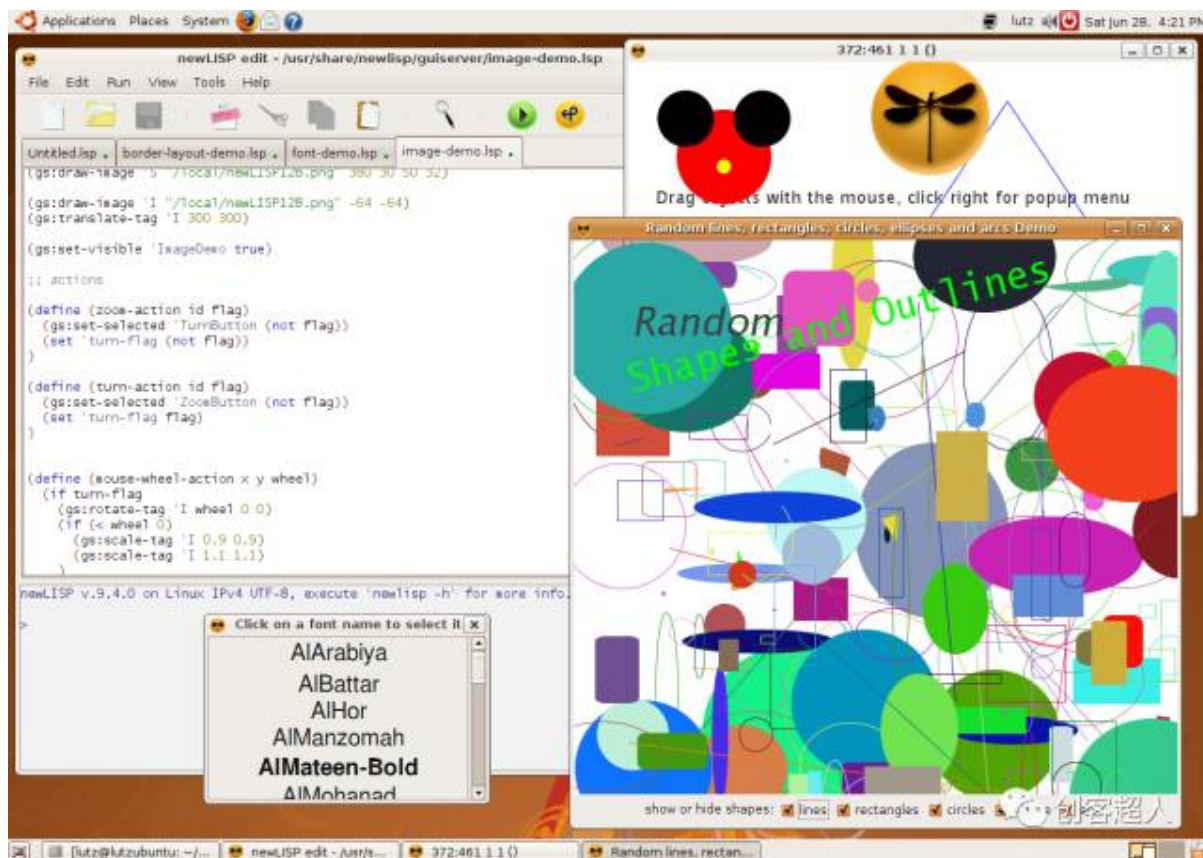
短时间内，美国以及世界各地的很多大学把第一门计算机科学课程改成了SICP和Scheme。而SICP也成了MIT Press最为畅销的图书之一。

