

基于App Inventor培养小学生计算思维的应用研究

袁 源

(苏州工业园区星洲小学 江苏苏州 215000)

摘 要: 学生计算思维的培养是近年来中小学信息技术教育中最为重要的思想之一,是Stem教育的重要组成部分,对学生信息技术核心素养的培养有着重要作用。文章主要阐述了如何利用App Inventor工具,采用项目式教学法,对学生进行计算思维方面的培养。

关键词: App Inventor 计算思维 stem 学习情境

中图分类号: G623 **文献标识码:** A

DOI: 10.12218/j.issn.2095-4743.2022.29.016

一、计算思维与项目式教学法

计算思维是卡内基梅隆大学计算机科学系主任Jeannette M.Wing教授在2006年提出的教育理念,是近十几年来信息技术教育、计算机科学教育最为重要的思想之一^[1]。计算思维是运用计算机科学的基础概念进行问题求解、系统设计以及人类行为理解等涵盖计算机思维之广度的一系列思维活动。计算思维的本质是抽象和自动化,它反映了计算的根本问题,即什么可以被有效地自动进行。计算是抽象的自动执行,自动化需要某种计算机去解释抽象。从操作层面来说,计算就是如何在一台计算机上去求解问题,也就是要确定合适的抽象。

计算思维中的抽象又区别于纯数学的抽象,数学的抽象是抛开现实事物的外在的、物理上、化学上、生物上的一切特性,只研究其数量和空间关系。计算思维中的抽象远超如此,比如链表、堆栈、树等常见的计算机数据结构,我们肯定不能仅用数学上的加减乘除对其运算。其次,计算思维中的抽象是对具体现实问题的抽象,而客观现实问题往往极具复杂性,因此往往将系统的方方面面都考虑到的话,系统就会变得无比的复杂,系统最终难以实现。这就使得计算思维需要根据不同的层次,选择性地忽略部分不重要的细节,分析系统的主要矛盾,最终控制系统的复杂性。并且为了降低系统的复杂性,确保自动化的实现,一般也会要求计算机厂家开放不同抽象层次之间的接口或者不同层次之间的翻译工具。

项目式教学法鼓励学生在老师的指导下进行信息的收集、方案的设计、项目的实施及评价等。项目式教学法中的项目可以来源与真实的情境,教师可以鼓励学生利用程序设计解决平时遇到的一些问题,以此训练学生对周围事物的抽象。笔者对学生计算思维培养研究,主要是采用项目式教学法。

二、计算思维与Stem教学理念

Stem是一种综合教育,是指集合科学(Science)、技术(Technology)、工程(Engineering)和数学(Mathematics)等多个学科领域的综合教育理念。Stem强调知识的综合性,意在培养学生的综合使用各种知识解决问题的能力。计算思维是Stem教育的重要组成部分,2018年12月份,美国白宫公布的《制定成功路线:美国的STEM教育战略》中重点指出:使计算思维成为所有教育的必要组成部分。Stem是一种先进的教学理念,并且和计算思维是相通的,我们在计算思维培养的过程中可以很好地融入Stem。Stem教育理念下,我们可以从以下几两个方面进行计算思维的培养。

1. 化繁为简,培养学生的抽象思维

现实情况纷繁复杂,同时小学生知识经验又相对缺乏,教师需要引导学生突破抽象思维能力这一难关,在应用抽象思维的最初阶段中,教师需要应用具体的形象,引导学生逐渐适应于阶梯形学习理解难度^[2]。我们鼓励学生对现实的问题进行归纳总结,抽取事物的共性,再利用这些抽象的概念去描述具体的事物,实现从具体到抽象再到具体的一个转变。

以小学程序教学中的画三角形为例,首先我们让学生画出一个边长60个单位的正三角形,并观察它,总结出该正三角形的特征:每个角 60° ;有三条边;每条边60个单位。观察后,学生自己总结(必要时老师帮助)正三角形每个角 60° 和有三条边是每个正三角形的共性,而每条边60个单位是该三角形的个性。此时我们引导学生画出来一个这样的形状:三条边、每个角 60° 、未知长度的边,其中未知长度的边用参数传入。此时,我们就实现了对一个具体的三角形的抽象。只要我们传入不同的边长,就可以利用该抽象去实现各种不同大小的正三角形。其次,我们还可以积极引导学

