

非计算机专业研究生Python教学创新研究

郭媛¹, 吴凇², 陶波¹

¹武汉科技大学冶金装备及其控制教育部重点实验室, 湖北 武汉

²武汉科技大学机械传动与制造工程湖北省重点实验室, 湖北 武汉

收稿日期: 2025年2月26日; 录用日期: 2025年4月2日; 发布日期: 2025年4月11日

摘要

随着信息技术的飞速发展, Python作为一种易学、功能强大的编程语言, 在数据分析、人工智能、Web开发等多个领域得到了广泛应用。对于非计算机专业的研究生而言, 掌握Python编程技能不仅能够提升其数据分析和问题解决能力, 还能为其未来的职业发展奠定坚实基础。然而, 传统的非计算机专业Python教学往往侧重于基础语法和简单应用, 忽视了学生实际应用能力和创新思维的培养。因此, 本文旨在探讨非计算机专业研究生Python教学的创新策略, 通过引入项目式学习、跨学科融合、在线资源利用及创建智能评估系统等多元化教学方法, 以激发学生的学习兴趣, 提升其编程实践能力和创新思维。

关键词

非计算机专业, 研究生教育, Python教学, 教学创新, 项目式学习, 跨学科融合

Research on Python Teaching Innovation for Non-Computer Science Graduate Students

Yuan Guo¹, Lin Wu², Bo Tao¹

¹Key Laboratory of Metallurgical Equipment and Control Technology, Wuhan University of Science and Technology, Ministry of Education, Wuhan Hubei

²Hubei Key Laboratory of Mechanical Transmission and Manufacturing Engineering, Wuhan University of Science and Technology, Wuhan Hubei

Received: Feb. 26th, 2025; accepted: Apr. 2nd, 2025; published: Apr. 11th, 2025

Abstract

With the rapid development of information technology, Python as an easy-to-learn and powerful programming language, has been widely used in many fields such as data analysis, artificial intelligence, and web development. For non-computer science graduate students, mastering Python

programming skills will not only improve their data analysis and problem-solving skills, but also lay a solid foundation for their future careers. However, the traditional teaching of Python for the non-computer majors often focuses on basic grammar and simple application, ignoring the cultivation of students' practical application ability and innovative thinking. Therefore, this paper aims to explore innovative strategies for Python teaching for non-computer science graduate students, and introduce diversified teaching methods such as project-based learning, interdisciplinary integration, and online resource utilization, so as to stimulate students' interest in learning and improve their programming practice ability and innovative thinking.

Keywords

Non-Computer Majors, Graduate Education, Python Teaching, Pedagogical Innovation, Project-Based Learning, Interdisciplinary Integration

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

在信息技术日新月异的今天, Python 已成为众多领域不可或缺的工具。对于非计算机专业的研究生而言, 掌握 Python 编程技能不仅能够提升其数据分析和问题解决能力, 还能为其未来的职业发展提供有力支持。然而, 传统的非计算机专业 Python 教学往往侧重于基础语法和简单应用, 忽视了学生实际应用能力和创新思维的培养。这种教学模式难以满足当前社会对复合型人才的需求, 因此, 对非计算机专业研究生 Python 教学进行创新显得尤为重要。

2. 非计算机专业研究生 Python 教学现状分析

2.1. 教学内容单一

当前, 许多高校非计算机专业的 Python 教学内容主要集中在基础语法、数据类型、控制结构等基础知识上, 缺乏与实际应用紧密结合的案例和项目。这种单一的教学内容难以激发学生的学习兴趣, 也难以培养学生的实际应用能力。

2.2. 教学方法陈旧

传统的教学方法多以教师讲授为主, 学生被动接受知识。这种填鸭式的教学方式忽视了学生的主体性和创造性, 难以培养学生的自主学习能力和创新思维[1]。同时, 缺乏互动和实践环节的教学方法也使得学生难以将所学知识应用于实际问题解决中。

2.3. 教学资源匮乏

非计算机专业 Python 教学往往缺乏丰富的教学资源, 如教材、案例、实验平台等。这导致教师在教学过程中难以提供多样化的学习材料和实践机会, 限制了学生编程实践能力和创新思维的提升。

3. 非计算机专业研究生 Python 教学创新策略

针对非计算机专业研究生 Python 教学存在的问题, 本文提出以下创新策略。

3.1. 引入项目式学习

项目式学习是一种以学生为中心的教学方法,强调通过完成真实世界的项目来促进学习和技能发展。从教育学的角度来看,建构主义理论为项目式学习提供了重要支持。建构主义认为,学习是学习者主动构建知识的过程,而非被动接受信息。在项目式学习中,学生通过探索、实践和反思,主动构建对知识的理解,这与建构主义的核心理念高度契合。项目式学习是一种以项目为载体,通过完成具体项目来培养学生实践能力和创新思维的教学方法。在非计算机专业研究生 Python 教学中引入项目式学习,可以让学生在实际项目中应用所学知识,提升编程实践能力。同时,项目式学习还能激发学生的学习兴趣,培养其自主学习和团队合作的能力[2]。