和SICP一起，Scheme也广泛应用，不仅仅是印第安纳大学，MIT，耶鲁大学的开设的少数课程，而是成为世界性入门语言的选择。

自那以后，Scheme社区为低年级课程提供了一些定制的教育编程环境。其中有PLT Scheme和DrScheme环境，还提供了免费的教师培训教材：How to Design Programs 和 TeachScheme!。

| newLISP

newLISP旨在提供一个磁盘占用和系统资源使用少的，快速的，强大的，跨平台的，全功能脚本版的Lisp程序设计语言。因为它的简单，一致，线性的特点，特别适合于初学者。



newLisp能够最大限度的减少学习的曲线并且带给程序员更多的创造性和快乐。

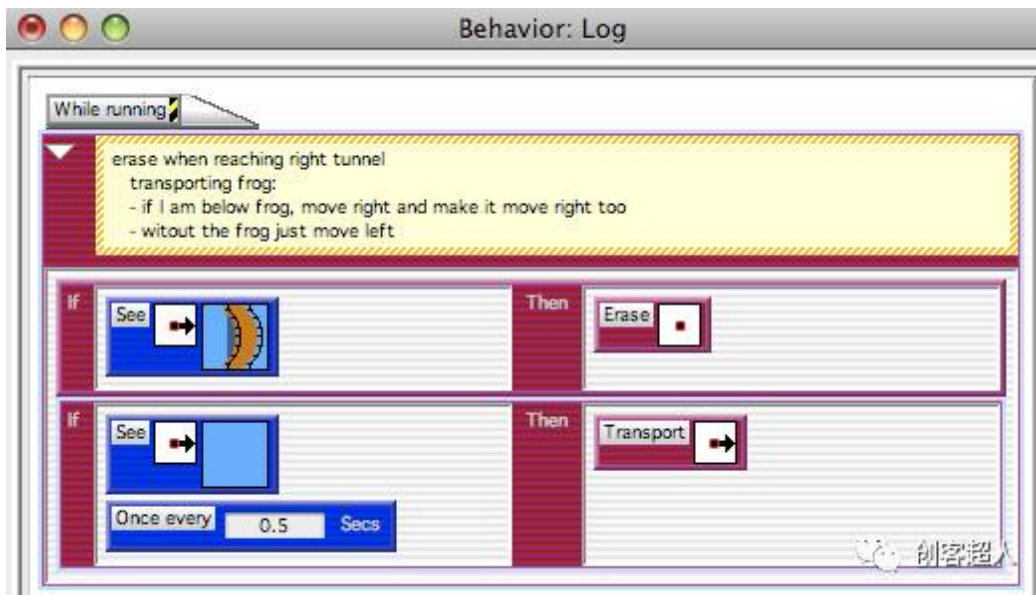
CommonLisp 是通用的（专业的），多泛型程序设计语言。它支持过程式、函数式和面向对象的编程范式。

作为一个动态程序设计语言，CommonLisp促进了迭代编译向高效运行时程序的快速发展。CommonLisp和大部分其他程序设计语言不同，它使用S-expressions（S表达式）来表示代码和数据结构，而函数和宏调用是用链表实现的。

通过Lisp的标准化统一各种Lisp的变体已经提上了日程，这不是一个特定的实现而是一个语言规范。现在有几种CommonLisp的标准实现，包括商业产品和开源软件。

AgentSheet

AgentSheet是屡获大奖的游戏和仿真制作工具，它可以简单到让中学生通过制作视频游戏来学习计算机知识，也可以复杂到让NASA的科学家来模拟航天飞机的有效负荷。



AgentSheet是由一个叫做Scalable Game Design（可扩展的游戏设计）的完整课程提供支持的，它从简单的像青蛙一样（Frogger-like）的游戏一直到复杂的人工智能仿真游戏。

AgentSheet支持游戏（动画，交互，声音，语音合成/识别（Mac）），科学应用程序（绘图，输出到电子表格，3D绘图（Mac））。AgentSheet有英语，希腊语和日语版本。