将正三角形再抽象为正多边形，我们传入边数，计算出每个角的角度，如此，又实现了从正三角形到正多边形的抽象。并且这些抽象可以与数学、科学等知识紧密结合，促进Stem教育的发生。

2. 化整为零，培养学生的工程思维

学生在使用程序设计解决现实问题的时候，往往面临的是一个对他们来说很大的工程问题，这个时候很多孩子会显得无从下手。教师可以先尝试带着学生对这个项目进行分解，对于六年级的学生来说，甚至可以让让学生自己分解项目。绘图能力比较强的学生可以将他们对于项目的讨论制作成思维导图或者流程图；美工比较好的可以构建程序的界面；最后大家各自完成程序设计的一部分。这对于教师或者学生团体里的领导者提出了更高的要求，这要求我们在分解项目时要尽量做到合理，各人负责的功能模块不能重合，不能出现眉毛胡子一把抓的情况。培养工程思维和上述的培养抽象思维是紧密联系不可分割的，我们只有将抽象做得非常的合理，各种抽象之间做到功能单一、职责明确，我们才能更好地对项目进行分解，实现项目的工程化。

三、App Inventor对计算思维的培养

App Inventor是一个开发移动端应用程序的编程环境，可以通过模拟器运行在安卓端或者IOS端，该环境是一个完全在线的web网页。平台对复杂的编程语言进行封装，以拖拽控件块的方式构建安卓应用，类似于Scratch程序设计。该平台不仅可以在手机端模拟程序的运行，并且能够将学生的项目导出成为一个可安装的安卓应用程序，运行在安卓手机端或者平板端。实际动手制作出来有一定使用价值的App，能够更好地激发学生的创作热情，也能增强学生发现问题、解决问题的能力。在具体实践中，App的创作往往依托于真实的情境。比如，各个班级都有图书角，学生会发现我们可以开发一个小型的图书管理系统，用于班级的图书角管理；又比如，学校运动会上，裁判员使用的秒表计时器，我们可以很容易使用App Inventor创建一个移动端的秒表应用。

对于学生来说，借助App Inventor，学生们可以个性化地解决问题，不仅可以潜移默化地学会基础的程序设计，并且在创作的过程中，学生会遇到很多的问题，在问题解决的过程中，学生的问题解决能力会得到提高，算法思维会得到锻炼。基于App Inventor培养小学生的计算思维，我们主要设计有抽取教学情境、制作需求文档、编码调试和作品展示这四个环节，如图1所示。教学情境是贯穿于整个教学环节的，从需求的分析、编码调试、作品展示都需要在我们的真实情

境中进行。每一个环节，学生们都需要协作学习，互相帮助，学生需要一起确定项目的功能，并一起制作需求文档；编码调试阶段也需要小组间互相合作共同完成；最后的展示环节，小组间也可以分工合作，确定好讲解人、操作人、幻灯片制作人等。下面是笔者利用App Inventor对小學生进行计算思维训练的一些具体的经验总结。

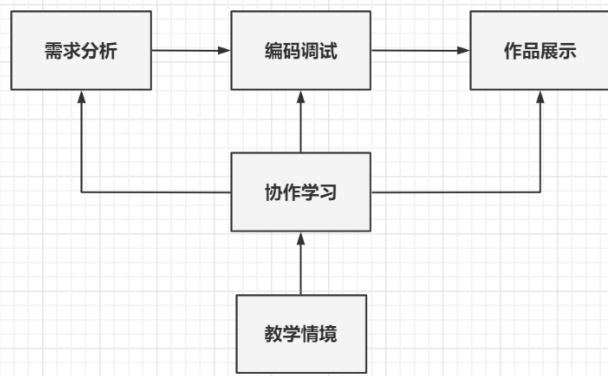


图1 设计的四个环节

1. 抽取教学情境

知识的意义寓于情境之中，学习情境不是一个无关因素，学生必须通过具体的情境才能获得某种知识，教师应该成为良好学习情境的创设者，学生乐于建构知识的促进者^[3]。教学资源最好来源于真实的情境，在真实情境下，学生的创作才会更有意义，学生可以更好地理解程序设计是为了解决生活中的问题，考虑到真实情境的复杂性。并且对于小学生来说，学生当前身心发展还很不成熟，我们很难完全用程序模拟现实情况，需要对真实情境抽取一些简单的共性特点。对于抽取后的情境，最好确保学生能够在教师的指导下完成程序设计。

