

# 基于计算思维培养的Mind+编程教学案例研究

黄 红

上海市金汇实验学校, 上海

收稿日期: 2023年10月8日; 录用日期: 2023年11月18日; 发布日期: 2023年11月24日

## 摘 要

在教育改革的背景下, 信息科技学科的育人价值越来越注重学生核心素养的培养, 其中就包括计算思维。而在小学的信息科技教学中, 运用图形化编程课程开展计算思维的培养和研究成为当前主要的教学趋势, Mind+作为我国自主开发的图形化编程软件, 集成各种主流主控板及上百种开源硬件, 支持人工智能(AI)与物联网(IoT)功能, 既可以拖动图形化积木编程, 还可以使用Python/C/C++等高级编程语言, 非常适合学校日常教学。本文就是运用Mind+软件自带的音乐模块, 引导学生进行音乐创编。运用任务驱动的方法, 帮助学生思考问题、分析问题、分解问题、解决问题。在整个处理问题的过程中, 实现学生计算思维的发展。

## 关键词

计算思维, Mind+编程教学, 小学信息科技

# A Case Study on Mind+ Programming Teaching Based on Computational Thinking Cultivation

Hong Huang

Shanghai Jinhui Experimental School, Shanghai

Received: Oct. 8<sup>th</sup>, 2023; accepted: Nov. 18<sup>th</sup>, 2023; published: Nov. 24<sup>th</sup>, 2023

## Abstract

In the context of educational reform, the educational value of information technology disciplines is increasingly emphasizing the cultivation of students' core competencies, including computational thinking. In primary school information technology teaching, the use of graphical programming courses to cultivate and research computational thinking has become the main teach-

ing trend. As a self-developed graphical programming software in China, Mind+ integrates various mainstream control boards and hundreds of open-source hardware, supports artificial intelligence (AI) and the Internet of Things (IoT) functions, can drag graphical building block programming, and can also use advanced programming languages such as Python/C/C++. It is very suitable for daily teaching in schools. This article uses the built-in music module of Mind+ software to guide students in music creation. This article uses task-driven methods to help students think, analyze, decompose, and solve problems and develops students' computational thinking throughout the entire problem-solving process.

## Keywords

Computational Thinking, Mind+ Programming Teaching, Primary School Information Technology

Copyright © 2023 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 计算思维概念综述

计算思维起源于 20 世纪 80 年代, 2006 年 3 月, 美国卡内基·梅隆大学计算机科学系主任周以真教授在美国计算机权威期刊《*Communications of the ACM*》杂志上给出, 并定义的计算思维。周教授认为: 计算思维是把一个看来困难的问题重新阐释成一个我们知道问题怎样解决的方法; 是一种采用抽象和分解来控制庞杂的任务或进行巨大复杂系统设计的方法; 是一种选择合适的方式去陈述一个问题, 或对一个问题的相关方面建模使其易于处理的思维方法[1]。过去约 15 年来, 计算思维再度受到学界关注, 该研究领域开始走向成熟。近年来, 大量关于计算思维的研究发表。《义务教育信息科技课程标准(2022 年版)》(以下简称新课标)明确信息科技课程建设需要反映数字时代正确育人方向, 信息科技课程要以培养学生的核心素养为目标。新课标中对计算思维核心素养做了新的定义。计算思维是指个体运用计算机科学领域的思想方法, 在问题解决过程中涉及的抽象、分解、建模、算法设计等思维活动。具备计算思维的学生, 能对问题进行抽象、分解、建模, 并通过设计算法形成解决方案; 能尝试模拟、仿真、验证解决问题的过程, 反思、优化解决问题的方案, 并将其迁移运用于解决其他问题[2]。计算思维的培养在 K-12 教育体系中占有重要的地位。各个国家及地区从课程项目、培养标准、研究机构、实施工具和师资培训等多个方面开展计算思维培养, 并将计算思维应用在跨学科学习方面。但不同国家地区对计算思维培养的理解、开展途径、评价角度和方式也是不同的[3]。朱珂等人结合 STEM 教育的教学过程, 将计算思维能力分为分解能力、概括能力、算法思维、评估能力以及抽象化五个要素[4]。Shute 等人开发了可用于 CT 能力评估的模型, 涉及抽象(数据收集与分析、模式识别、模型化)、分解、算法(算法设计、类比、效率、自动化)、调试、概括及迭代[5]。

