

■ 学前教育前沿

幼儿计算思维培养的途径与方法

——基于编程教育的视角

程秀兰, 沈慧敏

(陕西师范大学教育学院, 陕西西安 710062)

摘要:国外研究表明幼儿编程教育的主要目的在于培养幼儿的计算思维,学习计算思维有利于幼儿智力品质的发展以及更好地践行STEAM教育理念;国内对幼儿编程的探索尚处在初始阶段。本文通过文献查阅探明幼儿计算思维的内涵,指出基于编程教育视角,应该从树立幼儿编程教育理念、建立编程教育体系、合理运用编程工具及以日常生活为依托来培养幼儿的计算思维,旨在为幼儿编程教育开拓新视野。

关键词:计算思维;编程教育;幼儿

中图分类号: G612

文献标识码: A

文章编号: 2095-770X(2021)03-0016-08

PDF获取: <http://sxxqsfxy.ijournal.cn/ch/index.aspx>

doi: 10.11995/j.issn.2095-770X.2021.03.003

Approaches and Methods of Cultivating Children's Computational Thinking —Based on the Perspective of Programming Education

CHENG Xiu-lan, SHEN Hui-min

(School of Education, Shaanxi Normal University, Xi'an 710062, China)

Abstract: Foreign research shows that the main purpose of programming education in early childhood is to cultivate children's computational thinking, and learning computational thinking is conducive to the development of children's intellectual qualities and better practice of STEAM education philosophy. The exploration of children's programming in China is still in the initial stage. Exploring the connotation of children's computational thinking through literature review, introducing that the cultivation of children's computational thinking from the perspective of programming education should start from establishing the concept of programming education for children, establishing a programming education system, rationally using programming tools, and relying on daily life in order to provide a new perspective for early childhood programming education.

Keywords: computational thinking; programming education; children

一、计算思维与幼儿计算思维的内涵

计算思维(computational thinking)一词最先由美国卡梅隆大学的周以真于2006年提出,计算思维是利用计算机科学的基础概念解决问题、设计程序和理解人类行为的一系列思维活动,具有

多元抽象思考、反对机械记忆、强调数学和工程思维的补充与整合等特征,是每个人都需要掌握的必备技能^[1]。在儿童编程教育领域,派珀特于上世纪八十年代便提出了类似计算思维的理念,即“强大的思想”,重在强调学习者在获取新知过程中的自我探索^[2]。进入21世纪以后,教育学者

收稿日期:2020-12-25;修回日期:2021-01-07

作者简介:程秀兰,女,河南夏邑人,陕西师范大学教育学院教授,博士生导师,主要研究方向:儿童心理发展与教育,教师教育等;沈慧敏,女,陕西汉中,陕西师范大学教育学院硕士研究生。

对计算思维的内涵有了新的探索和突破。2014年欧洲学校网(European Schoolnet)所做的关于计算机编程与学校课程的调查研究中,将计算思维定义为:能够通过设置一系列指令,从已知起始点达到既定目标的一系列思维过程^[3]。

Papadakis等论述了在幼儿阶段计算思维的重要性,认为学前教育阶段的编程课程主要目的应该是帮助幼儿通过设计程序而非仅仅记住算法和操作来解决问题,在实现这一目的的过程中使幼儿的计算思维得到发展^[4-5]。致力于儿童与人类发展研究的Bers教授认为计算思维不仅是解决问题,也是表达和创造的过程,需要向幼儿提供外部创建的工具,编程语言可以有效充当这种工具以培养幼儿的计算思维,如利用可视化编程软件(ScratchJr)和编程机器人(kibo),让幼儿在游戏中探索学习和表达自己的想法,提出了幼儿阶段计算思维培养的七大要素:算法、模块、控制结构、指代、程序设计、排除故障以及了解软件和硬件^[6]。Lye等将幼儿计算思维分为三个方面:计算概念(程序的相关含义)、计算实践(实际操作下的问题解决能力)以及计算视角(幼儿能将自身与数字世界相联系的思维模式)^[7]。

