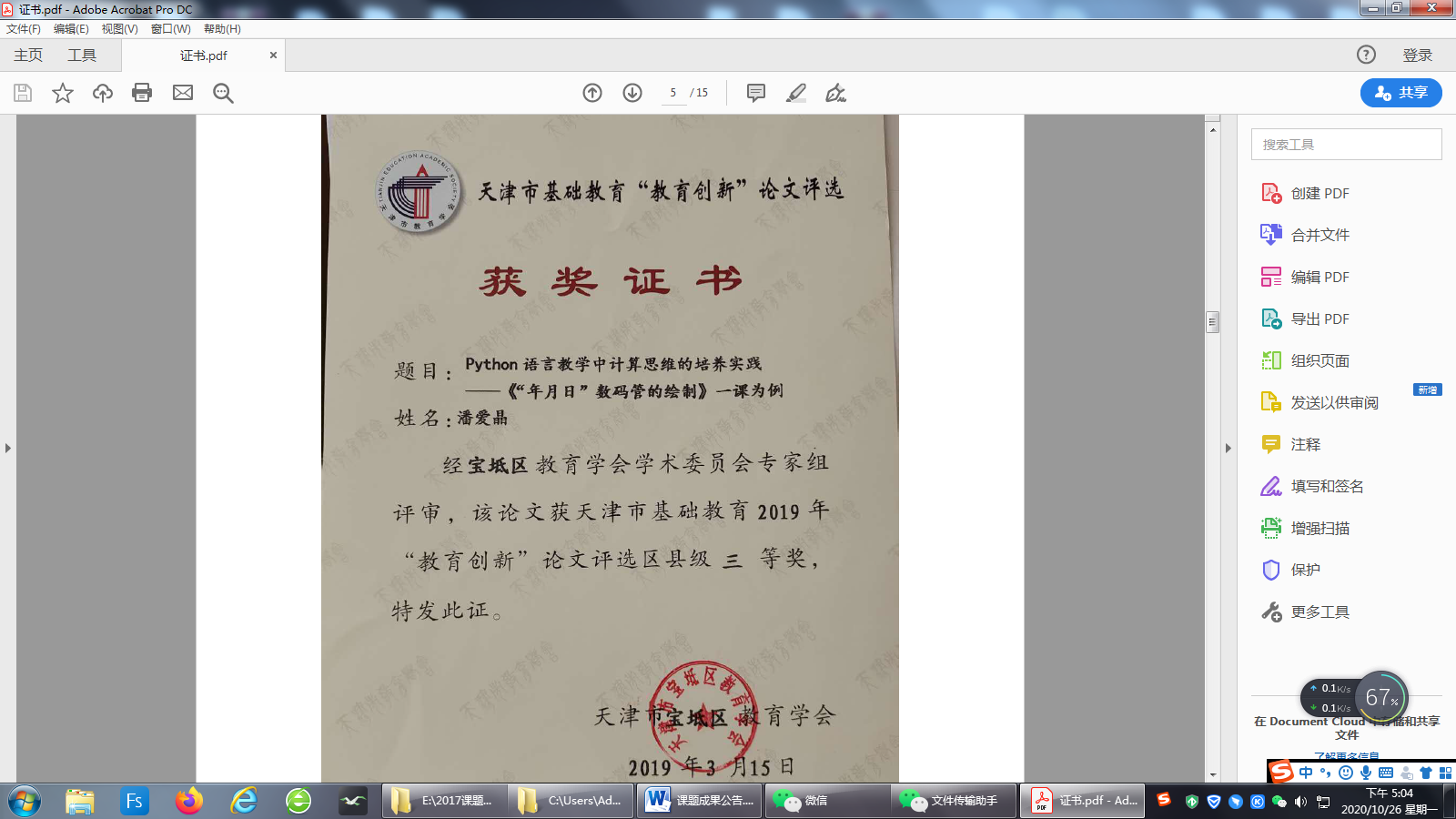
Python语言教学中计算思维的培养实践

——《“年月日”数码管的绘制》一课为例



摘要：在中小学信息技术课标实施过程中如何有效培养学生的计算思维，是当前信息技术教师十分关注的内容，本人阐述了通过Python语言培养计算思维能力的原因，《“年月日”数码管的绘制》的实例培养学生计算思维的方法，通过教学实践，笔者提出两点对计算思维的思考。

