



**作者简介:** 西摩·佩珀特 (Seymour Papert, 1928~2016), 美国麻省理工学院 (Massachusetts Institute of Technology, MIT) 教授, 是推动少儿编程的先驱。1967年, 佩珀特在 MIT 成立了儿童实验室, 致力于研究儿童是如何学习和思考的。经过十多年的探索和研究, Prentice Hall Europe a Pearson Education Company 出版社于 1980 年出版了佩珀特的专著 *Mindstorms: Children, Computers, and Powerful Ideas* (文中简称 *MindStorms*), 并于 1993 年出版第二版。2018 年, 新星出版社推出了由梁栋翻译的 *MindStorms* 一书的中文版——《因计算机而强大: 计算机如何改变我们的思考与学习》。该书基于皮亚杰儿童认知发展理论所提出的建构学习 (Constructionist Learning) 思想与“改变了全世界数百万儿童的创造和学习方式”<sup>1</sup> 的 LOGO 环境, 较为完整地阐述了佩珀特为儿童研发编程环境的用意。

## 儿童为什么学编程? \*



——*Mindstorms: Children, Computers, and Powerful Ideas* 一书的启示

汪 琼

(北京大学 教育学院, 北京 100871)

最近几年, 包括中国在内的许多国家又重新提出要普及儿童编程活动。儿童为什么要学编程、是否能够学会编程、应该怎样学习编程, 再次成为参与和推动儿童编程事业的人士需要思考和回答的问题。在 *MindStorms* 一书中, 作者佩珀特从两个方面阐述了儿童为什么要学编程: 一方面是分析、论述儿童应该采用的学习方式, 顺带批判学校教育破坏了儿童天生所具有的学习能力, 伤害了儿童的学习自信; 另一方面阐述了计算机不仅对人类的思维加工过程有工具性影响, 而且它还会影响人类的基本思考方式, 如使用计算机创造的微世界可以让儿童学习的天性得以发挥, 从而全面改善儿童与学习相关的人与事的关系。

### 一 提出儿童学编程的起因

佩珀特之所以要提倡和推动儿童学习编程, 与他花了 5 年时间跟随皮亚杰研究儿童认知过程有关。佩珀特接受皮亚杰关于“儿童是天生的学习者”的观察, 认同幼儿的学习方式是最有成效的学习方式, 其学习过程有着许多值得学校借鉴的特点: ①它是有效的 (所有的儿童都会); ②成本很低 (既不需要教师也不需要开发课程); ③人性化 (儿童似乎无忧无虑地做这件事, 没有明确的外部奖励和惩罚)。这种在与环境交互作用中产生的自然、自发的学习, 也被称为“皮亚杰式学习” (Piagetian Learning)。

伴随儿童年龄的不断增长及其思维不断发展, 周边的环境往往就不再具备继续支持他们自主进行高深知识探索的条件, 这时候他们就需要进入学校, 以开展更专业化的学习。佩珀特理解学校出现的必要性, 但又批评现在学校的教育内容和文化让儿童失去了学习的兴趣和能力:

<sup>1</sup> 这是佩珀特教授逝世后, MIT 讣告中的第一句话, 参见:

<http://news.mit.edu/2016/seymour-papert-pioneer-of-constructionist-learning-dies-0801>

## 1 学校教育提供的是抽象知识，较少能与学生的体验建立联系

以物理学为例，学生在课堂上学到的都是用数学公式来表达的定理定律，学习的过程就是做题、计算，学习效果主要看考试成绩。即使是物理实验课，也是在验证公式定律，而不是让学生去自主发现。在这种情况下，学生可能学了几年物理，但对物理学的基本概念有可能还没弄明白，这就很难在现实生活中灵活地运用物理学知识解决问题，更别说创造性地发展物理学了。佩珀特把这样的学校教育称为“分离的学习”（Dissociated Learning），即学生对学习这样的学科知识没有兴趣，更别谈动力。