Lego Mindstorms以AgentSheets为基础，开发了LegoSheets程序设计语言，比Brick Logo的学习曲线要平缓的多。

| Alice

Alice是一个免费的编程软件，旨在教会孩子们事件驱动的面向对象编程。编程者可以使用现代IDE，以拖拽的方式创建交互式的故事程序。



Alice目标群体是大多数中学阶段的有计算机使用经验的女生，可谓寓教于乐。还有一个叫

Story Telling Alice的Alice的变体，能够提供更强的讲故事能力。

| Baltie

Baltie是图形化的编程工具，适合儿童，青少年（成人）。Baltie是软件中一个主要的角色，一个巫师，用来执行各种命令、将出现在他视野的图片变魔术。

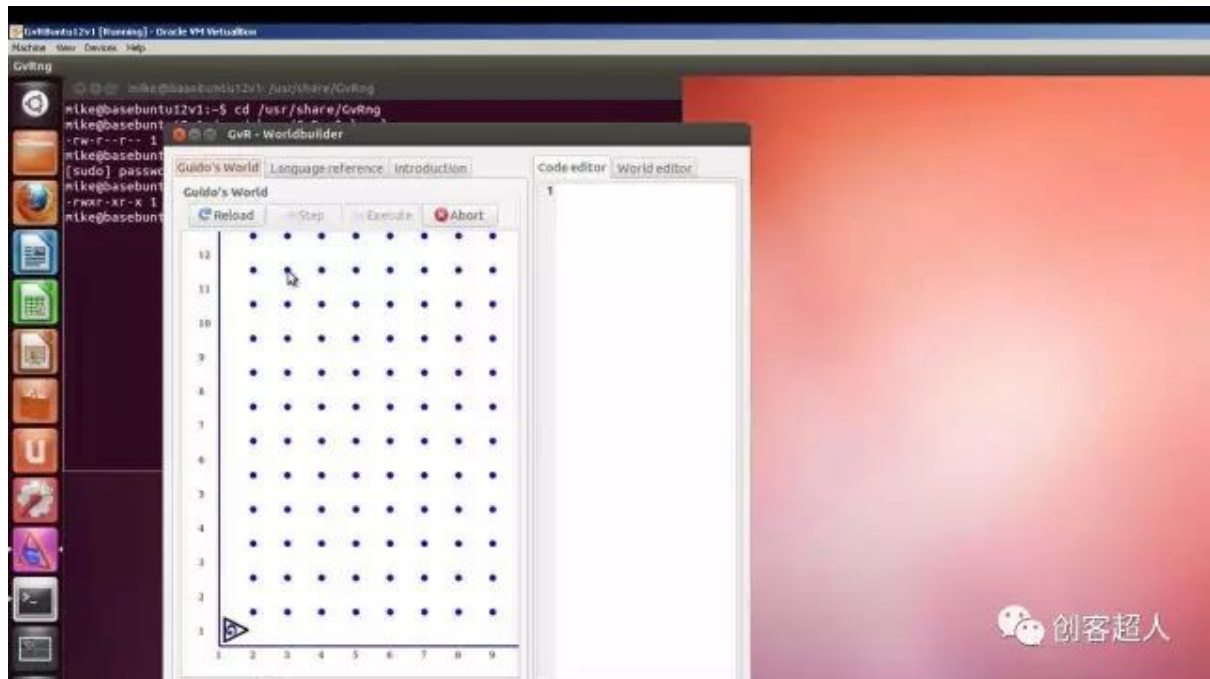


通过Baltie的帮助，孩子们能够很快认识到什么是计算机，如何对计算机进行操作和编程，这些都是玩的过程中实现的。Baltie同样能够帮助逻辑思维训练，而且对孩子们的知识并没有多少要求，只要求会玩和想象力。

