

面向三维目标的国外中小学 计算思维培养与评价研究

王旭卿

(上海师范大学 信息与机电工程学院 计算机系, 上海 200234)

[摘要] 当前以 Scratch 为代表的可视化图块式编程工具受到了国内中小学信息技术教师和学生的普遍欢迎,但许多教师却不知如何表征和评价学生利用 Scratch 开展的互动媒体编程活动及其学习结果。美国麻省理工学院终身幼儿园研究小组在多年研究互动媒体设计者的编程活动基础上提出了计算思维三维框架(计算概念、计算实践和计算观念),基于此开发了面向教师工作坊的 Scratch 创意计算课程,并设计了三种计算思维评价方法。这些国外中小学计算思维培养与评价的理论与实践为进一步的本土化探索与研究奠定了基础。

[关键词] 计算思维三维框架; 三维目标; Scratch 创意计算课程; 计算思维评价方法

[中图分类号] G434

[文献标志码] A

[作者简介] 王旭卿(1969—),女,浙江海宁人。副教授,博士,主要从事中小学信息技术教育、职前教师教育等研究。

E-mail:cnwxq@shnu.edu.cn。

一、引言

近年来,以 Scratch 为代表的可视化图块式编程工具受到了国内中小学信息技术教师和中小学生的欢迎。Scratch 编程语言和环境借鉴并发展了 Logo 编程语言面向编程初学者的设计初衷,它具有低门槛(Low Floor)、高界限(High Ceiling)和阔空间(Wide Walls)的特点,也就是说 Scratch 容易入门,学生有机会制作越来越复杂的项目,具有不同兴趣和学习风格的学生都能参与进来,完成类型各异的项目^[1]。一言以蔽之,以 Scratch 为代表的可视化图块式编程工具及其在线社区正在开创青少年学习创意计算(Creative Computing)的新范型。

如何表征和评价学生利用 Scratch 开展的互动媒体编程活动及其学习结果呢?周以真教授在 2006 年提出的计算思维是大家普遍认可的用来考量学生掌握计算概念、发展计算实践和能力的重要工具。目前许多专家和组织都从解析认知结构或思维过程等角度提出了计算思维的各种定义,比如,Cuny、Snyder 和 Wing 认为计算思维“是一个明确问题和制定解决方案的思维过程,由此解决方案就可以表示为能够被信

息处理代理有效执行的形式”^[2];又如 2011 年国际教育技术协会(ISTE)联合计算机科学教师协会(CSTA)共同给出了计算思维的操作性定义,他们认为计算思维是一个问题解决的过程,该过程包括制定问题、分析数据、抽象、设计算法、选择最优方案、推广等六大要素。支持问题解决需要一组能力和态度,包括自信、坚持、宽容、解决开放问题的能力和与他人交流合作的能力^[3]。这些定义强调了计算思维作为解决问题能力的特征,总体来说都比较抽象,操作性不佳,很难直接用来表征和评价学生的编程实践活动以及发生在互动媒体创建过程中的学习结果。

二、计算思维三维框架

2012 年,美国麻省理工学院媒体实验室(MIT)终身幼儿园研究小组(Lifelong Kindergarten Group)在多年研究 Scratch 在线社区、Scratch 教师工作坊以及互动媒体设计者的编程活动基础上开发了一个计算思维三维框架,它包括三个维度:计算概念(Computational Concept,指设计者在编程时所使用的概念)、计算实践(Computational Practices,指设计者在编程中所发展的实践)和计算观念(Computational

Perspectives,指设计者形成的有关他们身边世界和他们自己的观念)^[4]。这个理论框架比较通俗易懂,与我们所熟知的信息技术课程三维目标能够近似呼应,即计算概念对应于知识与技能,计算实践对应于过程与方法,计算观念对应于情感态度与价值观。下面结合Scratch设计者的编程活动阐述计算思维三维框架的内涵^[5]。

1. 计算概念

计算概念是学生在编程工具(如Scratch)创建互动媒体时直接频繁接触的、并在熟练运用中不断加深理解的一组概念,它们在可视化图块式编程工具中通常映射为一组指令块(或称代码块)。在各种Scratch编程制品(Programming Artifacts)中最常用的计算概念有七个:顺序(Sequence)、循环(Loops)、并行(Parallelism)、事件(Events)、条件(Conditionals)、运算符(Operators)和数据(Data)。这些计算概念可以迁移至其他编程情境或者非编程情境。

