

Python语言程序设计课程教学实践

张 铭

(石河子大学 新疆石河子 832000)

【摘要】随着计算机技术与人们平常的生活与工作间逐渐深入交融,在高校教学中培养学生的编程能力也对于学生未来职业规划十分重要。本文将详细地分析《Python程序设计课程》在教学过程中所存在的问题,针对课程的特点及其中存在的思想培养与实践欠缺、预习和复习缺乏引导与监督和考核方式单一等问题,结合计算机专业人才培养方案,提出建立知识体系、课堂精讲方法、应用“雨课堂”协助教学和多样化考核等策略,通过语言程序设计课程的学习,高校的各专业的学生可以更高效地培养编程能力与计算思维。

【关键词】Python语言教学; 课程设计; 能力培养

Python Language Programming Curriculum Teaching Practice

【Abstract】With the gradual integration of computer technology and people's ordinary life and work, cultivating students' programming ability in college teaching is also very important for students' future career planning.

【Key words】Python language teaching; curriculum design; ability training

DOI: 10.12361/2705-0416-04-05-84034

对于编程课程的教学质量,国家已提出了明确的要求,利用现代化的教学手段,指引Python这种编程语言的教学质量发展向更为高端的领域,《Python语言程序设计》课程作为计算机专业的必修课程,在高校教育体系中占据着举足轻重的地位,广大教育工作者需要思考Python语言程序教学与学生培养目标相结合,进而完善当下的计算机课程体系。

1 Python语言程序设计课程教学的特点及存在的问题

1.1 Python语言程序设计课程的特点

在过去几年中,Python语言获得了大众的关注和高速发展,既得益于国家对互联网行业的重视以及Python在此领域的优秀,也源于该语言的学习难度相对较低和性能的强大以及编程体验十分友好。因此Python程序设计正逐步成为高校计算机相关专业的必修课程,并呈现出以下特点。

1.1.1 简洁性

相较于C语言,使用Python语言所编写的代码要简洁得多,并且支持两种程序设计方法:面向过程和面向对象。这门课程可以作为学生学习其他计算机课程的起点,因此在教学的过程中教师需要让学生打好坚实的编程能力基础。熟练地掌握这门语言后在未来进行数据结构等课程的课程的学习会容易很多。

1.1.2 先进性

目前在云计算、大数据等平台都提供给Python语言十分良好的就业前景。相较于C语言和VB语言,Python语言的需求性更为广泛,它瞄准的是互联网行业的未来。因此对于计算机专业的学生来讲,掌握好Python语言是在为未来的互联网行业需求做准备。

1.1.3 开源性

Python具有一个丰富的开源库,因此适用于不同领域、不同专业的同学学习和运用,也便于教师通过与学生的专业应用范围相结合进行Python语言的教学,以此激起学生的学习热情,进而让学生做到长期的学习,在未来面对各种编程问题时都能尝试利用Python语言进行解决。

1.2 Python语言程序设计课程存在的问题

1.2.1 程序设计思想培养方面和实践方面有欠缺

在Python语言的教学中最常见问题是:教师过于关注讲解Python语言的方法教学和细致知识点,而忽略了培养编程思想和实践能力的重要性,使得学生在知识储备和设计技能方面都掌握得不

错,但实际分析编程问题和解决编程问题的能力不足,对学生掌握课程规律和特点,实现知识和技能内化十分不利。而实质上,这样的教学模式已然偏离了教学目的,会继而引起学生学习效率不高,教学效果不佳,学习兴趣积极性较低等诸多问题。

1.2.2 缺乏对预习和复习方面进行有效的引导和监督机制

预习和复习也是课堂之外提高学习效果的重要手段。Python语言程序设计中覆盖的语法多且细的特点,需要教师引导好学生课前课后做好预习和复习,除此以外,也需要较为合理的监督机制帮助教师把握学生在课后复习时的实际的学习效果,如若多数课后学习都是以作业的形式完成,这便成了教师掌握学生学习进度的重要手段,但如果学生间对作业进行相互抄袭的话,就难以透过作业的完成质量识别学生存在的问题,更不利于后续调整教学方法和内容。