Baltie已经在许多国家的小学中采用了。新版本的Baltie 4甚至完全支持C#，更多的信息可以在SGP Systems中找到。

| Guido van Robot

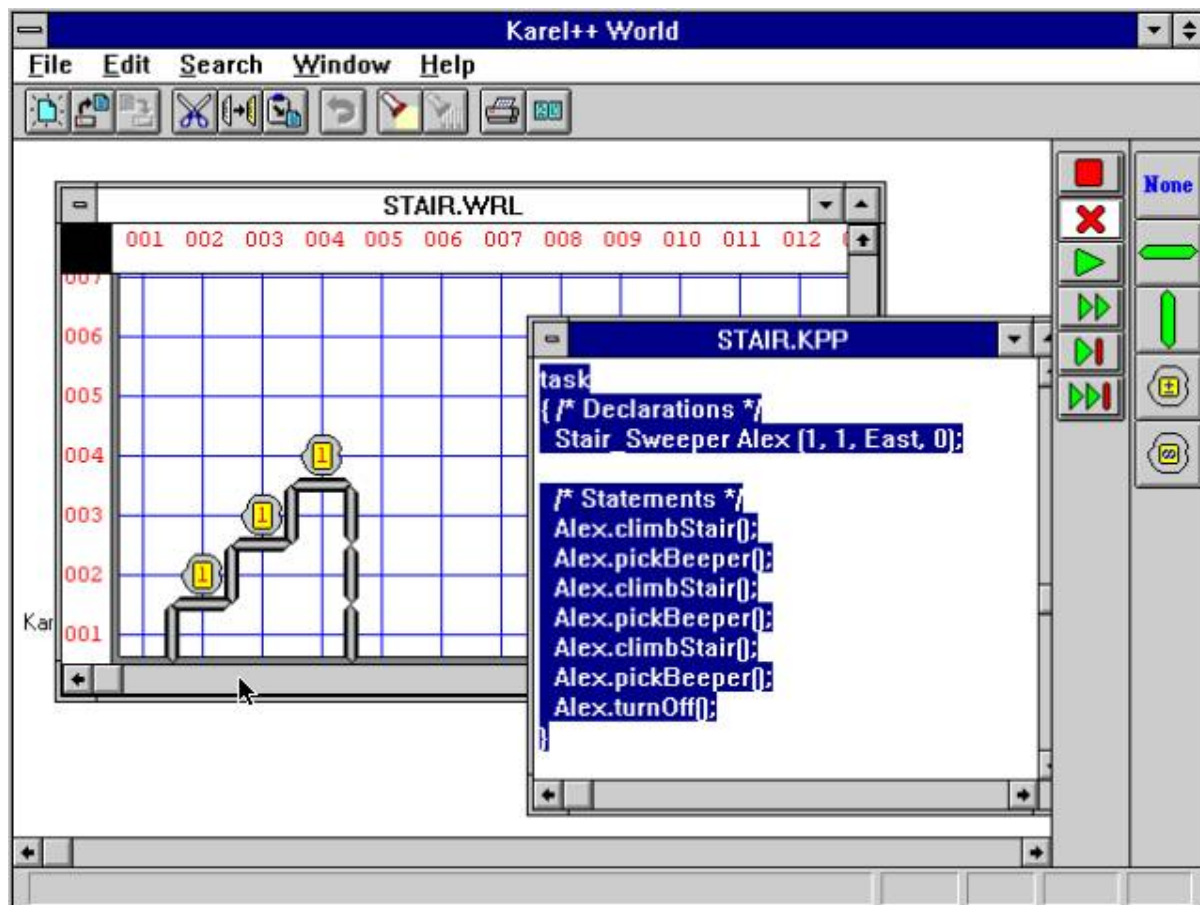
Guido van Robot是一个机器人控制程序，有点类似于Logo和Karel J. Robot，使用了Python语法中的很少一部分。



Guido van Robot被设计成对任何高级语言都尽量简化和通用。还有一个叫RUR-PLE的变体，它包括了完整的Python语法和一些经典的课程。

| Karel , Karel++ , 和Karel J.Robot

Karel , Karel++ , 和Karel J.Robot是为那些完全的初学者而准备的。用来在一个由长方形街道网格组成的城市中控制一个简单机器人。Karel是一个自主的程序设计语言，而Karel++是Karel的C++实现，Karel J. Robot则是Karel的Java实现。