我国新课标对计算思维培养的指导和规划契合了国家的发展进程。信息科技的快速发展与广泛应用创生出一个全新的数字化环境。这个数字化环境在改变人们行为方式的同时, 也改变着人们的思维方式, 赋予了信息科技课程新的内涵。目前, 国外计算思维研究较为成熟, 而我国关于计算思维的研究与国外相比还处于初级阶段。专家建议我国在未来开展计算思维培养上可将重点放在基础教育阶段。例如在基础教育教学中, 教师要借助编程语言的教学设计探索计算思维培养的有效方式和途径; 教师要尝试利用信息工具构建有效的学习环境; 教育部门和相关单位要健全计算思维培养的有评价体系等[6]。

## 2. 编程教学发展现状

国外关于小学生编程教学的研究可追溯至 1968 年，西蒙·派珀特(Seymour Papert)教授与他团队成员合作研发出世界首款儿童编程语言——Logo。美国发布的《青少年计算机科学框架》有力推动了中小学阶段编程教育的发展。2016 年澳大利亚正式将编程列入国家必修课程，作为教学的重要内容。在英国，将编程融入《计算》和《设计与技术》课程，分三阶段开展教学：第一阶段学生直接接触编程语言；第二阶段学生将发展更复杂的编程技能；第三阶段学生将转向更抽象的层次。日本在 2016 年提倡在中小学推行 STEAM 教学，并将编程学习列为重要学习内容，出台了小学编程教学的实施步骤，从小学学段开始重视编程教育。欧盟在 2014 年正式推行“编程周”，旨在通过编程提升学生的创造力、合作能力和解决问题的能力，后陆续的把编程教育课程纳入了中小学的课大纲中，各国便开发出独具特色的编程课程[7]。

由此可见，各国从国家层面推广编程教学的决心和行动，将编程全面纳入学校教育成为一种趋势。计算思维自诞生以来始终与编程教育密切联系在一起，编程教育也被认为是培养计算思维的主要方式。计算思维培养导向的编程教学仍处在探索阶段，存在培养内容不够全面、编程工具使用效果参差不齐等问题。因此，本研究将分析编程过程与计算思维发展的关系。编程过程涉及到问题的分析、算法的设计、代码的编写和调试等多个环节。在这个过程中，学生需要不断思考和调整自己的解决方案，从而培养和发展自己的计算思维能力。通过对编程过程与计算思维发展的关系进行深入分析，我们可以更好地理解编程对计算思维的培养作用。

## 3. Mind+编程软件介绍

由美国麻省理工学院研究开发的 Scratch 软件是一款面向青少年的图形化的编程工具。它也是图形化编程软件的鼻祖，后来随着使用的人越来越多，我国的很多软件公司也相应的推出了图形化编程软件，他们的操作界面及指令积木基本一致，Mind+软件就是其中之一。图形化编程不同于传统的编程工具，它将枯燥的代码集成在彩色的模块之中，儿童只需通过阅读和拖拽不同的模块进行有意义的组合，像搭积木一样，就可以实现各种功能。

### 3.1. 模块化的界面简洁，操作方便

Mind+编程软件的界面分为八大部分，标题栏和功能菜单与常见计算机软件的功能类似，指令区、脚本区、舞台区、角色区、工具按钮、控制按钮具有软件本身的特色。使用者可以在角色区根据自己的需要选取任意的角色，角色被导入后会直接出现在舞台区。学生只需用鼠标将指令区的指令块拖到脚本区即可实现编程，删除时只需将脚本区的指令块拖回左侧的脚本区即可。Mind+软件用不同的颜色标注了 10 种不同的命令组：动作、外观、声音、事件、控制、侦测、运算符、变量、函数、扩展(详见图 1)。学生只要能读懂每个命令块的中文含义，就基本能掌握每个命令块的功能作用，比如动作命令组中的向右旋转 15 度，顾名思义就是让角色向右转 15 度。

除了一些众所周知的英文缩写，如 IP、CPU、FDA，所有的英文缩写在文中第一次出现时都应该给出其全称。文章标题中尽量避免使用生僻的英文缩写。

### 3.2. 图形化编程的作品形式多样，编程简单易上手

图形化编程软件的设计迎合了儿童的心理特点，其素材库中的素材丰富，几乎涵盖了儿童认知的各个领域。如果儿童不满足于已有的素材也没有关系，还可以通过软件自带的绘图功能自己创作或者从外部导入。软件包括了顺序、分支、循环条件语句、链表数组和变量等常见的编程概念，但学生不需要专门去学习这些枯燥编程概念，而是可以在教师创设的情境中为实现某种功能而不自觉地学习和应用了编