关键词：计算思维 Python

引言

计算思维、逻辑思维和实证思维是人类三大科学思维方式。在2017版的《普通高中信息技术课程标准》中提出：将计算思维作为信息技术学科核心素养的重要组成部分。目前，随着大数据、人工智能、物联网、计算生态等新科技和新概念的提出与发展，计算思维已被大家知晓，并且将逐步成为人类生存发展的技能之一，那么如何在高中信息技术课程中培养学生的计算思维呢？这也成为高中信息技术教师的重要研究项目。

一、在Python语言中培养计算思维的可行性分析

面向人工智能时代，基于计算思维的程序设计能力，将是人和机器协同工作，人机深层交互的一项基本技能。计算思维在复杂专业问题求解方面的归宿是程序设计。对学生来说，使用Python语言作为通往计算思维世界的钥匙是由高中信息技术课程的现状和Python语言的特点决定的。

(一)高中信息技术课程现状分析

我每年在高一新生入学时，都会发一份调查问卷，以了解学生的计算机基础能力，和学习方向，为将来的教学做好准备。通过调查得出，学生们都能认识到程序设计语言的重要性，但同时表示学习编程语言是很困难的，主要原因在于教师在讲授程序语言课程时，教学重点是基本的语法学习和实例应用，采用的授课方式单一，造成课堂枯燥无味，导致学生兴趣不高。

（二）python语言特点

从信息技术课程的现状和计算思维能力培养的焦点来看，我们应选择能更好地融入信息技术课堂，引领学生真正进入复杂信息系统时代的程序设计语言——python,原因如下：

（1）python语言易学习

Python语言是一门语法简单，目前最接近自然语言的通用编程语言。相比其他语言，python能更加纯粹地表达算法和程序逻辑意图，它使学生将更多精力放在应用计算机解决问题的思路和方法上，而不是浪费在语法的练习上，更能训练学生抽象问题和设计算法的能力。Python语言简单易学、学生易于理解和书写，降低了学生的学习门槛，适用于零基础的高中生。

（2）Python 语言帮助学生克服畏难情绪

实际上，以往程序设计入门课程并未纯粹训练学生分析问题、解决问题的程序设计逻辑，而将大量学时用于编程语言所附加功能的教学上，学生在未理解基本程序逻辑前提下再去理解较高级的计算概念，学习曲线陡峭，难以保持学习热情。 Python具有相当简洁的语法，对程序逻辑设计比较纯粹，学习曲线平滑，改变了学生学习编程语言的畏难情绪，有助于保护学生学习热情。

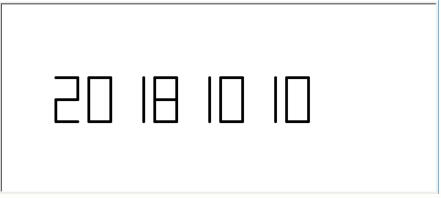
第三，Python 语言教学将能够激发学生的创新性。Python 语言有大量的第三方库，几乎覆盖了计算的各个领域，其中部分库适合初学者进行程序设计探索。例如学生使用turtle库来绘制同切圆，五角星等作品，还有许多学生使用Python的第三方库创作出令人惊叹的作品，比如：学校的校徽，大风车等。创新是新思想的源泉，Python 语言能够让学生结合特定编程模式探索思维空间的新事物。