Learn to Program BASIC

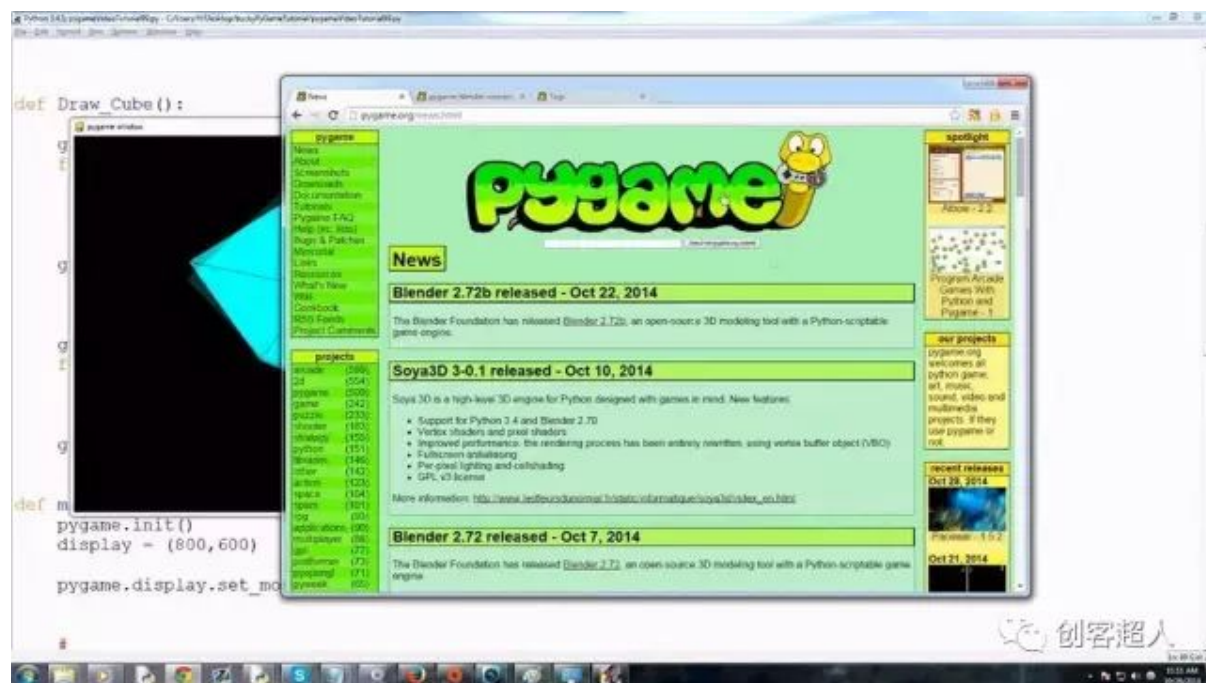
Learn to Program BASIC是一个通过互动课程教会中学生Basic的Basic解释程序，在Basic语言中加入了一些的游戏，包括对2D sprite支持。在LTPB中编写的程序能够脱离LTPB独立执行，即是一个可以自由分发的“runner”。



Phrogram

Phrogram（第二代的KPL，MS Kid's Programming Language）是2006年引入的一个商业的、易于学习的程序设计语言和集成开发环境。它强调图形和声音，使得它特别容易开发游

