

# 计算机教育中编程技能与批判性思维能力的协同发展研究

刘杰<sup>1</sup> 刘启蒙<sup>1</sup> 李英龙<sup>2</sup>

1. 吉林农业科技学院; 2. 赤峰曾军良实验学校(赤峰第四中学分校)

**摘要:** 在21世纪这个数字化时代,计算机教育已成为教育体系中不可或缺的一部分,其重要性日益凸显。编程技能作为连接现实与数字世界的桥梁,不仅关乎技术革新与产业升级,更是培养未来社会所需人才的关键能力之一。与此同时,批判性思维能力作为个体认知发展、问题解决及创新创造的基石,对于提升学生的综合素质具有不可估量的价值。本文旨在探讨计算机教育中编程技能与批判性思维能力的协同发展路径,分析两者之间的内在联系,并提出促进二者融合的教学策略。

**关键词:** 计算机教育; 编程技能; 批判性思维能力; 协同发展

**【DOI】** 10.12252/j.issn.2096-6288.2025.07.127

## 引言

随着信息技术的迅猛发展,编程技能已成为现代社会不可或缺的一项基本技能。它不仅是计算机科学专业的核心,还广泛应用于各个行业,如金融、医疗、教育等。因此,培养学生的编程技能已成为计算机教育的重要任务。然而,在解决实际问题时,学生还需要具备批判性思维能力,即能够独立思考、分析与解决问题,这种能力对于培养学生的创新精神至关重要。在此背景下,计算机教育开始关注编程技能与批判性思维能力的协同发展。

## 一、编程技能与批判性思维能力的内涵解析

### (一) 编程技能的定义与重要性

编程技能是指通过编写代码来创建、测试、调试和维护计算机程序的能力。它不仅要求学生掌握编程语言、算法设计等基础知识,还强调逻辑思维、问题解决和创新能力。在信息时代,编程技能已成为许多行业的基本工具,对于推动科技进步、促进经济发展具有重要意义。

### (二) 批判性思维能力的构成与价值

批判性思维能力包括分析、评估、推理、判断及创新等多个方面,是个体在面对复杂问题时能够独立思考、理性判断并做出合理决策的能力。它不仅是学术研究的基石,也是个人成长和社会进步的重要驱动力。在教育领域,培养学生的批判性思维能力对于提升其综合素质、增强社会适应能力具有深远影响。

## 二、编程技能与批判性思维能力的内在联系

### (一) 编程实践中的批判性思维应用

在编程实践中,批判性思维发挥着至关重要的作用。首先,编程本质上是一个问题解决的过程,而批判性思

维正是这一过程中的核心驱动力。面对复杂的编程任务,程序员需要首先识别问题,这要求他们具备对问题本质的深刻洞察能力。随后,他们需要对问题进行详细的分析,理解其各个组成部分及其相互关系,这涉及到批判性思维中的分析、评估和推理能力。在确定了问题的基本框架后,程序员需要设计并实现解决方案,这一过程中,批判性思维帮助他们审慎评估各种可能的方案,选择最优或最适合当前情境的方案。

### (二) 批判性思维促进编程技能的提升

批判性思维不仅在编程实践中发挥着重要作用,还通过不断的实践和应用促进着编程技能的提升。首先,批判性思维要求程序员对问题进行深入的思考和分析,这种深入的思考和分析过程有助于他们更全面地理解问题,从而设计出更加合理和高效的解决方案,这种能力的提升直接反映在编程技能上,使得程序员能够编写出更加健壮、可维护和可扩展的代码。

其次,批判性思维还促进了程序员对新知识和新技术的接受和应用。在快速迭代的软件开发领域,新技术层出不穷。批判性思维要求程序员在面对这些新技术时保持审慎态度,通过对比分析、试错验证等方式来评估其价值和适用性,这种对新技术的批判性接受和应用能力使得程序员能够紧跟技术发展的步伐,不断提升自己的编程技能。最后,批判性思维还促进了程序员之间的交流与合作。在团队协作中,批判性思维帮助程序员更好地理解 and 评估他人的观点和建议,通过积极参与讨论、提出建设性意见等方式来促进团队内部的知识共享和思维碰撞,这种交流与合作不仅有助于提升团队的整体编程水平,还能够激发团队成员的创新思维,推动项目向更高质量、更高效的方向发展。

### 三、计算机教育中编程技能与批判性思维能力的协同发展策略

#### (一) 项目式学习法

在计算机教育中,项目式学习法(Project-Based Learning, PBL)作为一种高效的教学策略,不仅能够显著提升学生的编程技能,还能有效促进批判性思维能力的协同发展,这种方法通过将学生置于真实或高度模拟的项目情境中,让他们亲身体验从需求分析、方案设计、编码实现到测试调试的全过程,从而在实践中深化对编程知识的理解和应用,同时锻炼出一系列重要的非技术技能,包括批判性思维、团队合作和问题解决能力等。