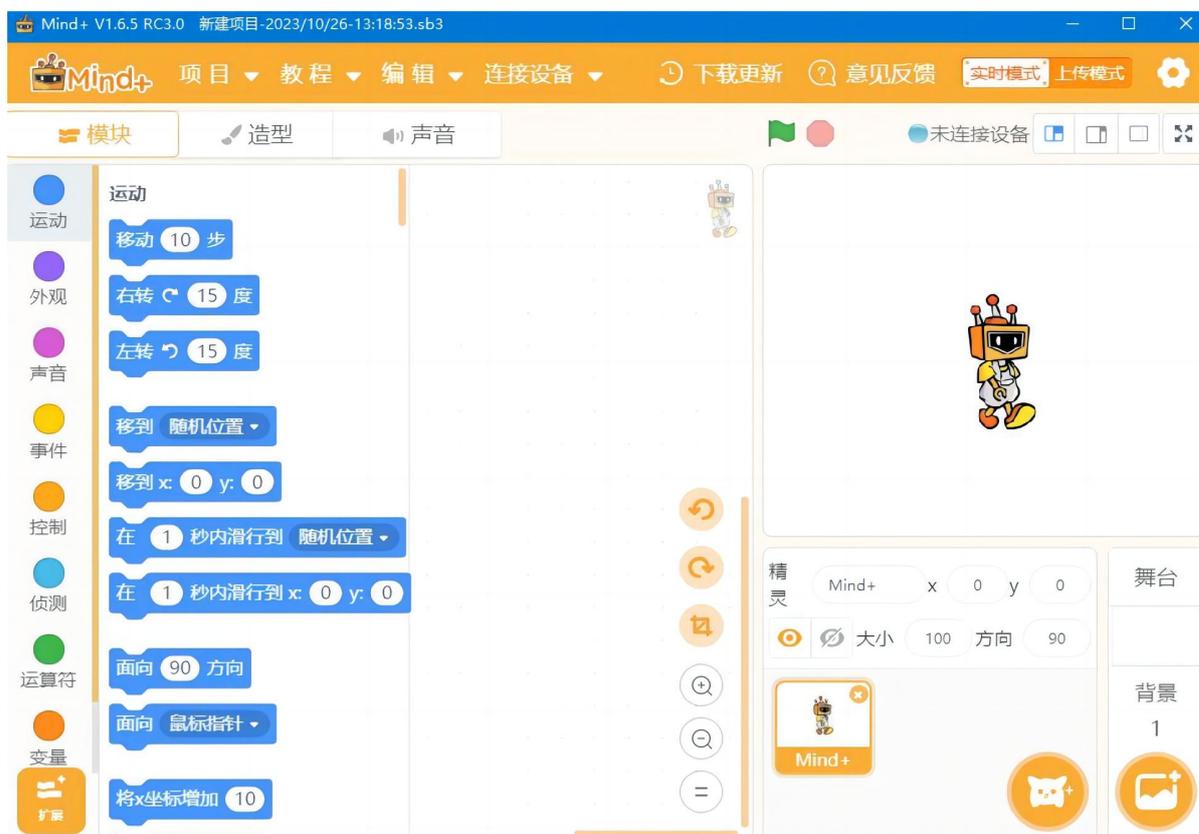


Figure 1. Software interface  
图 1. 软件界面

程思维。学生可以通过图形化编程创作富有个性化的动画、游戏、故事、音乐等小作品，获得极大的成就感。图形化编程与计算思维的培养之间存在着隐形的桥梁。使用图形化创意编程软件不是培养学生未来成为高级程序员，而是帮助他们形成计算思维，希望他们在创意编程中尽情表达想法。就目前来看，图形化编程软件无疑成为中小学编程教育的最佳软件之一，为学生学习编程提供了方便。

## 4. 提升计算思维的 Mind+编程教学实践

### 4.1. 小学编程教学问题分析

信息科技学科教学中的编程教学被认为是提升学生计算思维的有效路径，但在目前的编程课程的教学过程中，教学方法还是以“教师讲授 - 学生模仿范例”为主，学生的自主思维不能得到有效的激发，学习层次表面化，效果不理想，严重阻碍了学生综合素质的提高。学生学习后不能理解知识的本质，不能很好的利用所学来解决生活中的实际问题，这显然达不到国家全面深化课程改革的要求。因此，把培养学生计算思维为目标的教学必须把重点聚焦在学生的思维品质上，通过合理的问题引导，提高学生的思维参与度，促进深层学习，掌握学习和思维的正确方法。