1.2.3 考核方式与评价体系单一

以期末考试为主的考核现今还是目前大部分Python程序设计课程的主要考核方式,而在期末考试中又是主要以理论知识的考核,缺乏考核程序设计类课程着重培养的实际编程能力、分析解决问题等动手能力,过于单调,不全面,缺乏过程性评价的考核与平均方式,并不能全面的反映学生在学习Python程序设计课程这一课程后的实际编程能力。

2 Python语言程序设计课程教学策略

2.1 整合基本知识点,建立合理的知识体系

Python程序设计课程的知识点是相互联系以及彼此衔接的,教学在原有的教学基础之上,可以补充引导学生画出知识脑图,将知识点之间的联系表示出来,再结合当下的教学进度适当地调整课程知识结构,并在最终构建一个全新的可以产生联系的课程知识体系。Python课程一般会覆盖数据类型、程序的控制结构、函数和代码复用、组合数据类型、程序设计方法论、文件和数据格式化、科学计算和可视化、网络爬虫和自动化等主要知识块。不同的教材具有自身的特点,虽然涵盖知识的范围和深度也有所不同,但不变的是基本的Python语言知识,因此关键在于教师要引导学生对学到的知识点进行整合,构建出各个单元间相联系的知识体系,再利用思维导图工具建立对应的知识脑图,让之后的学习和理解后续的知识更加方便,在梳理知识点的时候,若能将常见的知识点错误使用的范例效果整理出来也会让学生的学习效果更佳。学生在实际上手的时候,往往会对于自己做的出现了错误的程序,会让自己留下较

为深刻的印象。学生在听完老师教授的编程知识后,可以在课中或者课后的实践操作,或是利用 Python 认证考试的原题或者模拟题,运用课堂学会的知识来解决编程问题,并提交给教师,通过教师的检查、调试与修改,深化自己对课堂学到的编程知识的掌握程度。通过对这些题目进行练习,可以帮助回顾那些易于忽略和出错的知识并加深印象。

加强练习和测试。要实现计算思维能力的提高,学生们必须先理解编程语言的计算思维。结合不同的实际案例,逐步加深学生的兴趣与积极性,让他们有足够的时间去分析案例,从而理解抽象的方法和概念。培养学生计算思维时,教导学生运用 Python 语言研究解决一系列的实际问题。鼓励学生完成各种难度、各种类型的练习题目和任务,根据学习结果评估学生的能力水平。

2.2 以问题作为引导和案例分析进行课堂教学精讲

在 Python 语言课堂教学中利用问题引导学生学习,进行课堂精讲,教师可以设计出真实的情境案例,与当前学习的章节相融合。以问题作为引导、验证程序、筛选误区、总结思考这几个程序开展 Python 的教学。学生在课堂提出问题,教师引导学生将问题进一步转化,变为可以通过 Python 语言解决的问题。再运用抽象化、概括比较等思维活动,找出问题解决的方法及最有效方案。每次完成后,都要对完成任务进行归纳总结,进一步强化学习活动的成果。

教师可以布置作业,让学生在课后通过 MOOC 课程进行学习,对于 MOOC 课程中遇到的问题,教师可以根据学生受困的问题点在课堂上解答,并以此强化学生的思维能力。在进行上机训练的过程中,教师要注意训练应着重于让学生发现自己不熟练以及易被忽视的知识点。另一方面编写程序过程中出现错误是十分正常的事情,教师要教导学生不要害怕出错,要勇于多多尝试。其次是要求学生能够自主地对程序中的错误进行查找与排除,通过找出自己的错误并改正错误,加强自己的编程与算法能力。多次的上机训练之后,学生便能将错误逐渐减少,同时也提升自己查出错误原因的速度,由此加强了编程的自信心和能力。在解决实际的编程问题时,教师应当对学生提出质疑表示鼓励,并引导积极寻求更多种题目答案,多方位、多角度地思考问题。让学生明白,编程语言只是工具,真正要学习的重点是 Python 语言的思维模式以及解决问题的多种思路。

通过多种案例分析,可在较为有限的教学时间内,让学生们快速掌握 Python 语言的入门技巧,掌握相对基本的语法知识。让学生的计算思维和创新思维活跃起来,融入语法知识于典型案例之中,逐步引导,在潜移默化中加强学生的思维能力。这之后再研究更加复杂的经典案例,让学生在案例中找出错误点,并仔细修改存在于案例中的错误。鼓励学生对案例错误点发表不同的意见,允许课上要有各种各样的意见,这样能有效规避教师的“一言堂”,显著提高 Python 语言程序设计教学质量,让更多学生在这样的探索活动中产生强烈的学习意识。