##### 1. 融合编程技能与批判性思维能力

在项目式学习法中,编程技能与批判性思维能力是相互依存、相互促进的。编程不仅仅是编写代码的过程,它更是一种逻辑思考和问题解决的艺术。学生在面对项目任务时,首先需要运用批判性思维对问题进行深入分析,明确问题的本质、边界和限制条件,这一过程锻炼了他们的分析能力、判断能力和创新能力。随后,在设计解决方案和编码实现阶段,学生需要将批判性思维转化为具体的编程逻辑和算法设计,这一转化过程不仅考验了他们的编程技能,也进一步加深了他们对批判性思维的理解和应用。

##### 2. 提升问题解决与反思能力

在项目式学习法中,问题解决能力是学生需要掌握的核心能力之一。学生在面对项目中的复杂问题时,需要运用批判性思维进行问题拆解、假设提出、实验验证和结果评估等一系列活动,以找到有效的解决方案。在解决问题的过程中,学生还会不断遇到新的挑战 and 困难,这要求他们具备持续学习和反思的能力。学生通过反思,总结经验教训,优化解决方案,提高自己的问题解决能力和批判性思维水平。

##### 3. 测试调试与强化批判性评估

测试调试是编程过程中不可或缺的一环,也是锻炼学生批判性思维能力的关键环节。在测试调试阶段,学生需要仔细分析程序运行的结果与预期之间的差异,找出潜在的问题和错误,这一过程要求学生具备敏锐的观察力和严谨的逻辑推理能力,能够运用批判性思维对问题进行深入分析和评估。学生通过不断的测试调试和修正改进,不仅能够提高编程技能,还能够增强自己的批判性评估能力和质量意识。

#### (二) 反思性教学

在计算机教育中,编程技能与批判性思维能力的协同发展是提升学生综合素质、培养其成为未来科技领域

创新者的重要途径。而反思性教学作为一种有效的教学策略,对于促进这两种能力的融合具有不可替代的作用。反思性教学强调学生在学习过程中不仅要掌握知识和技能,更要学会对自己的学习过程进行审视和思考。在编程教学中,这意味着学生不仅要编写代码、解决问题,还要思考自己的编程思路是否合理、代码实现是否高效、问题解决策略是否最优等。学生通过反思,更深入地理解编程的本质,发现自己在编程过程中存在的问题和不足,进而调整学习策略,提高学习效率和质量。

##### 1. 明确反思目标

教师在编程教学开始前,应与学生共同明确反思的目标和重点。例如,教师设定在每次编程任务完成后,学生需要反思自己的编程思路是否清晰、代码是否简洁易懂、问题解决策略是否有效等。明确的目标有助于引导学生有针对性地进行反思,提高反思的效果。

##### 2. 创设反思情境

为了激发学生的反思意识,教师应创设一些具体的反思情境。例如,在编程过程中设置一些“陷阱”或“难点”,让学生在遇到问题时不得不停下来思考自己的解决方案是否正确;或者在编程任务完成后,组织学生进行小组讨论,分享各自的编程思路和经验,让学生在交流中互相启发,发现自己的不足之处。

##### 3. 引导反思过程

在反思过程中,教师需要发挥引导作用,帮助学生掌握反思的方法和技巧。例如,教师引导学生从以下几个方面进行反思:一是审视自己的编程思路是否清晰、合理;二是分析代码实现过程中遇到的问题及解决方法;三是评估问题解决策略的有效性和可优化空间;四是总结本次编程任务的收获和教训。同时,教师还可以通过提问、引导讨论等方式,激发学生的思考兴趣,促进他们的深入思考。

##### 4. 强化反思结果的应用

反思不仅仅是为了发现问题和不足,更重要的是要将反思结果应用到后续的学习中。因此,在反思结束后,教师应鼓励学生根据反思结果调整自己的学习策略和方法。例如,如果学生在反思中发现自己的编程思路不够清晰,引导他们学习更高效的编程思维方法;如果学生在代码实现过程中遇到了问题,指导他们学习更先进的编程技术和工具。

#### (三) 跨学科融合教学

在计算机教育中,编程技能与批判性思维能力的协同发展是一项重要任务,而跨学科融合教学则为这一目标的实现提供了强有力的支持。教师应将计算机教育与

数学、物理、社会科学等其他学科相融合，设计出富有挑战性和启发性的学习项目，让学生在解决复杂问题的过程中，不仅提升编程技能，还能深化批判性思维能力。跨学科融合教学打破了传统学科之间的界限，使各学科的知识与方法相互渗透、相互融合，这种教学方式有助于培养学生的综合素质和创新能力，使他们在面对复杂多变的社会问题时，能够运用多学科知识和技能进行综合分析和解决。