### 4.2. 提升学生计算思维的必要性

随着科技的飞速发展，计算思维已经成为当今社会的一种重要能力。计算思维是一种解决问题的方法，它强调通过抽象、分解、模式识别等方法来理解、分析和解决复杂问题[8]。在这个时代，培养学生的计算思维已经成为育人的需要。

1) 计算思维有助于提高学生的创新能力。在信息时代,我们面临着大量的新问题和挑战,这些问题往往需要创新的解决方案。具备计算思维能力的学生能够更好地分析问题、提出创新性的解决方案,并付诸实践。这对于培养具有创新精神和实践能力的人才具有重要意义。

2) 计算思维有助于培养学生的团队协作能力。在解决复杂问题的过程中,往往需要多学科、多领域的知识和技能。具备计算思维能力的学生能够更好地与他人沟通、协作,共同解决问题。这对于培养具有团队协作精神和跨学科沟通能力的人才具有重要意义。

3) 计算思维有助于提高学生的问题解决能力。在现实生活中,我们经常面临各种问题,如环境污染、资源紧张、人口老龄化等。具备计算思维能力的学生能够更好地分析问题、找出问题的根本原因,并提出有效的解决方案。这对于培养具有解决问题能力和责任感的人才具有重要意义。

4) 计算思维有助于培养学生的终身学习能力。在信息爆炸的时代,我们需要不断地学习新知识、新技能,以适应社会的发展。具备计算思维能力的学生能够更好地利用信息技术进行自主学习、探究式学习,形成持续学习的能力和习惯。这对于培养具有终身学习能力和适应社会发展需求的人才具有重要意义。

总之,提升学生计算思维是时代育人的需要。我们应该重视计算思维教育,将其融入到课程设置、教学方法和评价体系等方面,为培养具有创新精神、团队协作能力、问题解决能力和终身学习能力的人才奠定基础。

### 4.3. 基于培养计算思维的教学设计

本课是以《闵行区小学信息科技学科教学基本要求》主题二使用图形界面软件(Scratch)的内容为基础,教学软件改为国产的图像化编程软件 Mind+。它是一款拥有自主知识产权的国产青少年编程软件,既可以拖动图形化积木编程,还可以使用 Python/C/C++等高级编程语言,让大家轻松体验创造的乐趣。整节课通过软件自带的扩展功能“音乐模块”的调用,让学生体验自主创编音乐的乐趣,在脚本语言的使用上,整个程序语句是一种顺序结构。通过程序的编写,学生能够知道程序是从上到下一步一步执行,每一步都有它特定的意义和实现的功能。详细的课堂流程包括:1) 布置挑战任务;2) 自然语言到程序语言过度;3) 梳理程序设计步骤;4) 音乐主程序设计;5) 调试与优化;6) 交流与评价。

图形化编程软件界面布局清晰,其游戏式的界面深受小朋友的喜爱,非常适合小学三年级的学生进行学习。本节课以课堂任务挑战的形式展开,整节课由三个不同难易梯度的任务串联起来,学生在挑战任务的同时体验编程的乐趣,从完成任务中获得成就感和满足感。学生从基于自然语言的挑战任务中提炼出关键信息,将关键信息转化为计算机能够识别的程序语言,最后通过调试程序自动化的来帮助 we 解决问题。下面以《我的音乐我做主》活动为例,介绍编程教学培养计算思维的策略与方法。

#### 4.3.1. 精读任务文本, 提炼关键字

本节课是本单元的第3课时,经过前面2节课的铺垫,学生对 Mind+软件的操作界面有了基本了解,知道程序设计三要素:舞台、角色、脚本。能跟随老师完成简单的程序设计,但在自主设计程序上还存在困难。本节课以任务驱动的方式,在教师带领下从分析任务入手,利用表格列出程序设计要素,提问“故事发生的舞台背景是什么?”“故事中有几个角色?”通过问题引导,帮助学生提炼文本中的关键信息,为脚本设计做准备。

#### 4.3.2. 自然语言向程序语言过渡, 提升思维品质

在程序设计过程中,自然语言和脚本语言还是有区别的,对于初学者来说,还不能很好的掌握两者的区别和过渡,在本案例中,我通过引导分析,帮助学生树立自然语言和程序语言之间的转换关系,为学生自主创编和完成拓展任务做准备。