2.3 应用“雨课堂”实时跟进学生学习情况

使用“雨课堂”,可以实现让教师给予学生推送学习视频和相关学习资料,而学生可以通过在“雨课堂”上进行课堂签到、回答问题、弹幕交流互动等行为参与到课堂当中,尤其是学生容量较大的大型课堂,雨课堂的应用能有效解决教学的效率问题,实现师生

实时的互动和信息交流、问题沟通。

应用“雨课堂”推送教学课件、预习题目。在课堂教学前,教师根据《Python 语言程序设计》教学目的,确定教学计划和教学内容,然后在收集和编制相关的教学资源后,及时规划课堂教学步骤,在上课前制作相关的上课资料。通过雨课堂上传资料至课程文件夹中,班级学生通过雨课堂学生端查看课件和题目,以此利用课余碎片时间预习新课程内容,避免受到时间和空间的限制,使用 MOOC 平台观看学习视频,在线完成预习习题,提高学生在课堂的学习效率。

利用雨课堂提高课堂互动频率。具体操作是教师启动雨课堂生成一个签到二维码课堂和暗号,由学生扫描签到或者是输入暗号,在投影的屏幕上老师可以看到已经到场的学生人数,课后教师还可以详细查看学生的签到数据情况,了解是否有迟到及缺勤的学生情况。此外,教师可以运用雨课堂发布课堂知识相关的测试习题,让学生在课堂上即时解答,及时发现问题并提出问题,也能提高学生的课堂参加度,增加学生的学习积极性。

2.4 课后及时进行知识巩固以及教学考核

与微积分、线性代数等传统的理科课程有所区别,编写程序代码所具有的独特的逻辑性,使得即便是小错误也会导致程序完全无法正常运行,因此 Python 程序设计课程并不是一项适合通过线下纸面试卷考核的课程,所以为了对学生进行合理的考核,更准确且更实际地反映学生的课程学习效果,便要求教师建立符合学生实际的考核评价制度。考核体系的确立首先要确定成绩组成。小组作业答辩占比 50%,在期中期末利用两次课用于考核小组答辩,教师针对小组整体答辩情况,再结合教师所了解到的各个学生在项目中作出的实际贡献,给予最恰当的个人成绩。平时的编程作业占比 30%,教师在学期内布置四个上机实操作业,要求学生运用课堂所学的知识解决这四个实际问题,应要求学生上交编程思路、过程以及运行结果,教师根据学生完成的作业质量,审核评价后给予其相应的成绩。考勤占比 20%,除了考核学生在课后的学习情况之外,课堂的学习表现也是很重要的一部分,教师要根据学生课堂出勤情况以及学生课堂回答问题的表现,给予相应的学生成绩。如此,以课堂表现、课后个人作业与小组作业等多个维度的考核不仅能够加深学生对知识的理解和应用能力与思维能力的提升,还能激起学生对学习 Python 语言的热情和积极性,进而加强学生自主分析与解决问题的能力,切实地提升课堂的教学效率和学生的编程能力。

3 结语

在 Python 程序设计课程中与雨课堂、MOOC 课程相结合的教学方式,可以有效地提高教学效率,运用小组合作作业和知识脑图,让学生在学习过程中,提高学生团队合作能力和创新思维能力,与此同时上机作业项目也能起到锻炼学生的实操能力的作用,通过多个维度的教学与考核制度,提升 Python 语言这门课程的教学质量,做到学生学习效率明显得到提升。做到培养出专业技能优秀与具有创新精神的人才,实现高校对培养高素质技能人才的培养目标。

作者简介:张铭(1981.10—),男,新疆石河子人,硕士,实习研究员,研究方向:信息技术。

【参考文献】

- [1] 王淑青,雷桂斌,金兰.Python 语言课程的线上线下混合式教学[J].福建电脑,2022,38(2):111-114.
- [2] 戴成秋.Python 语言程序设计课程混合式教学方案的设计[J].计算机教育,2022(1):162-166.