我国对计算思维的研究起步较晚,2007年王飞跃^[8]认可周以真教授对计算思维的论述,同时引用恩格斯关于劳动工具在从猿到人的过程中起关键作用的观点,认为随着网络技术及其工具的发展,人类必将产生计算思维。罗力强等^[9]通过对我国计算思维研究热点的分析,发现计算思维在2012年之后才得到系统发展。罗海风等^[10]通过梳理我国学者对计算思维的定义,根据定义特点和倾向将已有定义归纳为三个视角:(1)跨学科多领域融合下的综合性视角,认为计算思维是解决复杂问题的一种必备认知工具;(2)计算机编程基础上的专业性视角,倾向于把计算思维等同于程序思维;(3)语言、算法、抽象等分项能力表征下的功能性视角,指出计算思维聚焦于创造和管理抽象事物的过程,关注的是更多抽象层之间的紧密联系。对于幼儿的计算思维,我国学者的探索较少,以借鉴国外为主。程艺^[11]依据Bers教授所划分的幼儿计算思维的七大要素,运用表格将每种要素与幼儿教育相联系,例如计算思维里的算法转变为幼儿教育相关的概念时,旨

在教会幼儿排序,让幼儿知道编程指令需要按照一定的顺序排列,才能如幼儿所设想的进行。计算思维里的每一种核心概念,都能够真正体现在幼儿的教育活动中。总而言之,计算思维是根植于计算机科学的,是一种可以被普遍接受并且每个人都应该具有的态度和能力。编程是培养计算思维的重要途径,反过来计算思维也有利于提高编程能力。

二、培养幼儿计算思维的可行性与必要性

(一)培养幼儿计算思维的可行性

要想培养幼儿的计算思维,首先要考量幼儿是否有能力学习编程。21世纪以来,教育学家开始探索计算思维和幼儿时期认知能力的关系。研究表明,通过参与以建构主义教育理念为基础的机器人教学,4岁幼儿就可以学会计算思维、编程及问题解决,只要为幼儿提供适宜其年龄发展的操作材料、课程和教育指导,幼儿就能够充满兴趣的学习编程,为深入习得计算思维迈出第一步^[12]。新加坡5所幼儿园在进行为期7周,每周一小时的机器人教学课程(STEAM KIBO robotics curriculum)后,研究者通过运用“解决它”(Solve-Its assessment)这一评价体系,对幼儿编程教育课堂进行评估,发现幼儿即使不进行额外练习,也能够快速且成功地掌握基础的程序概念^[13]。编程为幼儿提供了一种有趣的实践操作方式,对于专为幼儿设置的编程工具,幼儿在使用的过程中都能习得相应的计算思维。通过在英格兰公立幼儿园利用编程机器人对幼儿进行编程教育,发现3岁幼儿已经可以为机器人创建系统及正确的程序,5岁左右的幼儿能够更好地在既定的编程任务中达成目标^[14]。

在确保幼儿有能力学会编程之后,教师是否能够有效对幼儿编程学习进行指导,依然是培养幼儿计算思维是否可行的另一大问题。虽然幼儿编程工具不断涌现,但编程工具本身并不等于编程教育,其他领域的教学方法也不能直接迁移到幼儿编程教育之中,只有结合幼儿编程教育特点,在一定教育方法的引导下幼儿编程工具才能真正发挥实效,幼儿的编程学习及计算思维培养才能更加系统全面。基于此Bers教授创造性地提出了CAL(coding as language)教学法,即用教语

言的方法给幼儿教编程^[15]。这一教学方法突破了传统的STEM教学法,是在皮亚杰建构主义,PTD(Positive Technological Development)框架和对话指导理念的基础上所建立起来的。皮亚杰建构主义强调自主操作探索^[16];PTD框架包含幼儿编程教育所要培养的6种能力,也称6C,content creation(内容创新),creativity(创造力),communication(沟通),collaboration(合作),community building(社区建设),choices of conduct(行为选择)^[17];对话指导即让幼儿在计算机科学的教育中就某一具体项目进行开放性的解释和辩论,以此加深幼儿对概念的理解同时发展幼儿的语言表达能力^[18]。CAL教学法的教学原则如下:有利于幼儿语言教育的策略应在幼儿编程教育中使用;编程活动应为幼儿提供有意义的创造和表达的机会;解决问题应该是自我表达和交流的过程;编程活动应引导幼儿多思考^[15]。传统的STEM教学法注重在实际操作中进行问题解决,因此这种理念应用到幼儿编程教育中更强调教学活动的游戏性、操作性与探索性。比如:幼儿运用指令让人物走出迷宫,指令的组合即编程中程序的设计。Bers教授认为虽然这样的教学活动确实有利于幼儿编程能力的发展,但却忽视了编程本身的交流和表达功能,因为编程的目的在于创造有意义的程序与他人数字世界分享想法、沟通交流。迷宫这类活动仅通过指令而非系统且专业的编程语言来设计程序,因而缺乏利用编辑语言进行自我表达这一要点,而编程语言和自然语言一样都是表述自身想法的媒介,因此CAL教学法是在编程语言和自然语言之间找平衡,更有利于实现幼儿编程教育的目标。这样的教学法既给幼儿教授了编程知识,同时编程程序的设计和表达也更有利于加深幼儿对数字世界的理解,从而促进其计算思维的发展。幼儿有能力学会编程,教师有方法指导幼儿使用编程工具学习编程,因此教会幼儿编程、培养幼儿计算思维是切实可行的。