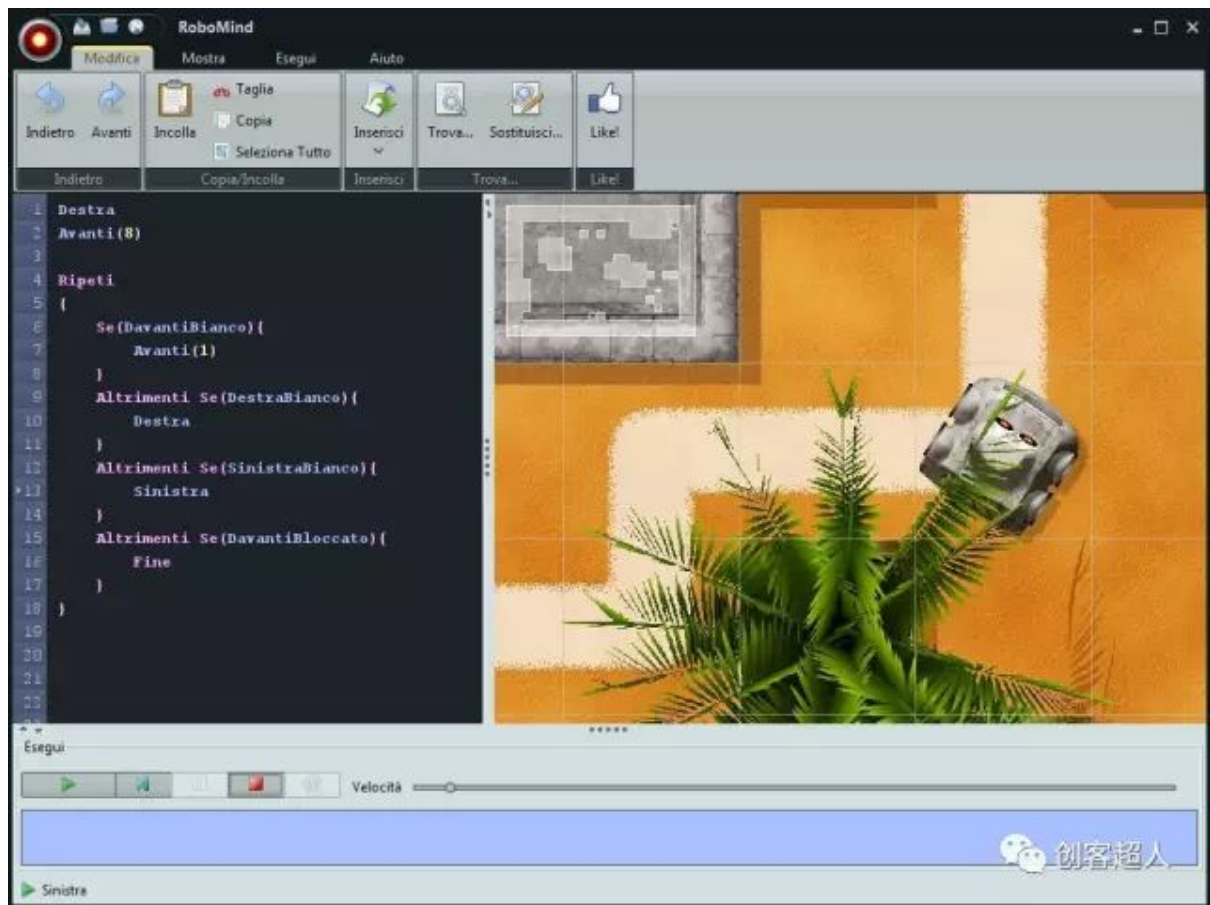
戏，寓教于乐的程序。



Phrogram是简化的结构化语言，提供了基于组件开发的特性，例如类和方法。Phrogram在现代IDE中进行建模，例如Eclipse 和Visual Studio. NET，并可以帮助初学者为这些或者其他的专业开发环境做好准备。

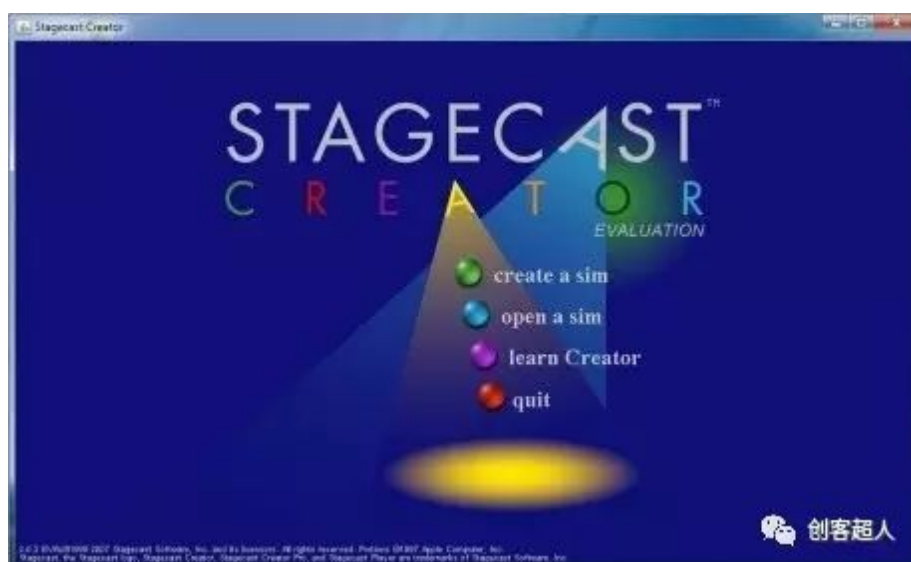
RoboMind

RoboMind是一个简单的教育性质的编程环境，能让初学者对机器人编程。RoboMind介绍了流行的编程技术、一些机器人学和人工智能的知识。人们能够使用阿拉伯语、中文、荷兰语、英语和瑞典语对机器人进行编程。