佩珀特强调：“定律和公式不是物理学的全部，科学的命题内容固然重要，但它只是物理学家知识体系的一部分。这些定律和公式不是历史上最先发展起来的部分，也不是在学习过程中必需首先理解的部分。”换句话说，学校教育在决定什么年级应该学什么知识的时候，往往考虑的只是学校所能够提供的学习条件（如在纸笔的条件下需要学习短除法，但有了计算器之后，美国很多学校都只要求学生会用计算器计算即可）和学生所具有的知识水平（如物理学之所以放到中学来学，就是因为学习和运用物理公式需要用到解方程的数学能力）。这种僵化的学习内容安排顺序违反了知识发生和发展的过程，人为地造成了学生理解的困难和僵化的学习。如果要让学生真正理解物理学，还需要告诉学生科学家发现物理定律的思维推导过程，或者创造条件让学生去亲身体验和发现规律，而基于计算机可以创造出支持这种探索的微世界。

## 2 学校的教育否定儿童的直觉思维，“对错文化”挫伤和损害儿童的自信

佩珀特认同皮亚杰关于“儿童是他自己智力结构的建造者”的观点，有很多证据表明儿童会自己建构对事物的解释。如有儿童认为风是树枝摆动的结果，因为手在脸庞附近挥动时，就会产生很明显的风感。如果对提出这一理论假设的儿童说：“这个想法很好，但这个理论是错的”，估计大多数儿童都会因此认为自己建立的理论是徒劳无功的。学校就是经常这样通过强迫儿童接受“正确的知识”，来完成人类文明所传承的任务。

在“学时有限”的情况下，学校很多学科都不教学生这些知识是怎么来的，只要求学生记住和会用。长此以往，学生就会认为做题只有做对和做错两种结果，而养成了按照标准答案回答问题的习惯，会坚信在学习过程中个人的认识和理解只要与标准答案不同，都是不可取的。而且，当儿童因为不理解而记不住、学不会的时候，就可能被周边人甚至自己暗示为不具备学习这门学科的“天赋”，从而失去自信，变得愈发不爱学习、不想学习。佩珀特认为这就是为什么“天生的学习者”会慢慢变得不会学习的文化根源。而要改变学校教育的沉痾，就需要从改变学校教学的理念和学习环境入手，溯本清源，去创建适合学生学习的环境。那么，什么样的学习环境才能既保护儿童的学习天性，又能支持他们学习高深的知识呢？

佩珀特是一位通过对自己的学习过程进行反省而对学习发生机理有诸多领悟的认知科学家，他之所以要为儿童研发编程系统，也是因为他幼儿时期痴迷齿轮变速器，这段经历让他认识到儿童在思考时需要有一个可以辅助其思考的“中介物”（Object-to-Think-with）。佩珀特回顾儿时借助齿轮学习的过程，总结出理想学习环境有四个关键特点：①这些齿轮是他生活环境中可以接触到的，且像玩具一样可以由他自己做主怎么玩。佩珀特认为儿童只要动手做，就能够“发现”和“掌握”事物运行的规律，虽然未必说得清楚背后的科学原理，但是动手的经历足以帮助儿童领悟这些原理。②这些齿轮是他周围成人世界的一部分，因此建立了当时还是孩童的他（或她）与成人的关系。换言之，儿童是通过使用成人世界的物件，采用类似成人的学习方式

在学习。如果在这样的环境中专家和新手都在其中学习，就更为理想了——儿童（新手）可以通过观察和请教成人（专家）而不需要通过课程学会所想学的一切。③即使是幼儿，也可以通过想象自己的身体转动来体会齿轮的转动，由身体知识（Body Knowledge）来推导和思考齿轮运转系统。由此，佩珀特提出学习应该是协同的（Syntonic Learning），直觉、感觉和情感与认知一起发挥作用。④因为齿轮之间的关系本身蕴含着很多数学知识，所以他能够用齿轮来思考形式化系统，齿轮充当着“思维中介物”，帮助他成长为数学家和教育研究者，而且并不影响他对实物的直觉思维过渡到形式化思维。

佩珀特赞同 Alan 有关未来“电脑如同铅笔”的愿景描述，并深受启发：铅笔既可以用来写字，也可以用来乱涂乱画；既可以用来做随意的笔记，也可以用来完成正式作业。总之，未来的电脑应该可以被随意且个性化使用，以完成多样化的目的。佩珀特谈到他之所以要研发 LOGO 系统，就是希望用计算机创造“简化世界复杂性”的微世界，让每个学生都可以在这个世界中通过操作“思维中介物”（如 LOGO 中的“海龟”）开展自主性探索。编程语言是学生操作微世界的工具，是计算机在教育领域发挥作用的一种形式。佩珀特研究少儿编程环境的目标，就是要支持儿童自己设计出可以操作的物件，让儿童在操作这些物件的过程中建立所学内容与个人体验的关系，享受学习的乐趣，就如他小时候使用齿轮创造了很多学习的快乐一样。