(二)培养幼儿计算思维的必要性

当今的幼儿是“数字原住民”,其成长伴随着数字工具,创新的科技产品不仅浸入到文化生活,也对每一个人的行为、思维产生着重大影响。在经济全球化的今天,各国之间的经济竞争

本质上是人才的竞争。教育作为强国之本,各个国家都在不断探索完善,计算思维被认为是21世纪的重要必备能力。中国互联网络信息中心(CNNIC)发布的第46次《中国互联网络发展状况统计报告》中显示,截至2020年6月,我国网民规模高达9.40亿,互联网普及率为67.0%^[19],在这样的互联网背景下,对幼儿进行信息素养培养,让幼儿理解自身与数字时代的交互是迫在眉睫的。2018年熊雪芹等人对866名3-6岁幼儿进行了一项调查,结果显示,866名幼儿家中都拥有电子产品,每天看电视者占99.31%,每天玩手机者占81.87%,每天用电脑者占68.36%。幼儿对于电子产品的使用不仅是为了娱乐,也包含家长为幼儿选择的益智游戏、绘本阅读等^[20]。这表明幼儿的学习和生活与电子产品息息相关,并且手机和电脑也不再是传统意义上的通讯工具,而是集通讯、娱乐、生活等多功能为一体的生活工具。

尽管过度使用电子产品会对幼儿带来消极影响,比如:导致幼儿注意力分散、出现饮食睡眠失调以及学校适应困难等^[21]。但研究同样表明,幼儿适度使用电脑、平板及手机等电子产品有利于更好地培养其结构性和概念性知识、语言能力、长时记忆、问题解决能力和肢体协调能力^[22]。因此,以电子产品为载体,以各种幼儿编程教育软件为媒介对幼儿进行编程教育是当代幼儿的发展需求。此外,幼儿编程教育的工具并不仅限于电子产品,后文所提到的不插电的计算机科学是更适合幼儿阶段的编程教育方式,幼儿不用看电子屏幕更有利于其视力的健康发展。

(三)编程被视为一种新文化

20年前信息能力被引入教育之中,并开始受到重视^[23]。2010年美国颁布的Common Core State Standards (CCSS),这项标准强调要为幼儿的信息学习提供支持,培养幼儿的数字能力^[24]。关于数字能力的定义,Bawden认为数字能力是为了与他人交流而去理解和创造多种数字文本^[25]。Gillen and Barton扩充了数字能力的含义,认为数字能力不仅包括对工具的使用和相关信息的获取,也包含个体为了达到目标运用科技工具与人交往^[26]。Hsin等认为数字能力包括学习使用电子设备的相关概念和使用方法、利用工具进行创造表达以及自己在设备上编程^[27]。在

数字能力的基础上,2018年Bers教授提出编程是21世纪的必备能力和文化。将文化定义为运用符号系统和技术工具创造一种可分享的产品(可以是文本、故事或者机器人)来与他人相互理解、组织和交流,认为这样的文化既是以文字做载体的文化,同时也是编程文化。文化的功能主要在于塑造人的思想与沟通方式^[28],作为一种新文化,编程带来了新的思维、交流和表达方式。Bers教授认为这种新的思维就是计算思维,编程和读、写一样是人们表达想法的方式,编程语言也和很多自然语言一样可以表达自己的需求与愿望^[29]。Norman提出编程语言和自然语言有三个共同点,它们都是有意义的、输出的、在时间和空间上分离的^[30]。也就是说语言是可以超越时空区传递信息的语法和句法系统,同理,幼儿编程教育的目的应该在于让幼儿通过解释性的编程语言创造出可以与他人交流表达的产品,这与Bers教授的观点不谋而合。幼儿通过使用编程语言不仅可以连接算法逻辑,还能够讲解数字故事,这在当下的数字时代是至关重要的。