表1 计算概念列表和简要描述

概念	描述
顺序	识别完成一个任务的一系列步骤
循环	重复运行相同的顺序多次
并行	使多件事情同时发生
事件	一个事情会促使另外的事情发生
条件	根据条件作出决策
运算符	支持数学和逻辑表达式
数据	存储、检索和更新数据

具体而言,顺序概念有助于学生将一个特定活动或任务表达为可以被计算机执行的一系列单个步骤或指令(即编程指令序列)。循环概念使学生懂得重复执行相同代码序列的机制。事件概念可以让学生设计多种触发动作的不同情境,如当绿旗被点击,那么……;当空格键被按下,那么……;当对象被点击,那么……,并体会互动媒体的本质。并行概念可以让学生设计一个对象内部的同时执行的多个脚本和不同对象间同时执行的多个脚本。选择条件概念有助于学生理解程序具有根据某个条件作出判断的能力。运算符概念有助于学生实际使用算术运算符、关系运算符和逻辑运算符,组成各种表达式,并理解计算机执行数值运算、字符串运算和逻辑运算的机制。数据概念可以让学生通过使用两种数据的容器(变量和链表)来理解如何存储、检索和更新数据,变量可以保存一个数值或字符串,而链表则可以保存一系列的数值或字符串,它相当于一维数组。实现游戏中的计分功能是学生探究变量含义和变量应用的最常见的动因,它

使得学生学习计算概念变得更有意义。以上七种计算概念的简要描述见表1^[6]。

2. 计算实践

计算实践关注的是学生利用编程工具创建互动媒体的学习过程和问题解决策略,它使我们从关注学生学会什么,转向去关注学生如何学习。通过对Scratch设计者的访谈和观察,MIT研究人员发现了学生在创建互动媒体时所投入的四组实践策略:递增和重复(Being Incremental and Iterative)、测试和调试(Testing and Debugging)、再利用和再创作(Reusing and Remixing)以及抽象和模块化(Abstracting and Modularizing)。创建互动媒体是发展这些实践策略的强有力情境性任务,它们有助于学生形成信息化情境下解决问题的特有方法与策略,并将创建互动媒体过程中形成的问题解决能力迁移至其他各种设计活动之中。

具体而言,递增和重复,让学生懂得设计一个作品不是一个直接的、线性过程:先明确作品的概念,接着开发设计计划,再用代码就能实现设计。实际上,设计是一个自适应的摆弄(Tinkering)过程:计划会在一步步不断试验中得到修正。一些Scratch设计者提出了从想象到创建互动媒体的迭代式循环,如提出想法,先开发一小部分试验,基于经验和新想法继续开发更多的部分。测试和调试实践有助于学生掌握预测、修正问题的方法,Scratch设计者可以利用Scratch编程环境的即时反馈(如正在执行的代码块周围的高亮条)和可视化数据(如变量值),并通过试误、来自其他活动的迁移或者得到更有能力的其他人的支持来开展测试和调试实践,比如明确问题的来源、通读脚本、测试脚本、尝试重写脚本、找到成功有效的脚本示例、询问其他人如何解决问题等。再利用和再创作实践,是借助网络技术,将自己创建的互动媒体建立在他人成果之上的编程实践。利用Scratch在线社区,学生可以通过后备素材包(Backpack)再利用他人已创建的多种素材(角色、脚本、声音、图片),也可以找到能够加以利用的想法和代码,进行再创作(如修改、增添代码或角色、背景等),这样学生有可能创建比他们独立完成时更复杂的作品。再利用和再创作也有助于锻炼学生的代码阅读能力。抽象和模块化,是所有设计和问题解决的重要实践。人们常常通过抽象建模的方式把现实问题转换为编程问题,即从现实问题中提取关键的数据并考虑如何处理(或操作)数据。针对比较复杂的问题,人们往往通过组合更小的部件(或成为模块)来搭建一个整体。在Scratch中,设计者往往

在多个水平上使用抽象和模块化,从理解问题并形成想法开始,再把想法转换成一个个角色、脚本和自定义模块。例如,从口头叙述的故事中提取多个角色或舞台,并通过编写多个脚本或自定义模块赋予角色或舞台不同的行为表现。以上四组计算实践的简要描述见表 2^[7]。

表 2 计算实践列表和简要描述

实 践	描 述
递增和重复	先开发一点点,然后不断试验,再开发更多的部分
测试和调试	确保可以运行,并发现和修正错误
再利用和再创作	在别人或自己已完成的作品基础上创建新作品
抽象和模块化	通过把更小的部件集合在一起创建更大的作品