**Table 1.** Comparison between “natural language” and “program language”  
**表 1.** “自然语言”与“程序语言”对照

自然语言	程序语言
1 点击“吉他”，开始播放音乐	
2 播放的音乐是“吉他”音乐	
3 音乐的演奏速度是中速	

#### 4.3.3. 借助流程图，提升计算思维素养

计算思维中的分解问题是将一个复杂的大问题分解成若干个易于解决和实现的小问题。要完成“吉他播放音乐”的全过程，就需要进行分解，逐个解决。让学生去归纳程序设计的步骤，对学生来说是比较困难的，我采用的教学策略是任务驱动式，先呈现程序设计的一般流程，借助流程的引导去思考程序设计中对应的四个问题。问题 1：程序是如何启动的？问题 2：程序中应该设定哪些初始化脚本？问题 3：音乐的主程序是什么？问题 4：程序执行结果与预期效果不一致时，应该怎么办？学生在分析问题和解决问题中，逐步掌握了程序设计的一般思路，提升了思维品质，也为后续的学习打下很好的基础。

### 5. Mind+编程教学的实践价值

本案例的学习者是小学三年级的学生，这一年龄的学生具有初步的自主学习能力，他们的认知发展正处于从具象思维到抽象思维的过渡时期。他们对新事物充满好奇，喜欢挑战，整节课围绕挑战任务“点击‘吉他’，演奏一段小星星音乐”开展教学。学生的学习兴趣浓厚，课堂专注度高，每一位学生都能积极的投入到学习中。

#### 5.1. 任务驱动式的编程活动，提升学生问题解决能力

在课堂教学中，教师将“演奏一段小星星音乐”的编程任务，分解为三个子任务：提炼文本关键字、自然语言向程序语言转换、程序实践验证。每一个子任务对应计算思维模型框架中的四个过程——“抽

象、分解、算法、调试”。针对每一个子问题，学生在教师引导下抽象问题本质，建立问题模型；尝试分解问题，逐个化解；在程序实践阶段，学生能够提炼算法，自主实践。通过本课的实践，为学生的编程学习打开了学习思路，培养学生具象思维到抽象逻辑思维的过渡，能够理解事物之间的内在联系，从而提高学生的问题解决能力。

## 5.2. 任务驱动式的编程活动，提升学生创新能力

任务驱动式的教学过程中，教师设置开放性的任务，更有利于发挥学生的创造性。本案例中，整个任务执行过程中，因为学生的选择不同，呈现的学习结果也不同。学生可以设计他的音乐播放器为“钢琴”、“吉他”，或者任何其他可供选择的乐器，在进行音乐编辑时，学生掌握了用程序语言播放音乐的速度以及音乐节奏等，都会使学生最终呈现的作品大不相同。学生成果展示环节，学生的作品充满了创新点，比如运用循环模块与主程序结合，实现音乐旋律的循环播放；引入不同乐器角色，利用复制的方式复制程序脚本，实现多乐器同台合奏的演出效果。通过这节课，学生的思维活跃度得到很好调动，课堂灵动而充满活力。

## 参考文献

- [1] 360 百科. 计算思维[EB/OL]. <https://baike.so.com/doc/3092926-3260084.html>, 2022-08-10.
- [2] 中华人民共和国教育部. 义务教育信息科技课程标准(2022 年版) [M]. 北京: 北京师范大学出版社, 2022.
- [3] 郁晓华, 肖敏, 王美玲. 计算思维培养进行时: 在 K-12 阶段的实践方法与评价[J]. 远程教育杂志, 2018(2): 18-28.
- [4] 朱珂, 贾鑫欣. STEM 视野下计算思维能力的发展策略研究[J]. 现代教育技术, 2018, 28(12): 115-121.
- [5] 多召军, 刘岩松, 任永功. 编程教育促进儿童计算思维发展的内在机理与教学实践研究[J]. 电化教育研究, 2022(8): 101-108.
- [6] 范文翔, 张一春, 李艺. 国内外计算思维研究与发展综述[J]. 远程教育杂志, 2018, 36(2): 5-19.
- [7] 刘佳慧. 编程教学中提升初中生问题解决能力的实验研究[D]: [硕士学位论文]. 广州: 广东技术师范大学, 2022.
- [8] 王蕴辉. 面向计算思维培养的初中编程游戏化学习过程设计研究[D]: [硕士学位论文]. 长春: 东北师范大学, 2021.