三、培养幼儿计算思维的价值与意义

(一)有利于培养幼儿的智力品质

幼儿阶段是智力迅速发展的重要阶段,同时也是各种行为习惯养成的关键时期。在智力品质方面,研究表明,编程有利于培养幼儿的排序能力^[31],数学、逻辑以及目标导向思维,幼儿拥有计算为基础的学习经验,通过对自己的假设进行验证,有利于培养幼儿的科学与创新素养;通过与同伴的交流合作,有利于培养幼儿的社会情感。例如,在合作中共同操作的编程环境有利于培养幼儿的社交认知,3D或真实化的积木组件有利于培养幼儿的空间感知能力,编程语言可以利用口语中的句法规则培养幼儿语言的逻辑表达能力^[32]。编程也有利于培养幼儿的空间思维,幼儿在将自己的程序设计转变为现实的过程中,能够在心理上把现实的地图和符号与程序设计之下的机器人动作相匹配,并且能在心理上设想不同命令与机器人实际动作的关系,这种空间思维的获得有利于提高幼儿的数学能力^[33]。并且,幼儿在操作编程机器人时,也可以加深幼儿对数学概念的理解,如:数量、尺寸和形状等。此外,对

编程机器人的组装有利于培养幼儿的手眼协调能力^[34]。同时,编程也可以通过让幼儿完成指定的程序设计,达到讲故事的目的从而促进幼儿的自我表达,这种自我表达同样也是幼儿学习编程的一种方式。幼儿在具备了一定的语言表达和理解能力之后,可以根据人物及对话进行叙事,编程不再是最终的学习目标,应该注重让幼儿以趣味性的方式来表现故事内容^[35],编程的核心也正在于运用代码实现自己的想法,这一实现的过程是在讲程序员自身心里的故事,也是在解决某一具体的问题。

(二)有利于践行STEAM教育理念

STEAM是科学(Science)、技术(Technology)、工程(Engineering)、艺术(Arts)、数学(Mathematics)五门学科的英文首字母缩写,STEAM教育就是指融合这五门学科的综合教育。STEM教育起源于美国。二战以后,美国极其重视科学和技术的发展,1986年美国国家科学基金会的报告《科学、数学和工程本科生教育》拉开了STEM教育的序幕^[36]。2006年,维吉尼亚理工大学的格雷特·亚克门提出将艺术纳入STEAM教育中,创造性地提出了STEAM教育理念,使其更加完整^[37]。2009年11月,全美州际首席学校官员(教育局长理事会)发布了一项政策声明《隐藏的危机:迫切要为五岁以下的幼儿建立早期教育体系和质量计划》,首次提出在学前、早期教育阶段整合幼儿的STEM教育。自此,学前领域的STEM教育不断发展^[38]。同年,美国国际科学董事会建议奥巴马要将STEM教育放在早期教育的优先位置,并且认为孩子越早接受STEM教育,就能更好地适应未来生活,STEM教育的核心理念应包含在开端计划(Head Start)以及其他早期编程教育中^[39]。STEM教育重在培养幼儿的科学、技术、工程以及数学素养,倡导幼儿通过项目活动自我探索、主动建构,在操作中解决实际问题。研究表明,幼儿STEM教育有利于幼儿脑部发展、形成批判性思维;有利于提高社会劳动力的创新性和竞争力;有利于缩小贫富差距推进社会公平^[40]。编程教育能融合并教授科学、技术、工程、艺术和数学方面的知识,是STEAM教育的一个很好的切入点,目前已成为STEAM教育中的学习领域。美国学者发现幼儿编程教育不仅可以在传统课

堂中实现,而且还可以同STEAM教育相融合,幼儿通过操作编程机器人,可以学会机器的组装、排序、了解软件和硬件的相互联系等,这些知识可归入数学、工程等学习领域,并且装扮个性化的机器人能够促进幼儿感知美、欣赏美和创造美即艺术领域的发展。在指导思想上,STEAM教育和编程教育都以建构主义思想为主导;在教学形式上,二者都倡导做中学,通过亲身体验、操作材料在探索的过程中获取新知。因此,幼儿编程教育可以很好地贯彻实践STEAM教育理念。

四、培养幼儿计算思维的途径与方法

(一)广泛树立幼儿编程教育理念

乔布斯在一次演讲时谈到,当下最重要的两种语言,一种是英语,英语是让人走向国际的语言;另一种就是编程,编程是引领人走向未来的语言。我国目前编程教育普及率不高,人们对编程教育的认识还存在一定偏差。大众普遍认为编程是计算机行业人才所必备的专业技能,没必要拓展到教育领域,而在幼儿时期就开始学编程更未引起人们的重视,认为幼儿时期学编程,为时过早、难度较大。但实际上,编程是儿童在未来社会生活中所必备的技能,学习编程并非要掌握代码编写能力,而是要具备新的思维方式,运用计算思维来解决问题,适应人工智能等新兴领域的发展。并且,美国幼儿编程教育领域已就幼儿能不能接受编程教育、接受编程教育对幼儿有何助益、幼儿如何接受编程教育等问题进行了大量的研究与实践。因此,我们应当树立起幼儿编程学习的观念,积极探索适宜我国国情的幼儿编程教育模式,让更多的幼儿受益于计算思维的学习。