3. 计算观念

计算观念是青少年设计者在使用 Scratch 创建互动媒体过程中不断形成的对自己与他人关系以及他们周围世界的理解,这是除概念、实践之外的一种有关人格塑造、思维习惯养成的学习结果,对青少年的社会认知和价值观形成有重要影响。它可以细分为三种观念:表达(Expressing)、联系(Connecting)和质疑(Questioning)。

具体而言,表达观念是指青少年设计者不满足于自己作为消费者与现有的互动媒体打交道,而是乐于使用 Scratch 等可视化编程工具创造性地创建互动媒体并表达自己的想法,“我可以创建”和“我可以通过新媒体表达想法”是他们作为技术生产者的自信表现。联系观念表现在青少年设计者善于借助面对面或者在线社区等不同交流方式与他人交往,以促进自己的创造性实践,他们坚信“当我与他人交往时,我能做与众不同的事”。通过访谈青少年设计者,MIT 研究人员发现了青少年设计者常用两种与他人交往的方式:与他人一起创建和为他人创建。与他人一起创建,是指青少年设计者善于借助各种外在支持,如使用在线论坛寻求帮助、研究和再创作他人的代码、与他人一起组建“设计工作室”合作创建作品,从而创建比他们独立完成时更好的作品。为他人创建,是指青少年设计者通过创建娱乐性作品、调查问卷作品、可被他人利用的素材资源或者教程作品来体验满足真实受众需求的价值,如娱乐他人、使他人投入、装备他人、教育他人等,从而使更多的人使用他们的创建成果,欣赏他们的创建成果。质疑观念,是指青少年设计者有权提出有关技术的问题和使用技术的问题——“我能

够(使用计算)提出问题来理解世界的计算性事情”,比如有些青少年设计者提出了 Scratch 编程工具功能不足、受限等质疑性问题,并给出自己的建议。以上三种计算观念的简要描述见表 3^[8]。

表 3 计算观念列表和简要描述

观念	描 述
表达	认识到计算是创作的媒介(我能够创作)
联系	认识到利用和为其他人创作的力量(当我与其他人联系的时候我能够做与众不同的事情)
质疑	感觉被赋予提出有关世界的问题的能力

值得关注的是,MIT 提出的计算思维三维框架可以与我国国内中小学信息技术教师熟悉的三维目标近似呼应。当前中小学信息技术课程目标——信息素养具有三个维度,即知识与技能维度(信息技术基础知识和信息技术工具操作)、过程与方法维度(用信息技术解决问题的能力)和情感态度与价值观维度(信息安全意识、使用规范和道德准则以及学习兴趣);计算思维也具有三维目标,即知识与技能(计算概念)、过程与方法(计算实践)和情感态度与价值观(计算观念)。可见,计算思维不仅仅局限于算法与编程等概念、知识和原理层面,还包括超越算法和编程之外更多的学习结果,包括学生在创建互动媒体过程中的问题分析与解决能力、系统思考与设计能力、社交能力、人格塑造与思维品质发展以及价值观的形成。计算思维三维框架将学生利用编程工具创建互动媒体活动中不断形成的计算概念、计算实践与计算观念有机融合,为我们有效地表征和评价学生的计算(编程)活动和发生在互动媒体编程过程中的学习结果提供了重要的理论框架。

三、基于计算思维三维框架的 Scratch 创意计算课程案例分析

基于计算思维三维框架,MIT 终身幼儿园研究小组开发了旨在课堂中培育计算创意(Computational Creativity)能力的 Scratch 创意计算课程,并于 2010—2013 年暑期面向所有的媒体专家、技术协调员、计算机科学教师和其他关注新技术与中小学课程整合的各学科教师,联合谷歌计算机教师培训项目(Google's CS4HS Initiative),共同举办了教师工作坊培训活动。虽然这个课程是为教师培训量身定制的,但其内含的课程理念、课程内容和教学策略同样适用于中小學生,它为我们开发面向中小學生计算思维发展的相关课程提供了借鉴。以下重点从课程理念与方法、课程结构与内容、教学策略、教学资源与支架等四个角度,

对 Scratch 创意计算课程设计与实施 (即暑期教师工作坊活动)进行分析^[9]。

(一)课程理念与方法

Scratch 创意计算课程的理念是教师要学会引导青少年以设计者或创建者的身份投入到创意设计活动中,在设计、创建、实验、探究和分享动态的、互动的计算媒体的过程中发展成为计算思维者(Computational Thinkers)和计算创造者(Computational Creators),以便在跨越不同学科、情境的工作、生活中都能利用计算概念、计算实践和计算观念,为未来的职业发展做好准备。由此可见,创意计算课程的目标不是单纯为了培养青少年成为计算机科学家或程序员。