## 二 儿童学编程的价值与意义

在通过编程来教计算机上的“思维中介物”行动的时候，儿童首先需要想清楚自己是如何思考的，并需要用清晰的语言将其表述为计算机可执行的指令。计算机编程语言是儿童表达思维步骤的理想工具，不仅可以帮助儿童整理思考逻辑，也可以帮助儿童了解自己的思考过程。当思考逻辑以程序执行结果显性化表示的时候，也就更容易发现其中存在的问题，从而更易于修正错误。佩珀特在书中用很多案例解释了学习编程可以给儿童思维发展带来的诸多好处：

### 1 分层结构化思想有助于隔离错误、诊断问题

LOGO 语言是高级程序设计语言，它允许儿童编写子程序，用结构化的逻辑从整体到部分分层思考，这样可以控制复杂的思维。但佩珀特也指出：许多孩子一开始会抵触结构化编程，因为直线思维更符合儿童做事逻辑，直到有一天自己编的程序实在太长而难以找到错误时，他才会转向使用子程序。一旦养成结构化思维习惯，他就会体会到这种分层思考的好处，不仅可以“俯视”系统，还可以重复使用已有的程序，从而做出更庞大、更复杂的系统。佩珀特指出编程环境并不强迫儿童一定要采用“正确的方式”（如结构化编程）编程，而是给他们足够的灵活性和力量，等待其通过自己的探索而觉悟——这样宽松的环境更适合不同儿童的发展。

### 2 编程经历让儿童不怕犯错，并学会纠错

前面谈到学校教育文化让学生对于做错题感到沮丧，而学习编程会让他们体会到：很少有人第一次编程就能成功。学习编程的经历让儿童学会面对错误，如调试程序就是分析程序运行结果，找到错误所在并予以改正的过程是最有意义的学习。计算机编程的经验比任何其它活动更能有效地引导儿童相信“失败是成功之母”。佩珀特也谈到由于传统学校教育文化的影响，儿童刚学编程的时候会碰到程序出了一点小错，往往不是去查原因，而是删除这段程序，重新来写，这种不找错误根源只是重复尝试的做法是低效的。儿童学编程的过程也是纠正其对“错误”认识的过程，这种转变会让其受益终身。

### 3 程序化思维可以帮助儿童运用启发式方法解决问题

佩珀特指出，程序语言非常适合描述操作过程，如解题步骤。在 *MindStorms* 一书中，佩珀特多次使用通过分解计算步骤的方式展示运用计算思维（Computational Ideas）解决非程序问题的例子。如有儿童采用分解步骤的方式，找出了成功实施三球杂耍的关键要点（眼睛要盯着抛出球的最高点）和踩高跷的要点（高跷和腿要联动），这样的例子佐证了佩珀特提出的观点：程序化编程思想可以推广而成为学习原则，即思考如何通过对复杂过程的适当描述，来协助学习过程中问题的诊断和策略性调试，从而促进学习。程序化思维是一种强大的智力工具，一旦掌握后，就可以迁移到其它领域，无需计算机，也可以帮助学生思考、辅助解决问题。就像他当年学代数的时候，头脑中想着齿轮系统就可以，不需要手上真的有齿轮。现在儿童在学校中接触和运用程序化思维的机会并不多，学习编程是习得并运用程序化思维的机会。

### 4 计算机支持学生创建实现和检验其直觉假设的实验场

佩珀特在 *MindStorms* 一书中发展了对皮亚杰儿童认知发展阶段的解释，认为皮亚杰描述的是在自然环境中儿童认知发展的阶段特征规律。借助计算机系统创造的微世界，提供儿童建构知识结构所需要的材料和支持儿童建构学习的合适环境，就有可能缩短每个认知发展阶段的时长。而且，让高深的科学知识从大学下放到中学甚至小学也是可能的，因为很多伟大的科学思想，其原理和本质都是非常简单的，与儿童的直觉非常接近。如儿童凭直觉用 LOGO 小海龟画圆的过程（To Circle Repeat [Forward 1 Right 1]）就体现了微分几何的思想。佩珀特在书中用几个例子说明人的直觉思维孕育着伟大的思想，但是学校的课程只有干巴巴的事实和技能，缺乏对儿童直觉思维的保护和培养。LOGO 这样的编程环境可以帮助儿童使用直觉建模，而且过程化思维有助于表达直觉和修正直觉。通过建立自己的微世界，儿童在探索过程中不断建立起自己的认识模型。根据皮亚杰的同化理论，学习任何事情都可以变得很容易，只要能够将新学知识与已有的模型同化。所以，学习 LOGO 的很多儿童后来在学校的成绩全面提升也就不奇怪了。