(二)政府稳步建立幼儿编程教育体系

在互联网发展的推动下,世界各国都开始重视计算机学习和编程教育,并逐渐降低编程学习的年龄门槛。早在2010年,美国新的联邦计划将计算机科学与技术文化置于幼儿发展的优先位置^[41]。2013年,英国为向年轻学生引入计算机编程教育,在公布的国际课程框架中,将计算机编程置于教育中的重要领域,并要求从早期教育开始学起^[42]。近年来,随着我国政府对编程教育的重视和研究,加之与世界各国的学习借鉴,2017

年国务院印发了《新一代人工智能发展规划》,明确指出人工智能成为国际竞争的新焦点,应逐步开展全民智能项目,在中小学阶段设置人工智能相关课程、逐步推广编程教育。重庆市在2018年响应并贯彻国家人工智能教育方针,印发《关于加强中小学编程教育的通知》,强调小学、初中、高中编程教育在内容和水平上要有机衔接、各有侧重、避免重复。对各学段的编程教育课时和重点作了详细说明^[43]。这一系列政策的颁布,充分说明,我国已经开始重视和规范少儿编程。目前在我国盛行着一部分适合儿童编程的工具,比如时下热门的编程猫、乐高编程机器人等,但并不适合幼儿的编程教育,能够同幼儿园课程相融合并且适合幼儿的编程工具少之又少,同时我国在幼儿编程课程方面也有很大缺口。纵观美国编程教育的发展,美国已经建立起了基本的幼儿编程教育体系。我国政府应该在借鉴成功经验的同时,结合本国国情,有针对性地建立幼儿编程教育体系,为幼儿编程教育提供保障。

(三)通过编程工具促进幼儿计算思维发展

编程教育承载着幼儿计算思维的发展,美国幼儿编程教育发展迅速,主要体现在编程语言的可视化和编程机器人的完善。可视化编程语言能够使幼儿通过图形化程序模块而非输入文字代码来进行编程,这符合幼儿的心智发展水平。机器人技术的成熟为幼儿编程教育提供了新思路,研究表明将代码变成现实更有利于激发幼儿的学习兴趣,建立自我效能感^[44]。因此运用编程工具,以适宜幼儿身心发展特点的方式来引导幼儿学习编程更有利于培养幼儿的计算思维。幼儿编程工具的类型主要分为数字工具和不插电的计算机编程。

1. 运用数字工具培养幼儿的计算思维

在互联网飞速发展的今天,5G的诞生更是加速了万物互联的脚步,数字工具已然成为人们联接世界的高效媒介。部分编程软件适应幼儿身心发展特点,充当了有效促进幼儿计算思维发展的数字工具。ScratchJr是美国Bers教授所带领的DevTech Research Group与MIT Lifelong Kindergarten Group等团队于2014年在Scratch语言的基础上,结合幼儿特点,奉行“少即是多”这一理念,所设计的一款针对5-7岁幼儿的图片化编程软

件。幼儿可以在故事情境下自主设计程序,运用6种不同颜色的指令控制屏幕中人物的动作,人物类型多达150多种。设计者将幼儿在肢体、认知、语言、社会情感的发展特点和目标与软件相融合,从而实现幼儿学习和编程能力的发展,在幼儿园和家庭中应用广泛。其设计理念遵循以下原则:低门槛、高天花板、宽墙壁、灵活、欢乐、教室支持^[44]。除了上述以可视化编程语言为代表的幼儿编程教育工具,TurtleTalk^[45]是一款以语言为交互的幼儿编程软件,幼儿可以通过语言和对话控制屏幕上的Turtle,从而达成自己的行为目标,并从Turtle的动作中获得反馈。智能化的Turtle能够将幼儿表述的自然语言与基于神经网络模型的Turtle的图片语言(Turtle Graphic Language (TGL))进行匹配,并执行命令。研究结论表明,TurtleTalk有利于培养幼儿序列、循环等基础的程序概念,语言的使用让幼儿在编程游戏中更有代入感,同时降低幼儿的编程难度,树立编程自信。