创意计算是创意设计与计算思维的综合体,它强调依靠创造性、想象力和兴趣来发展个人兴趣与计算的联系,并通过创建互动计算媒体(Interactive Computational Media)来获得和体验与计算相关的知识与实践。

基于设计的学习(Design-Based Learning)是创意计算学习的主要方法,它强调设计(创建作品,而不只是使用或与作品互动)、个性化(创建对个人有意义和相关的作品)、协作(与他人协作创建作品)和反思(回顾和反思自己的创意实践)。该方法贯穿于整个 Scratch 创意计算课程中。

(二)课程结构与内容

Scratch 创意计算课程分 20 课时(每课时 60 分钟,工作坊用时三天),共有五个单元主题,学员有机会探究不同类型的创意表达和形式,在这个过程中逐步发展计算概念和计算实践的熟悉度和流畅性。Scratch 创意计算课程整体结构^[10]见表 4。

表 4 Scratch 创意计算课程整体结构

主题	描述	课时
简介	通过范例作品和动手体验,了解创意计算和 Scratch	2
艺术	通过创建包括音乐、设计、绘图、舞蹈等要素的作品来探究艺术。重点学习计算概念(顺序和循环)和计算实践(递增和重复)	3
故事	学生通过创建包括角色、场景和叙述的作品来探究讲故事。重点学习计算概念(并行和事件)和计算实践(再利用和再创作)	3
游戏	学生通过创建定义目标和规则的作品来探究游戏。重点学习计算概念(选择条件、运算符和数据)和计算实践(测试和调试)	4
最终作品	学生通过确定一个要开发的项目、与他人合作改进项目和展示作品及开发过程来创建一个独特的作品。重点学习计算实践(抽象和模块化)	8

从上述 Scratch 创意计算课程整体结构来看,作品类型和计算概念/实践的不同要素是组织课程结构的主要线索,代码块的学习与使用则作为隐性线索融入作品创作过程中。计算概念和计算实践的各要素渗透于各单元主题中,从易到难,逐步展开,不仅适应了初学者的认知习惯,而且这些概念和实践策略对应于各单元主题所需完成的作品,如艺术作品强调顺序、循环;故事作品主要应用广播消息/接收消息的机制,与并行和事件的计算概念有紧密联系;游戏作品强调随机、逻辑判断、成绩保存,这与选择分支、运算符和数据的计算概念以及测试和调试的计算实践关系密切。

这里以第四单元(游戏)为例,具体说明 Scratch 创意计算课程的内容,游戏单元包括 4 个课时(第 9~12 课时),第 9 课时要求学员对几个有问题的 Scratch 作品进行调试并设计一个新的调试情境;第 10 课时要求学员找出一些游戏设计的通用要素并使用 Scratch 创建迷宫游戏;第 11 课时要求学员理解什么是变量和变量的应用,并从成绩、计时、敌人、关卡、奖励等不同角度拓展修改迷宫游戏;第 12 课时要求学员探究其他类型游戏(碰撞游戏、互动文字游戏、横向卷轴游戏)。从游戏单元的内容组织来看,Scratch 创意计算课程不是单纯为了学习代码块,而是将计算概念与实践的具体要求渗透于形式多样的自主探究与动手体验中,学员在设计、创建、实验、探究和分享互动计算媒体的过程中逐步理解和掌握计算概念,并内化计算实践策略。

(三)教学策略

Scratch 创意计算课程以教师工作坊形式实施,注重学员自主探究、合作学习和动手体验。主讲教师很少采用直接讲授式教学,“做中学”和“评价贯穿教学全程”是该课程比较突出的教学策略。

1. 强调“做中学”

Scratch 创意计算课程设计者在每一课时都精心设计了需要学员亲身参与的教学活动,包括规划(Planning)、联系(Connecting)、探究(Exploring)、创建(Creating)和反思(Reflecting)等类型多样的活动。比如,在第 10 课时,学员开展“游戏头脑风暴”的联活动,在小组中讨论他们喜欢的一组游戏、这些游戏的共同点和它们成为游戏的设计特征,从而找出一些游戏设计的通用要素。又如,在第 11 课时,学员分小组开展“拓展迷宫”的创建作品活动,学员们先分若干小组,每个小组承担探究一个拓展迷宫游戏的任务(如成绩、计时、敌人、关卡、奖励)。然后,进行“这是我所理解的内容”的反思活动,通过不同小组之间的相