### 1. 设计跨学科学习项目

跨学科学习项目如同一座桥梁，巧妙地连接了知识的岛屿，让学生在广阔的知识海洋中自由航行。在这样的框架下，教师化身为引路人，根据学生的独特兴趣与实际需求，精心策划了一系列融合多学科的编程项目，旨在激发学生的探索欲与创造力。以物理学与编程的交汇点为例，我们设计了一系列引人入胜的项目，让学生们在虚拟的数字世界中重现物理定律的辉煌。学生不再局限于课本上的公式和图表，而是亲手编写代码，模拟复杂的物理实验。从控制实验参数到观察微妙的现象变化，每一步都凝聚着他们的智慧与汗水。在这个过程中，编程不仅成为了他们探索物理奥秘的工具，更让他们在实践中深刻理解了物理原理的精髓。这样的跨学科项目，不仅锻炼了学生的编程技能，提升了他们的逻辑思维与问题解决能力，更重要的是，它激发了学生对科学的无限热爱与追求。学生们在编程与物理的碰撞中，发现了知识的力量，也为他们未来的学习与职业生涯奠定了坚实的基础。

### 2. 引入实际问题情境

教师将实际问题引入编程教学中，能够激发学生的学习兴趣。教师应根据其他学科的教学内容，设计具有实际应用价值的编程题目。例如，在社会学领域，学生利用编程技能来分析社会调查数据，通过数据可视化等方式揭示社会现象背后的规律，这样的实践过程不仅锻炼了学生的编程技能，还培养了他们的数据分析和批判性思维能力。

### 3. 强化团队协作与交流

跨学科融合教学往往涉及多个学科知识和技能，因此需要学生之间的团队协作与交流。在编程项目中，学生组成跨学科小组，共同讨论问题、设计方案、编写代码并分析结果，这样的团队协作过程不仅有助于提升学生的编程技能，还能培养他们的沟通协调能力和批判性思维能力。学生通过交流不同的观点，学会从不同角度审视问题，从而找到更加全面的解决方案。

### 4. 开展综合评价与反馈

跨学科融合教学需要开展全面的评价与反馈机制。教师应根据学生的学习过程和成果，从编程技能、批判性思维能力、团队协作能力等多个方面进行评价。同时，教师还应及时给予学生反馈和指导，帮助他们发现自己的不足之处并寻求改进方法，这样的评价与反馈机制有助于激发学生的学习动力，促进他们的全面发展。

### 结语

综上所述，计算机教育中编程技能与批判性思维能力的协同发展是时代赋予教育的新使命。本文深入分析两者之间的内在联系，探索并实施有效的教学策略，希望更好地培养学生的综合素质和创新能力，为社会输送更多具备高度竞争力的复合型人才。未来，随着教育技术的不断进步和教育理念的持续更新，我们有理由相信，编程技能与批判性思维能力的协同发展将在计算机教育领域绽放出更加璀璨的光芒。

### 参考文献

- [1] 尹鸿峰, 周子程, 毛国兴. 计算机编程教育中的计算思维与教学策略研究 [J]. 电脑知识与技术, 2024, 20(8): 172-174.
  - [2] 王慧, 王璜. 计算机编程教育在中小学课程中的融合与创新 [J]. 电脑校园, 2024(4): 223-225.
  - [3] 马宁. 计算机基础教育中编程思维培养的实践与思考 [J]. 中文信息, 2024(12): 167-168.
  - [4] 刘绘绘. CDIO 视角下的高中编程教育教学模式及应用研究 [D]. 贵州: 贵州师范大学, 2023.
  - [5] 陈兰芳. 高校信息教育中的计算机编程教学方法研究与实践 [J]. 信息技术时代, 2023(13): 158-160.
  - [6] 龚珊珊. 面向学生高阶思维能力培养的 CDIO 教学模式实践应用研究——以初中 Python 编程教育为例 [D]. 渤海大学, 2023.
  - [7] 张欢, 袁军. 基于创客教育的中小学编程教学实践探讨——以贵州省 D 市 S 学校为例 [J]. 中国教育技术装备, 2023(6): 82-85.
  - [8] 孙利平. “双减”背景下初中计算机编程翻转课堂教学策略——以“While 循环”一课为例 [J]. 中小学信息技术教育, 2022(10): 57-60.
- 作者简介: 刘杰, (1990.02), 女, 汉族, 吉林省松原市, 初级、本科、计算机科学与技术。
- 基金项目: 本文系吉林农业科技学院教育教学改革课题“农业机械类专业的先进成图技术课程建设研究”(项目编号: 2024XJYB03)。