二、在实际教学中培养计算思维

自2006年周以真教授提出计算思维的定义后，国内外相关学者或组织纷纷对计算思维进行了关注。目前，对计算思维没有统一的定义：但很多学者认为它的主要构成要素有：抽象、分解、算法等。在这里，我们认为。计算思维可以划分为四个主要组成部分：分解、模式识别、抽象化，和算法开发。

《“年月日”数码管的绘制》典型案例分析：

在交通灯路口用数码管来显示倒计时，家庭电子表上以数码管来显示时间，那么同学你能不能绘制一个用数码管来显示的年月日呢？如下图所示。



1. 分解

分解是把问题进行拆分，同时厘清各个部分的属性，明晰如何拆解一个任务。在此案例中，教师引导学生分析得出，这年月日中的数字可以拆分成8个单一的数字，进而发现每个数字是由一个七段数码管来显示的，那么也可以理解成一个数字是由七个基本线条组成的。所以要想画一个数字，我们要先会画一条线。方法如下：

def drawLine(draw): #绘制单段数码管

turtle.pendown() if draw else turtle.penup()

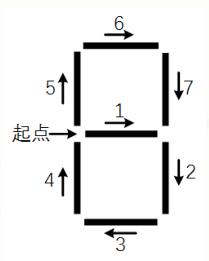
turtle.fd(40)

turtle.right(90)

代码分析：定义一个基本函数drawline（），画一条线，绘制一条线分两种情况：是真实的画一条线还是让海龟移动过去但是并不绘制一条线。通过一个变量draw来控制，如果draw是真值（true）我们就让画笔落下，画一条线；draw的值是FALSE，就让画笔抬起来，向行进方向前进40像素，并且让它转向90度，这样就构成了绘制一条线的函数。

（二）模式识别

模式识别是找出拆分后问题各部分之间的异同，为后续的预测提供依据。一个线条和七段数码管之间的关系是什么样的呢？学生观察单个数码管，由于它叫七段数码管，是由七个基本线条组成的，首先我们能绘制这七个基本线条，七段数码管的基本线条是有固定顺序的，构成了一个数字“8”的形状，但是如果我们从左边的一个位置开始，将它逐一编号，形成一个行动的序列，这样一个行动序列是可以覆盖这七段线条的，在七条线段中，绘制不同的线条，就能形成不同的数字。



使用最简单的方法就是绘制每一条线，先绘制第一条线，再往下绘制第二条线，因为draw函数绘制后会右转90度，如果我们连续调用drawline（）函数，就会绘制折线，再结合turtle.Left()绘制了七条线。在绘制过程中，不加判断drawline（）函数的值为true的话，那每一条线都会被绘制，那绘制的结果是数字“8”。为了控制它，设计一个函数drawdigit（），根据参数digit，来绘制对应的数字。Digit的取值范围是0—9，根据所给的数字来判断绘制哪条线段。从另一个角度来看，从起点绘制第一条线，有哪些数字需要绘制第一条线？数字2,3,4,5,6,8,9需要绘制中间这条线，所以判断输入的数字是以上几个数字之一，就真实的画一条线。那就是调用drawline()函数，并且参数是true。如果输入的数字不是这几个数，我们也要画一条线，但是这条线是让画笔飞过去，并不真实绘制，那就赋予参数flase。通过这种方式，我们就能够根据不同的数字决定第一条线的绘制方式。与此类似，我们决定后面的七条线的绘制方式。通过这样的设计，我们定义一个函数drawdigit（）根据用户输入的数字，来绘制出的效果。代码如下：

def drawDigit(digit): #根据数字绘制七段数码管

drawLine(True) if digit in [2,3,4,5,6,8,9] else drawLine(False)

drawLine(True) if digit in [0,1,3,4,5,6,7,8,9] else drawLine(False)

drawLine(True) if digit in [0,2,3,5,6,8,9] else drawLine(False)

drawLine(True) if digit in [0,2,6,8] else drawLine(False)

turtle.left(90)

drawLine(True) if digit in [0,4,5,6,8,9] else drawLine(False)

drawLine(True) if digit in [0,2,3,5,6,7,8,9] else drawLine(False)

drawLine(True) if digit in [0,1,2,3,4,7,8,9] else drawLine(False)

turtle.left(180)

turtle.penup()

turtle.fd(20)