互教学,把本小组所研究的拓展迷宫游戏的方法教给其他小组。再如,在第13课时,学员分小组开展“为最终项目做准备”的规划活动,先在组内就可能的最终作品进行头脑风暴,然后集中起来,分享学员的最终作品想法,最后每个学员按项目规划书要求(任务大纲、资源列表、故事板)填写项目计划书。

2. 评价贯穿教学全过程

Scratch 创意计算课程采用贯穿教学全过程的评价方法,创造多种机会让学员交流他们自己和他人的创意作品和创意实践,并通过收集面向过程的数据(包括声音、视频或文本)来评价学员。文本评价数据包括第13课时的作品规划材料、第16课时的作品评价材料和第18课时的作品反思材料。作品规划材料包括项目大纲(包括作品概述、制作步骤、资源列表)和作品故事板(描述各场景下的舞台活动效果)。作品评价材料主要用于学员之间的互评,要求评价者从作品意图的清晰性、功能完整性、创意吸引力等角度填写被评作品中自己喜欢的部分和建议增添、修改的部分。作品反思材料用于学员的自我反思,要求学员回顾自己做了什么(作品的功能和想法来源)?怎么做的(制作作品的过程和收获)?并思考接下来有什么打算(新作品的打算)?通过“瞻前顾后”式的反思引导学员将所学的创意计算概念与方法用于未来的课堂教学实践中。此外,评价的参与者也是多元的,除了作品创建者外,也鼓励同伴、教师、家长和其他人员参与评价。

(四)教学资源与支架

为了鼓励学员的自主探究和动手体验,Scratch 创意计算课程开发者提供了包括课时计划、纸质分发材料、作品范例和教学微视频在内的多种教学资源与支架,并提供了这些教学资源的详细清单,以方便学员下载和查找。

由于 Scratch 创意计算课程把代码块的学习与使用作为隐性线索融入不同类型作品的创作过程中,考虑到初学者理解代码块功能和用途的需要,因此课程开发者专门为不同类型的作品创作设计了三份纸质分发材料(用于艺术主题作品的有用代码块清单、用于故事主题作品的有用代码块清单、用于游戏主题作品的有用代码块清单)。这样,初学者可以对照这些清单,根据自身情况选择部分代码块进行自主探究和学习,然后在不同类型的作品创作中有的放矢地选用代码块,完成作品所需的特定功能。除了针对不同类型作品提供有用代码块清单以支持学员的自主探究外,Scratch 创意计算课程还提供了完成范例作品的简易教程分发材料,它有助于初学者跟随教程指引独立完

成作品。

四、计算思维评价初探

计算思维三维框架为表征学生投入互动媒体创建的学习结果提供了重要工具,但评价计算思维学习结果的具体表现程度,需要使用基于实证、基于数据的评价方法。MIT 终身幼儿园研究小组在多年的探究、实验中摸索出三种评价方法:作品档案袋分析、基于编程制品的访谈法、情境设计法。以下简要描述这三种计算思维的评价方法,并归纳这些方法的优缺点^[11]。

作品档案袋分析法(Project Portfolio Analysis)使用称之为 Scrape 的专门用来分析 Scratch 作品中代码块使用频度的一组可视化工具,通过将 Scratch 作品档案袋(包括一段时间内完成的 Scratch 作品)输送至 Scrape 中的“用户分析”工具,就可以产生每个 Scratch 作品已用或未用代码块的可视化图示,由此来分析计算思维的发展历程与代码块使用偏好。

基于编程制品的访谈法(Artifact-Based Interviews)是对 Scratch 网站的用户随机采样,选择访谈对象,研究人员设计了包括背景、作品创建、在线社区和未来发展在内的四组访谈问题,以了解被访者用 Scratch 做什么、如何形成作品创作的最初想法、如何应对作品创作中的困境、如何在 Scratch 在线社区与其他用户互动、是否喜欢 Scratch 编程工具等大量信息。在访谈中,被访者会选择两个自己创作的作品,与研究人员进行深入交流,从中了解 Scratch 设计者在创作作品过程中的细节,尤其是详细了解被访者的计算实践。