1) 项目设计与选择

项目的设计应紧密结合非计算机专业的特点和学生实际需求,注重项目的实用性和挑战性。在项目式学习中,教师应注重创设真实或模拟真实的学习情境,设计具有挑战性的项目任务[3],并提供适当的支架支持。此外,教师还应关注学生的个体差异,设计多样化的项目任务,以满足不同学生的学习需求和智能优势。例如,对于经济管理专业的学生,可以设计基于 Python 的数据分析项目;对于工程类专业的学生,可以设计基于 Python 的数值模拟项目。

2) 项目实施与指导

在项目实施过程中,教师应充分发挥指导作用,为学生提供必要的技术支持和方法指导。同时,鼓励学生自主学习和团队合作,通过查阅资料、讨论交流等方式解决项目中的问题。此外,教师还应定期检查项目进度,及时发现并解决存在的问题,确保项目顺利实施。

3) 项目评价与总结

项目完成后,教师应组织学生进行项目评价和总结。评价内容应包括项目完成情况、团队合作情况、创新点等方面。通过评价和总结,让学生认识到自己的不足和进步,为后续学习提供借鉴和参考。

3.2. 跨学科融合教学

跨学科融合作为一种创新的教学策略,旨在打破学科界限,促进知识的整合和应用。从心理学的角度来看,认知负荷理论为跨学科融合提供了重要启示。该理论强调,在设计跨学科学习任务时,需要考虑学生的认知负荷,合理安排学习内容的复杂度和难度,以避免认知过载。通过优化教学设计,教师可以帮助学生更有效地整合不同学科的知识 and 技能。自我决定理论则为跨学科融合中的动机激发提供了指导。该理论认为,满足学生的自主性、胜任感和归属感需求,能够有效激发其内在学习动机。在跨学科融合教学中,教师可以通过给予学生更多自主选择权、设置适当挑战和营造支持性学习环境,来增强学生的学习动机和参与度。因此,在非计算机专业研究生 Python 教学中引入跨学科融合教学,可以让学生将 Python 编程技能与其他学科知识相结合,提升解决实际问题的能力。

1) 跨学科课程设计

跨学科课程设计应注重不同学科之间的内在联系和互补性。例如,可以将 Python 编程与经济学、管理学、工程学等学科相结合,设计跨学科课程。通过跨学科课程的学习,让学生掌握不同学科的基本理论和方法,为跨学科融合教学奠定基础。

2) 跨学科案例分析

跨学科案例分析是跨学科融合教学的重要手段之一。通过选取不同学科领域的典型案例,运用 Python 编程技能进行分析和解决,可以让学生深刻体会到跨学科融合教学的价值和意义。例如,可以选取经济学中的数据分析案例、管理学中的决策支持案例、工程学中的数值模拟案例等进行分析和讨论。

3) 跨学科实践活动

跨学科实践活动是跨学科融合教学的重要环节之一。通过组织跨学科实践活动，如跨学科竞赛、跨学科项目等，可以让学生将所学知识应用于实际问题解决中，提升跨学科融合能力。同时，跨学科实践活动还能培养学生的团队合作精神和创新思维。

3.3. 充分利用在线资源

随着互联网技术的飞速发展，在线资源已成为非计算机专业研究生 Python 教学的重要补充。通过充分利用在线资源，可以为学生提供丰富的学习材料和实践机会，拓展其知识视野和实践能力。

1) 在线课程与视频教程

在线课程和视频教程是非计算机专业研究生 Python 教学的重要资源之一[4]。通过选取优质的在线课程和视频教程，可以让学生随时随地学习 Python 编程技能。同时，在线课程和视频教程还能提供丰富的教学案例和练习题，帮助学生巩固所学知识。

2) 在线编程平台与社区

在线编程平台和社区是非计算机专业研究生 Python 教学的重要实践场所之一。通过参与在线编程平台和社区的活动，如编程竞赛、代码审查、技术讨论等，可以让学生与其他编程爱好者交流互动，提升编程实践能力和创新思维。同时，在线编程平台和社区还能为学生提供实时的技术支持和反馈，帮助其解决编程过程中遇到的问题。