以学生制作的图书管理系统为例，教师和学生一起概括出主要的业务：图书角书目浏览；借出书目浏览；学生借书；学生还书；添加书目；删除书目。剔除掉复杂的非主要的业务，比如书目超期未归还、书目损坏赔偿等复杂业务逻辑。

2. 制作项目需求文档

项目需求文档的制定非常重要，因为小学生思维不成熟，他们往往还没想清楚就开始动手编码，做到一半发现项目进行不下去。学生做的需求文档不需要面面俱到，但整体的思维导图、主要的业务流程和用户交互是必须的。学生制作文档时，需要小组间共同协作（如图2），确定好依据现有技能能够完成项目的主体功能，对于复杂模块，小组间也需要确定有一定的解决方案，或者通过学习能够解决。学生的

需求分析需要教师过目，以防止不切实际的功能。比如，学生制作的“记作业系统”中的“导入作业功能”，要提示学生，作业文件类型有很多种：Word，PPT，Excel，每种文件的模板也有很多种，我们的程序如何解析上传的作业文件类型？我们需要引导学生在完成主要业务的前提下，再去设计复杂的需求。



图2 小组间共同协作

3. 编码调试

项目需求文档设计好后，学生开始着手编码，编码过程中，如果学生有没接触过的知识点，可提前指导学生。遇到复杂的功能模块，可指导学生画流程图，通过流程图的方式降低编码的难度。编码是计算思维培养的最重要的一环，因为在编码环节，我们需要将具体的现实进行抽象，在小学阶段，学生还没学过“面向对象”的编程，笔者在抽象这一环节的通常是指导学生尽量制作单一的可复用的积木或者程序块。

即便我们是做非常简单的项目，项目中的错误往往还是难以避免。编码完成后，应带领学生多加调试，找出有问题的地方并修正。

4. 作品展示

经过几轮编码调优后，可以鼓励学生向自己的家人或者同学介绍自己的创作过程、项目中遇到的困难，以及如何解决这些困难的，邀请身边的人体验自己的作品（如图3）。我们需要尽可能依据真实的情境，赋予项目一定的实际意义。比如，教师可以带领学生举办一个小型运动会，利用我们的“秒表”程序给运动员们计时。还有，我们也可以在手工时，利用“水平仪”程序检验一些手工作品表面是否平整等等。这些都是很好的贴近现实的应用。学生可以以此充

分体会到项目创作的乐趣，并且在这个阶段可以收集同学们的体验反馈，为下面的程序优化做准备。



图3 作品展示

结语

综上所述，“抽象”是培养计算思维的核心，具体程序的“模块化”“可复用”是微观的抽象，而程序设计前的项目需求分析、功能模块设计、思维导图、流程图的设计是宏观上的抽象。在App Inventor的教学实践中，我们可以充分利用项目式教学，在真实的情境中去激发学生的创作热情，整个项目过程中学生应协作学习，共同解决项目中遇到的问题。学生从无到有地设计一个App，本身就是一个极具抽象的过程，从零到一去编码实现自己的设计也是一个从抽象到具体的过程。并且使用App inventor创作真实的应用程序，相比较于其他的块语言编程工具来说，有较强的现实意义，也更能模仿真实的情境。

参考文献

- [1]Jeannette M.Wing.Computational Thinking[J].Communications of the ACM, 2006, 49 (3):33-35.
- [2]徐刘志.小学信息技术学科学生计算思维培养的策略与方法[J].才智,2019,(36):104.
- [3]林崇德,罗良.情境教学的心理学诠释——评李吉林教育思想[J].教育研究,2007(02):72-76,82.

作者简介

袁源（1990—），男，汉族，江苏人，教师，研究生，苏州工业园区星洲小学，研究方向：Stem教学与课程融合。