（三）抽象化

抽象化是探寻形成这些模式背后的一般规律。绘制七段数码管的时候，并没有根据数字绘制一个具体的形状，而是对于七段数码管，无论是什么数字，都走了七条线，只是在走不同线的时候，用参数来控制是真实的绘制还是飞过去，事实上我们用这样的方式抽象了绘制七段数码管的过程，将这种过程变为一种规则，规则中用参数来控制它，并且由计算机自动执行，这是抽象化思维的体现。

（四）算法开发

算法开发是针对相似的问题提供逐步的解决办法。通过上面的分析，学生由浅入深的设计了“年月日”数码管的制算法规则，再由python语言编写程序来自动执行过程，成功绘制出如下图所示效果，实现自动化。七段数码管的问题有许多扩展，有的时间显示带小数点，尝试绘制带小数点的七段数码管；用数码管绘制带刷新的时间倒计时效果，绘制高级的数码管，通过应用的扩展，同学们进一步学会利用计算思维解决问题的方法。

在《“年月日”数码管的绘制》的教学过程中，引入计算思维的先进理念，把计算思维的分解、模式识别、模块化、算法引入到教学过程中，使学生理解计算思维的理念。学会运用计算思维的方法，去发现问题，寻找解决途径，最后解决问题。通过此案例，推及到Python语言的整体教学中，达到培养学生计算思维的能力，为程序设计课程探索一条新的教学途径。

三、反思

基于教学实践，在Python语言课堂中培养学生计算思维的过程中提出以下两点思考:

（一）提升教师培养计算思维的能力

在Python语言教学过程中培养学生计算思维能力的关键影响因素有二：一是教师计算思维能力的短缺。教师对计算思维的理解程度、自身计算思维水平的高低都直接决定了计算思维的教育质量、学生学习的效果，二是以计算思维为核心的Python语言课程的教学设计和教学实践能力。教师应具备完备的Python教学内容知识，掌握在教学实践中开展计算思维培养的方法，学校应组织教师参加有关计算思维基础知识及教学方法的课程。

（二）计算思维=程序设计?

有人认为发展计算思维就是培养学生编写程序的能力，这是不对的。程序设计是发展学生计算思维的一个载体，计算思维是培养学生解决问题的能力，培养学生用计算结解决问题的思维能力，计算思维不仅可以通过程序设计来培养，也可以通过数学、科学等课程来培养我们认为，计算思维能力的培养重在设计表达计算思维的教学内容。内容是知识的第一载体，是教学的纲。如果内容本身很难渗透计算或者基于计算的思维过程，计算思维能力培养就很难展开。

四、结束语

计算思维是一种科学的思维方式。Python语言肩负着培养高中生计算思维的重要使命。作为信息技术教师，要从教学中不断挖掘，积极参与到Python语言的教学改革中，构建全新的教学体系和教学模式，将计算思维的培养与教学内容有机结合，从而达到学生具有计算思维能力的目的。

参考文献：

[1]嵩天, 李凤霞, 蔡强, 等. 面向计算思维的大学计算机基础课程教学内容改革[J]. 计算机教育, 2014(3): 7-11.

[2] 陈国良, 董荣胜. 计算思维与大学计算机基础教育[J]. 中国大学教学, 2011(1): 7-11.

[3] 梁明 . 在高中信息技术教学中培养学生核心素养 [J].中国教育技术装备 ,2017(19):87-88.

[4] 郑戟明. Python程序设计课程中的计算思维的应用[J]. 大学教育, 2016(8): 127-129.

[5] 张敏. 以计算思维为导向的 C 语言程序设计课程案例教学法研究{J}微型电脑应用,2015,31(9):63- 65.