3) 在线数据与工具

在线数据和工具是非计算机专业研究生 Python 教学的重要辅助手段之一。通过利用在线数据和工具，如数据分析平台、机器学习库等，可以让学生更加便捷地进行数据分析和机器学习任务。同时，在线数据和工具还能提供丰富的数据资源和算法支持，帮助学生更好地掌握 Python 编程技能在数据分析、人工智能等领域的应用。

3.4. 创建智能评估系统

智能评估系统是一种结合自动化评估工具和人工评估的综合评估系统，可以实现对学生编程能力和工程应用能力的全面评估。在 Python 教学中，我们可以利用智能评估系统，对学生的代码质量、工程应用能力等进行综合评估。智能评估系统可以根据学生的编程表现和工程应用能力，自动生成评估报告和反馈建议，帮助学生及时了解自己的编程水平和工程应用能力，并给出针对性的改进建议。

4. 非计算机专业研究生 Python 教学创新实践案例

为了验证上述创新策略的有效性，本文选取某高校非计算机专业的 Python 课程作为实践案例进行研究。该课程面向机械工程学院的研究生开设，旨在通过 Python 编程教学提升其数据分析和问题解决能力。

4.1. 课程背景与目标

该课程面向机械工程学院的研究生开设，旨在通过 Python 编程教学提升其数据分析和问题解决能力。课程采用项目式学习、跨学科融合、在线资源利用及创建智能评估系统等多元化教学方法，注重培养学生的实际应用能力和创新思维。

4.2. 教学内容与方法

4.2.1. 教学内容

课程内容包括 Python 基础语法、数据类型、控制结构、函数与模块、文件操作、异常处理、数据分析与可视化等。同时，结合机械工程专业的特点和学生实际需求，设计一系列具有挑战性的项目，如机

机械臂的运动控制、机械系统的仿真优化等。学生需要在教师的指导下，完成项目需求分析、系统设计、编程实现、测试调试等全过程，从而在实践中掌握 Python 编程的基本知识和技能。

4.2.2. 教学方法

课程采用项目式学习、跨学科融合、在线资源利用及创建智能评估系统等多元化教学方法相结合的方式进行授课。具体做法如下：

1) 项目式学习：将课程内容划分为若干个项目模块，每个模块围绕一个具体的项目展开。学生以团队形式完成项目任务，通过查阅资料、讨论交流、编程实践等方式解决项目中的问题。教师在项目实施过程中提供必要的指导和支持。例如：在机械臂运动控制项目教学中，我们可以选取一些典型的机械臂运动控制案例，如机械臂的避障运动、协同作业等。通过对案例的深入剖析，学生可以了解机械臂运动控制的实际应用场景和解决方案。同时，我们可以利用虚拟实验环境，模拟机械臂的运动过程和性能表现。通过虚拟实验环境，学生可以直观地观察机械臂的运动轨迹、速度、加速度等参数，深入理解机械臂运动控制的基本原理和方法。再结合实验室的机械臂设备和传感器等设备，设计一系列实景工坊模块，如机械臂的路径规划、运动控制算法验证等。学生在教师的指导下，利用 Python 编程实现对机械臂的控制和数据分析，从而在实践中掌握机械臂运动控制的核心技能。通过这种方式，学生可以更加全面地掌握机械臂运动控制的核心知识和技能，提高编程能力和工程实践能力。

2) 跨学科融合：随着智能制造和工业 4.0 的推进，机械工程领域对具备编程能力和数据分析能力的复合型人才需求日益增加。跨学科融合教学能够拓宽学生的知识视野，提升其综合素质和创新能力，为未来的职业发展奠定坚实基础。在教学中，我们将 Python 编程技能与机械工程学科的知识相结合，设计跨学科案例分析和实践活动。例如：利用在线教学资源，让学生在课前自主学习 Python 编程和机械工程的基础知识，课堂上则侧重于讨论、答疑和实践操作。鼓励学生组建跨学科团队，共同完成项目任务，促进不同领域知识的交流和融合。每年组织学生参加机器人大赛等。通过跨学科融合的教学实施，学生不仅掌握了 Python 编程在机械工程领域的应用技能，还学会了团队协作、项目管理和问题解决等方法。

3) 在线资源利用：充分利用在线课程、视频教程、编程平台与社区等资源，为学生提供丰富的学习材料和实践机会[2]。同时，鼓励和支持学生建立在线学习社区和交流平台，如微信群、QQ 群、论坛等。通过这些平台，学生可以分享学习心得、讨论问题、协作项目，形成良好的学习氛围和合作机制。学生可以利用 Jupyter Notebook 等在线编程环境，进行实时编程演示和代码审查。教师积极参与学生的在线学习过程，提供必要的引导和监督。教师通过定期组织线上讨论、答疑、作业批改等方式，了解学生的学习进度和困难，及时给予帮助和指导。通过开展相关培训和讲座、分享优秀学习资源等方式，提升学生的信息素养和学习技能。通过充分利用在线资源，构建了高效、灵活的 Python 教学体系，提升了学生的编程能力、数据分析能力和创新思维。