2. 以不插电的计算机编程引导幼儿计算思维发展

除了数字工具,不插电的计算机编程在幼儿编程教学中的应用也较为广泛。不插电的计算机编程(unplugged programming)是指在不使用计算机和平板的情况下,通过肢体活动或一定的设备,教授受教育者一定编程思想及技能的计算机编程教育理念^[46]。比如用小珠拼图案;在图片中辨认模式;一名幼儿扮演机器人,另一名幼儿发出相应的指令让对方根据指令表演瑜伽或体育活动等,都属于不插电的计算机编程教育^[47]。

皮亚杰的认知发展理论将个体从出生到成熟的发展过程分为四个阶段:感知运动阶段(0-2岁)、前运算阶段(2-7岁)、具体运算阶段(7-11岁)以及形式运算阶段(11-15岁)。其中前运算阶段发展表现出符号性的特点,思维具有具体形象性,这一阶段正是幼儿园幼儿(3-6岁)所处的发展阶段。根据这一理论,在经过适当的训练之后,幼儿能够识别以符号为特征的模式,并且可以运用符号或简单的词组来进行序列和算法设计^[48]。因此运用可触摸的材料对幼儿进行编程教育是更适合幼儿的学习方式。Kibo是一款面向4-7岁幼儿的积木编程机器人,包括软件和硬

件,软件为可触摸的程序积木块(tangible programming blocks);硬件为机器人本身(the robot itself),在美国National Science Foundation投资下,由DevTech research group于2014年创立,这款积木编程机器人无需电子屏幕和键盘,是集中教师、早期教育专家以及幼儿本身的需求和期望,为满足幼儿学习发展需要而设计的^[49]。幼儿能够通过可触摸的方式利用马达、传感器和手工材料组装机器人变身工程师,通过探索序列、循环及变量等承担程序员的角色。在操作机器人的过程中,还可以增进幼儿对于算法、模块化、控制结构等概念的掌握,在此学习和实际操作过程中促进幼儿计算思维的发展^[50]。幼儿思维发展以具体形象思维为主,直接感知、亲身体验和实际操作更有利于幼儿的学习和理解,将编程语言及思维嵌入实体机器人,既能让幼儿将想法变成现实激发其学习兴趣,又能有效促进幼儿对知识的内化吸收。相对于编程软件,编程机器人可以培养幼儿在现实中的动手能力,幼儿不用看电子屏幕也更有利于他们的健康成长^[51]。

3. 数字工具与不插电的计算机编程相结合

Otterbor等对瑞士某幼儿园的109名幼儿教师进行调查,结果表明前文提及的两种幼儿编程教育方式在实际教学过程中,通常是二者相互结合使用的。一方面,体现在编程数字工具的导入活动中。此项调查显示有31名幼儿教师会先对幼儿进行不插电的计算机编程教育,幼儿获得具体经验之后,再让幼儿操作数字工具进行编程教育。这种编程理念的提前感知让幼儿的编程学习更有意义。另一方面,体现在两种教育活动的补充和融合。例如:让幼儿先利用数字工具(手机或平板),在城市中步行并拍照来探索了解自己的居住地,回到幼儿园后重现拍摄地形成地图;然后运用编程设计程序,让机器人抵达地图中的指定地点最终回到幼儿园^[52]。两种幼儿编程教育方式的结合,实则是幼儿直接经验与间接经验的相互联系和作用,将抽象的程序设计与幼儿日常生活相联系,更有利于加深幼儿对编程的理解,从而有效培养其计算思维。但如何在特定主题下将幼儿编程的数字工具与不插电计算机编程有效结合,还需要幼儿教育工作者进一步探索。

(四)以日常生活为依托培养幼儿的计算思维

家庭教育对幼儿个人成长及社会适应都有深刻影响,然而现实中有一些家长考虑到幼儿年纪较小,很多事情都包办,为幼儿代劳,这是不利于幼儿发展的。根据埃里克森的人格发展理论,3-6岁儿童的发展冲突任务为:主动对内疚的冲突。在这一时期,如果幼儿表现出的主动探究行为受到鼓励,幼儿就会形成主动性,反之,幼儿则会逐渐失去自信。这表明,这一阶段的幼儿随着年龄的增长有能力也需要参与一些家庭事务,并从中获得自我效能感。通过参与家庭事务,幼儿动手动脑的能力能够得到全面的锻炼,逐渐形成适应未来生活的基本素质和基本能力,其中思维能力也可以得到很好的提升。就计算思维而言,幼儿通过计算思维的相关学习,不仅有利于加深幼儿对信息时代生活环境的理解,而且可以利用所学到的编程能力和计算思维去解决实际的生活问题。