StagecastCreators

StagecastCreators是基于示范编程的可视化编程系统。用户通过移动屏幕上的图标，为物体（角色）制定规则来操作整个系统。用户可以创造2D模型来模拟一个概念，多层次的游戏，交互的故事等。



| Etoys

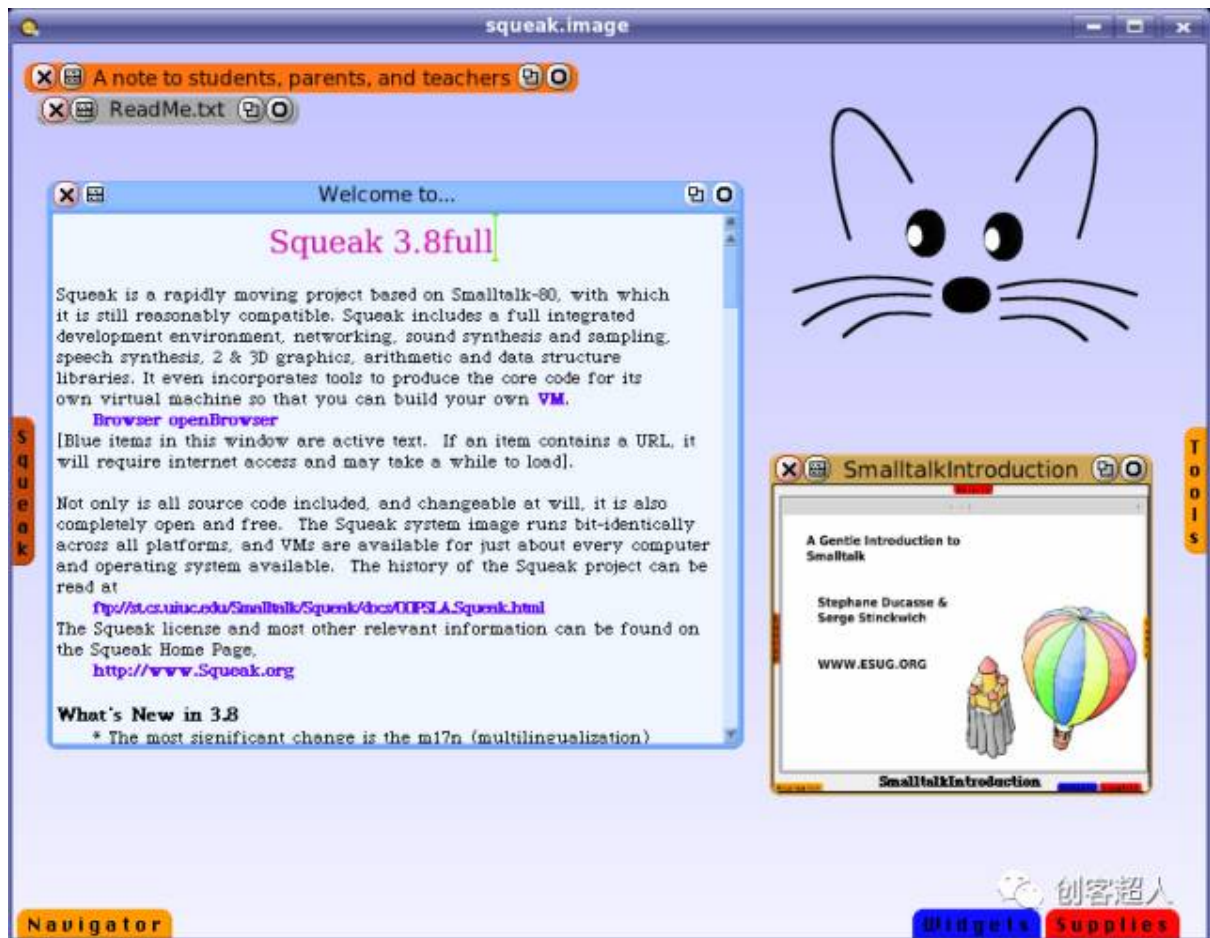
Etoys的创意来源计算机屏幕上那些可编程的实体。Etoys提供一个简单高效的脚本对象模型，和一个用户媒体丰富的创作环境，涵盖了2D和3D图形，图片，文本，粒子，演示文档，网页，视频，声音和MIDI音乐，能够与其他Etoys用户实时共享桌面，通过互联网进行身临其境的指导。