情境设计法(Design Scenarios)是使用预设情境对被访者进行访谈、观察的方法。研究人员事先设计三组不断递增复杂度的 Scratch 项目,每一组项目有两个 Scratch 作品,作品涉及相同的概念和实践,但有不同的美学感受,以吸引不同的兴趣。在访谈时,向被访者呈现预设情境,然后要求被访者从一组项目中选择一个作品来完成以下四项任务:(1)解释所选的作品,(2)描述它的拓展,(3)修正错误,(4)通过增加功能来再创作作品。这四个活动分别对应一些独立的活(展示、评论、调试、挑战和再创作)。

以上三种计算思维评价方法各有优缺点,其评价的侧重点也有差异^[12],具体见表5。总体来说,由于计算思维是学生在不同情境下、借助不同的支持、带着不同的学习动机、经过不同的时间跨度而发展起来的,因此对学生计算思维的评价应该综合使用多种方法,通过支持后续学习、结合编程制品、阐明实践过程、设置多个检查路径点、探究知识的方式、包容

表5

三种计算思维评价方法的优缺点及其评价侧重点

计算思维的评价方法	优点	缺点	评价计算概念/计算实践/计算观念的侧重点
作品档案袋分析法	反映出评价的形成性特点	完全面向作品,没有揭示作品制作过程;专注计算概念,没有反映任何计算实践方面的信息	可视化的代码块使用频度侧重评价计算概念;不能评价计算实践和计算观念
基于编程制品的访谈法	更好地理解 Scratch 设计者能否流畅使用特定概念;将分析焦点从作品转向过程,有助于了解 Scratch 设计者的计算实践	费时,讨论受制于 Scratch 设计者回忆,不能实时探究他们的计算实践	基于自身真实的设计经历,侧重评价计算实践;很难通过直接询问来评价计算观念
情境设计法	情境提供了系统探究知识的方式,如评论、拓展、调试和再创作以及不同概念和实践的流畅性;设计情境逐步递增复杂度,突出了发展性、形成性;情境强调了行动中的过程,访谈者能够观察到 Scratch 设计者所用的设计实践和调试策略	费时,尤其是调试和拓展活动;问题的特征以及使用外部选择的项目可能不能联系个人兴趣和学习者的内在动机	使用实时、新颖的外部项目情境,侧重评价计算实践;很难通过直接询问来评价计算观念

不同评价参与者(自身、同伴、家长、教师)的观点,才能有效地评价学生的计算思维的学习结果。

五、结束语

计算思维是近年来在高校计算机专业教育领域和中小学信息技术教育领域普遍关注的热点问题。在深刻理解计算思维的概念、特征基础上,我们还需要深入探究如何全面、恰当地表征和评价学生编程活动中的计算思维的学习结果。MIT 终身幼儿园研究小组在这一领域的理论建构与实践探索卓有

成效,他们不仅界定了计算思维三维框架,而且开发了面向计算思维三维框架的 Scratch 创意计算课程,并初步设计了三种计算思维评价方法,这为我们进一步开展中小学计算思维培养与评价的本土化理论研究与实践探索提供了重要基础。例如,我们可以研究基于计算思维三维框架的学生表现性行为特征及具有可操作性和等级体系的表现性标准^[13],也可以将计算思维三维框架应用于面向计算思维培养的教学设计,从而使计算思维培养真正融入课堂教学实践过程中。

[参考文献]

- [1] Resnick, M., Maloney, J., Monroy-Hernandez, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., & Kafai, Y.. Scratch: Programming for All[A]. Communications of the ACM[C], vol. 52, no.11, Nov.2009:60-67.
- [2] Cuny, J., Snyder, L., & Wing, J.M.. Demystifying Computational Thinking for Noncomputer Scientists [EB/OL]. [2013-10-05] <http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf>.
- [3] Operational Definition of Computational Thinking for K-12 Education [EB/OL]. [2013-10-05] <http://www.iste.org/docs/ct-documents/computational-thinking-operational-definition-flyer.pdf?sfvrsn=2>.
- [4] [5] [11] [12] Brennan, K., & Resnick, M.. New Frameworks for Studying and Assessing the Development of Computational Thinking [EB/OL]. [2013-10-05] http://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf.
- [6] [7] [8] [9] [10] Creative Computing: A Design-Based Introduction to Computational Thinking [EB/OL]. [2013-10-05] <http://scratched.media.mit.edu/sites/default/files/CurriculumGuide-v20110923.pdf>.
- [13] 李锋,王吉庆. 计算思维:信息技术课程的一种内在价值[J]. 中国电化教育, 2013, (8): 19-23.