在家庭生活中,家长以实际的生活问题为教育契机充分利用日常生活来培养幼儿的计算思维是极具智慧和针对性的。首先,规划日程安排和事物处理顺序。如周末一家人外出游玩,可以让幼儿进行大致的规划,先去哪里再去哪里。如果去朋友家做客或接待客人,也有一系列的流程,幼儿熟知这些流程后,再具体安排时可以考虑事情的先后,用时的长短,有利于养成良好的思维习惯。第二,整理物品或资料。如家长在家中培养幼儿整理物品的习惯,教会幼儿整理家中常见的物品,并且持续下去。在物品和资料的整理中,幼儿逐渐学会思考物品分类、组合、打包、储存等。这一过程需要幼儿思考物品的特点,不同事物有没有关联或排斥等,因此对幼儿思维训练有很大帮助。比如,教幼儿整理衣物,裤子、上衣、袜子分开存放,或者按季节存放^[53]。

随着科技发展和信息化2.0时代的到来,计算机的学习与应用越来越受到人们的关注,但编程作为计算机领域的重要思维和技能在幼儿编程教育领域尚未受到关注。国外就幼儿能否学习编程、幼儿如何学习编程以及教师如何指导幼儿学习编程均已开展了一系列研究和实践,结果表明3岁幼儿已经具有学习编程的能力、幼儿主要通过数字工具和不插电的计算机

编程来学习编程、教师可以通过STEAM教学法和CAL教学法指导幼儿的编程学习。未来,幼儿计算思维与幼儿编程教育研究领域,不仅需要更多前沿概念的引进,更需要研究者着眼于幼儿编程教育实践的探索、幼儿编程教育工具的研究以及本土化幼儿编程课程的开发,让更多幼儿能够受益于编程。

[参考文献]

- [1] Wing J M. Computational thinking[J]. Communications of the ACM, 2006, 49(3): 33-35.
- [2] Papert S. Mindstorms: Computers, children, and powerful ideas[M]. NY: Basic Books, 1980.
- [3] Balanskat A, Engelhardt K. Computing our future: Computer programming and coding—Priorities, school curricula and initiatives across Europe[M]. European Schoolnet, 2014.
- [4] Wang H Y, Huang I, Hwang G J. Comparison of the effects of project-based computer programming activities between mathematics-gifted students and average students[J]. Journal of Computers in Education, 2016, 3(1): 33-45.
- [5] Papadakis S, Kalogiannakis M, Zaranis N. Developing fundamental programming concepts and computational thinking with ScratchJr in preschool education: a case study[J]. International Journal of Mobile Learning and Organisation, 2016, 10(3): 187-202.
- [6] Bers M U. Coding as a playground: Programming and computational thinking in the early childhood classroom[M]. Routledge, 2017.
- [7] Lye S Y, Koh J H L. Review on teaching and learning of computational thinking through programming: What is next for K-12[J]. Computers in Human Behavior, 2014, 41(dec.):51-61.
- [8] 王飞跃. 从计算思维到计算文化[C]//中国科学技术协会学会学术部. 新观点新学说学术沙龙文集7:教育创新与创新人才培养. 中国科学技术协会学会学术部, 2007.
- [9] 罗力强,王冬青,方远豪,等. 我国计算思维的研究热点及趋势分析[J]. 中国教育信息化, 2020(2):1-6.
- [10] 罗海风,刘坚,罗杨. 人工智能时代的必备心智素养:计算思维[J]. 现代教育技术, 2019, 29(6):26-33.
- [11] 程艺. 美国幼儿编程教育初探[D]. 上海:上海师范大学, 2019.
- [12] Bers M U, Flannery L, Kazakoff E R, et al. Computational thinking and tinkering: Exploration of an early child-