总之，通过编程，儿童能够看到自己努力的成果，并接受一开始可能会不完美，不全对也不全错，而且可以在不断修改中日趋完善，这对儿童的成长意义重大。儿童在这个过程中不仅学到了数学知识，还学到了许多可以受益终身的、关于学习的知识。

## 三 儿童学编程的启示

*MindStorms* 一书之所以会产生这么大的影响，是因为它探讨的是教育领域应该如何使用计算机，才能更有利于儿童的学习与发展。佩珀特认为可以将计算机作为儿童的思维辅助工具，改变现在学生在学校的学习方式和思考方式。书中多次抨击那种让计算机成为“题海战术”帮凶的用法，认为那是在“强制向学生灌输前计算机时代（Precomputer Culture）遗留下来的难以消化的材料”，是“计算机对孩子编程（the Computer is being Used to Program the Child）”，他更希望看到的是“孩子对计算机编程（Child Programs the Computer）”。佩珀特也承认让儿童学习编程，只是在教育领域发挥计算机作用的一条路径。佩珀特所在团队经过十多年的研究，证明了这条路径的可行性，看到了因编程体验而带来的儿童对于学科知识的感悟（如通过教计算机作诗而理解主谓宾）、学习态度的变化（如觉得数学不是那么难了）和学习能力的提升（如将程序化思维用于指导运动技能的学习）。

编程语言有很多，并不是所有的语言都能够实现佩珀特所期望的能够支持儿童创造微世界

的理想。就像 BASIC 语言也曾经在中小学流行过，但是佩珀特还是嫌 BASIC 语言过于简单，做不了什么事情，而且与儿童的直觉思维还有些距离。如用 BASIC 语言在屏幕上画圆，需要有坐标概念，并且只能在屏幕的指定位置画圆；但是用 LOGO 语言，儿童是通过想象自己沿着圆周走的步骤“前移一点，右转一点”来完成画圆的——相较于前者，后者更加直观，没有坐标系概念的低龄儿童也能够胜任，而且还可以画在屏幕上的任何位置。之所以能够“随心所欲”，是因为 LOGO 系统“深埋”了很多先行知识，表面上看到的是接近自然语言的表述，而在实现层，其程序语言的解释系统做了很多工作，而这些复杂性都对用户屏蔽了。比如，为了支持数学学习和物理学习，佩珀特分别定义和实现了几何龟和牛顿龟，从而造就了“简化的世界”。

在 *MindStorms* 一书中，LOGO 系统只是证实佩珀特关于计算机在教育中可以发挥更重要的作用此设想的一个证据。比儿童编程概念更上位的概念是“微世界”的概念，这是对研究和开发计算机教育应用最有启发的核心概念，也是大有可为的发力点。目前，有很多智能 APP 都具备成为可探索微世界的潜质。

*MindStorms* 一书写于 1980 年，当时个人电脑刚开始进入家庭，佩珀特预见计算机进入每个人的日常生活，因此他畅谈了在人人都有计算机的时代，计算机可以为儿童的学习带来什么变化。如今已过去了 40 年，计算机确已普及，有些经济发达的地方甚至实现了人手一台，但是计算机在教育中的应用很多都没有达到当年 LOGO 系统的高度。与 20 世纪 80 年代类似，“目前大多数以教育技术之名的实验还处于将陈旧的教学方法与新的技术线性混合的状态 (the Linear Mix)”，要“将基本的教育原则与新方法更有机地交汇 (more Organic Interaction) 并转变为现实”，还需要更多从事计算机教育应用的人士保持清醒的头脑，不困难为而不为。

---

\*基金项目：本文受教育部—中国移动科研基金 2018 年度项目“中小学编程教育与人工智能工程素养研究”（项目编号：MCM20180611）资助。

作者简介：汪琼，教授，博士，研究方向为教育技术学基础理论与方法、教学设计，邮箱为 wangqiong@pku.edu.cn。

收稿日期：2020 年 1 月 3 日

编辑：小西