4) 创建智能评估系统

a) 代码质量评估

代码质量评估是一种利用自动化工具对代码进行静态分析和动态测试的方法，可以评估代码的规范性、可读性、可维护性等指标。在 Python 教学中，教师利用 Pylint、Flake8 等静态分析工具，对学生的代码进行静态分析，检测代码中的语法错误、逻辑错误、风格问题等；利用 Pytest、Unittest 等动态测试工具，对学生的代码进行动态测试，验证代码的正确性和稳定性。通过代码质量评估，教师可以及时发现学生的编程问题，并给出针对性的改进建议[5]。

b) 工程应用能力评估

工程应用能力评估是一种利用实际工程项目对学生的工程应用能力进行评估的方法。在 Python 教学

中,我们结合机械工程领域的实际需求,设计了一系列具有挑战性的工程项目,如机械系统的故障诊断、优化设计等。学生需要在教师的指导下,完成项目需求分析、系统设计、编程实现、测试调试等全过程,并提交项目报告和代码。教师可以通过审查项目报告和代码,评估学生的工程应用能力,包括需求分析、系统设计、编程实现、测试调试等方面的能力。

4.3. 教学效果

经过一个学期的教学实践,该课程取得了显著的教学效果。具体表现在以下几个方面:

1) 学生编程能力显著提升:通过项目式学习和在线编程平台与社区的实践机会,学生的编程能力得到了显著提升。他们能够熟练运用 Python 进行数据处理和分析任务,并能够将所学知识应用于实际问题解决中。

2) 学生创新能力明显增强:通过跨学科融合教学和在线资源的利用,学生的创新能力得到了明显增强。他们能够结合机械工程学科的知识 and Python 编程技能提出创新性的解决方案和思路。

3) 学生学习兴趣浓厚:通过项目式学习和在线资源的利用,学生的学习兴趣得到了充分激发。他们积极参与课堂活动和实践环节,形成了积极向上的学习氛围。

4) 学生团队协作能力提高:通过项目式学习和跨学科融合教学的实施,学生的团队协作能力得到了提高。他们能够与他人合作完成复杂任务并共同解决问题。

5. 结论与展望

5.1. 结论

本文通过对非计算机专业研究生 Python 教学现状的分析,提出了引入项目式学习、跨学科融合、在线资源利用及创建智能评估系统等多元化教学方法的创新策略。通过具体的教学实践案例验证了这些策略的有效性。研究结果表明,这些创新策略能够激发学生的学习兴趣 and 积极性,提升其编程实践能力和创新思维。同时,这些策略还能够促进非计算机专业研究生与其他学科之间的交流与合作,为其未来的职业发展奠定坚实基础。

5.2. 展望

未来,我们将继续深化非计算机专业研究生 Python 教学的创新与实践工作。一方面,我们将不断优化教学内容和方法,引入更多前沿技术和应用案例,提升课程的针对性和实效性;另一方面,我们将加强与企业和科研机构的合作与交流,为非计算机专业研究生提供更多实践机会和合作平台。同时,我们还将关注非计算机专业研究生的个性化学习需求和发展方向,为其提供个性化的指导和支持。通过持续的努力和创新实践,我们相信非计算机专业研究生 Python 教学将取得更加显著的成效 and 更广阔的发展前景。

参考文献

- [1] 杨小红. 主题研究式 Python 语言程序设计教学改革探索与实践[J]. 创新教育研究, 2023, 11(12): 4071-4075.
- [2] 王丹, 周维斌. 项目式学习在 Python 程序教学的应用研究[J]. 福建电脑, 2021(8): 145-147.
- [3] Yeom, S., Herbert, N. and Ryu, R. (2022) Project-Based Collaborative Learning Enhances Students' Programming Performance. *ITI/CSE'22: Proceedings of the 27th ACM Conference on on Innovation and Technology in Computer Science Education*, 1, 248-254. <https://doi.org/10.1145/3502718.3524779>
- [4] 李松, 张丽平, 丁博. 新工科背景下 Python 程序设计课程教学模式设计与实践[J]. 文山学院学报, 2024(37): 77-87.
- [5] Al-Shanfari, L., Abdullah, S., Fstnassi, T. and Al-Kharusi, S. (2023) Instructors' Perceptions of Intelligent Tutoring Systems and Their Implications for Studying Computer Programming in Omani Higher Education Institutions. *International Journal of Membrane Science and Technology*, 10, 947-967. <https://doi.org/10.15379/ijmst.v10i2.1395>