Etoys是多语言的，已经在美国，欧洲，南美，日本，韩国，尼泊尔等地取得了成功。这个项目是针对9至12岁的儿童的。

| Squeak

Squeak是一个现代的，开源的，功能齐全的Smalltalk程序设计语言和执行环境。Smalltalk是一个面向对象，拥有动态类型和反射特性的程序设计语言，用来支持“人机共栖”这个“新的世界”的计算典范。



像Lisp一样，它有影像式持久性（image-based persistence），所以通过语言本身能对所以任何事物进行修改（参见Smalltalk # 反射）。

Smalltalk在面向对象编程和JIT及时编译等方面给行业带来了深远的影响。Squeak是在广泛开发项目中使用的工具，包括多媒体应用，教育平台和商业网络应用程序等。

Squeak被设计为高度可移植的，调试，分析和维护亦是如此，这是因为Squeak的虚拟机完全用Smalltalk编写的，Squeak的主页维护着免费的教程列表，Stéphane Ducasse则维持着大量Smalltalk和Squeak相关的在线书籍。

现如今使用最广泛的，采纳了许多Smalltalk的设想的语言是Objective-C。

大家再看看搜到的一张表格，是现在比较流行的少儿编程软件，有详细解释：

| 年龄 | 经验水平 | 合适的编程语言 |
|----|------|---------|
|----|------|---------|

| | | |
|-----------|----------------|---|
| 学龄前 - 二年级 | 无/不适用 | Logo style: Logo, Guido van Robot, Karel. Scratch, Baltie 2, Stagecast Creator |
| 二年级 - 四年级 | 无/不适用 | Logo, Scratch, Etoys, Stagecast Creator |
| 五年级 - 八年级 | 很少或没有经验 | Lego Mindstorm, Etoys, AgentSheets, Alice, Baltie 3, learning oriented BASIC, Phrogram, Stagecast Creator , smallBASIC |
| 五年级 - 八年级 | 有一些经验 | Squeak, full featured BASIC, Greenfoot, Pascal |
| 高中 | 很少或没有经验 | Squeak, Greenfoot, Pascal, full featured BASIC |
| 高中 | 有一些经验 | Squeak, Greenfoot 或 BlueJ, newLISP, OZ. Most other programming languages. |
| 大学 | 不是主要专业, 假定没有经验 | Squeak, Greenfoot 或 BlueJ, newLISP, full featured BASIC |
| 大学 | 开始计算机科学或程序设计课程 | Haskell, OZ, Scheme, Qi, Squeak, NetBeans BlueJ |

本文转自网络，版权归原作者，如果您觉得不好，请联系我们删除！

广告

关于立创商城

立创商城 (WWW.SZLCSC.COM) 成立于2011年，致力于为客户提供一站式电子元器件线上采购服务，成交量全国领先。拥有10000多平方米现代化元器件仓库，现货库存超100000种，集团电子全产业链自营服务涵盖：在线EDA (LCEDA) + 行业领先的PCB打样/中小批量+元器件商城+钢网制造+SMT贴片+电子设计教育及方案。作为一家品种齐全、自营库存、质量有保障的电子元器件垂直商城，立创商城所有元器件均从原厂或代理商正规渠道采购，保证原装正品，为您提供专业的一站式元器件采购服务。

[阅读原文](#)