- hood robotics curriculum [J]. Computers & Education, 2014, 72: 145–157.
- [13] Sullivan A, Bers M U. Dancing robots: integrating art, music, and robotics in Singapore's early childhood centers [J]. International Journal of Technology and Design Education, 2018, 28(2): 325–346.
- [14] Elkin M, Sullivan A, Bers M U. Programming with the KIBO robotics kit in preschool classrooms [J]. Computers in the Schools, 2016, 33(3): 169–186.
- [15] Bers M U. Coding as another language: a pedagogical approach for teaching computer science in early childhood [J]. Journal of Computers in Education, 2019, 6(4): 499–528.
- [16] Schutz R E. Mindstorms: Children, computers, and powerful ideas [J]. Journal of School Psychology, 1981, 19(3): 279–280.
- [17] Shah N. Designing Digital Experiences for Positive Youth Development: From Playpen to Playground [J]. Journal of Children and Media, 2012.
- [18] Andrews J. Educational Dialogues: Understanding and promoting productive interaction [J]. International Journal of Lifelong Education, 2011, 30(6): 831–832.
- [19] 中国互联网络信息中心(CNNIC).第46次中国互联网络发展状况统计报告[EB/OL],中国互联网研究中心.http://www.gov.cn/xinwen/2020-09/29/content_5548175.htm.
- [20] 熊雪芹,刘佳,石菡,等.屏幕时间与亲子关系及学龄前儿童社会能力和行为问题的关系[J].重庆医学,2018,47(14):1917–1921.
- [21] Trifunović A, Čičević, Lazarevi D, et al. Comparing tablets (touchscreen devices) and PCs in preschool children's education: testing spatial relationship using geometric symbols on traffic signs [J]. IETI Transactions on Ergonomics and Safety, 2018, 2(1): 35–41.
- [22] Clements Douglas, Sarama Julie. Young Children and Technology: What Does the Research Say? [J]. Young Children, 2003, 58(6): 34–40.
- [23] Coatney S. Information power: Building partnerships for learning [J]. Teacher Librarian, 1998, 26(1): 9.
- [24] Decarlo M J, Grant A, Lee V J, et al. Information and Digital Literacies in a Kindergarten Classroom: An I-LEARN Case Study [J]. Early Childhood Education Journal, 2018, 46(3): 265–275.
- [25] DAVID BAWDEN. Origins and Concepts of Digital Literacy [J]. Digital Literacies Concepts, 2008, 37(1): 173–177.
- [26] Gillen J, Barton D. Research briefing for the TLRP–TEL (Teaching and Learning Research Programme – Technology Enhanced Learning [M/OL]; London: London Knowledge Lab, Institute of Education, 2010.
- [27] Hsin C, Li M, Tsai C, et al. The Influence of Young Children's Use of Technology on Their Learning: A Review. [J]. Educational Technology & Society, 2014, 17(4): 85–99.
- [28] Bers M U. Coding as a literacy for the 21st century [J]. Education Week, 2018.
- [29] Bers M U. Coding, playgrounds and literacy in early childhood education: The development of KIBO robotics and Scratch Jr [C]//2018 IEEE Global Engineering Education Conference (EDUCON). IEEE, 2018: 2094–2102.
- [30] Kent L. Norman. Cyberpsychology An Introduction to Human–Computer Interaction [M]. Cambridge: Cambridge University Press, 2017.
- [31] Kazakoff E, Bers M. Programming in a robotics context in the kindergarten classroom: The impact on sequencing skills [J]. Journal of Educational Multimedia and Hypermedia 2012, 21(4): 371–391.
- [32] Amanda S, Umaschi B M. What they learn when they learn coding: investigating cognitive domains and computer programming knowledge in young children [J]. Educational Technology Research and Development, 2019, 67(3): 541–575.
- [33] Palmér H. Programming in Preschool—With a Focus on Learning Mathematics [J]. International Research in Early Childhood Education, 2017, 8(1): 75–87.
- [34] Sullivan A, Bers M U. Robotics in the early childhood classroom: learning outcomes from an 8-week robotics curriculum in pre-kindergarten through second grade [J]. International Journal of Technology & Design Education, 2016, 26(1): 3–20.
- [35] Mannila L, Dagienė V, Demo B, et al. Computational Thinking in K–9 Education [C]. Integrating technology into computer science education, 2014: 1–29.
- [36] 王春林. 基于项目活动的幼儿STEM课程实施的行动研究 [D]. 武汉: 华中师范大学, 2019.
- [37] Yakman G, Lee H. Exploring the exemplary STEAM education in the US as a practical educational framework for Korea [J]. Journal of the Korean Association for Science Education 2012, 32(6): 1072–1086.
- [38] 丁嫻. 美国学前阶段幼儿STEM课程研究 [D]. 上海: 华东师范大学, 2018.
- [39] National Board of Science (2009). National Science Board STEM education recommendations for the President–Elect Obama Administration [EB/OL]. Retrieved from https://www.nsf.gov/nsb/publications/2009/01_10_stem_rec_obama.pdf
- [40] 陈倩茹. 美国学前STEM课程的研究及启示 [D]. 上海: 华东师范大学, 2019.