

工匠精神融入高职《Python程序设计》课程的机制与实施路径探索

吴瑶

金华职业技术大学信息工程学院，浙江金华

DOI: 10.62836/jer.v4n9.1434

摘要：针对高职《Python程序设计》课程中素养目标笼统、思政内容与编程实践衔接不足，以及教学重功能结果、轻质量过程等问题，本文将工匠精神细化为规范、质量、责任、协作、改进五个维度，并对应到可观察的编程行为；据此构建“目标引领—内容映射—项目承载—过程渗透—评价反馈”五环联动机制，以递进式项目、“示范—实践—检查—改进”教学闭环和过程评价贯穿课程实施。教学实践表明，实施班学生在代码规范、边界测试和异常处理方面均有改善，阶段项目和综合项目平均成绩高于对照班。该机制可为高职程序设计课程协同培养知识、能力和职业素养提供参考。

关键词：工匠精神；课程思政；Python程序设计；递进式项目；过程评价

Exploration of the Mechanism and Implementation Path for Integrating Craftsmanship Spirit into Higher Vocational Python Programming Courses

Yao Wu

Information Engineering College, Jinhua University of Vocational Technology, Jinhua, Zhejiang

Abstract: Higher vocational Python programming courses often state broad competency goals while giving insufficient attention to code quality and the development process. This study defines craftsmanship in five dimensions—standards, quality, responsibility, collaboration, and improvement—and translates them into observable programming behaviors. It proposes a five-link model of “objective guidance—content mapping—project-based implementation—process integration—evaluation and feedback,” delivered through progressive projects, a closed teaching loop, and process-based assessment. The intervention class improved in code standardization, boundary testing, and exception handling, and obtained higher average scores than the comparison class in both staged and comprehensive projects. The model offers a practical reference for integrating knowledge, skills, and professional values in higher vocational programming courses.

Keywords: spirit of craftsmanship; curriculum-based ideological and political education; Python programming; progressive projects; process-based assessment

*基金项目：金华市公益性技术应用研究项目(2025-4-043)，2026年度金华职业技术大学教改项目。

作者简介：吴瑶（1993-），女，浙江金华人，硕士，讲师，研究方向：人工智能、数据挖掘、目标跟踪。

1 引言

产业数字化转型和技能型社会建设，对高职学生的技术技能和职业精神提出了同步培养要求。《中华人民共和国职业教育法》将职业道德、专业知识、技术技能和行动能力列为高素质技术技能人才的基本要求[1]；《国家职业教育改革实施方案》提出促进职业技能与职业精神融合[2]；《高等学校课程思政建设指导纲要》要求统一推进价值塑造、知识传授和能力培养[3]。《Python程序设计》中的程序编写、调试、测试和优化，涉及规范、质量、责任、协作和改进，适合将工匠精神落实到具体编程活动中。

但在实际教学中，素养目标往往停留在原则性表述，未与知识模块和项目任务逐项对应；思政内容多靠案例讲解，与编程实践衔接不紧；教学重程序能否运行，轻代码规范、边界测试和异常处理；评价重最终结果，轻学习过程和团队贡献。工匠精神容易停留在认知层面，难以形成稳定的编程习惯。

本文围绕两个问题展开：（1）如何把工匠精神转化为具体编程行为；（2）如何让这些行为要求贯穿课程目标、教学内容、项目任务、教学过程和评价反馈。为此，本文构建多维度的联动机制，并提出配套的实施路径。

2 工匠精神融入课程的机制构建

2.1 工匠精神的课程化转化

职业教育课程思政需要结合专业特点和学生发展需求，将职业精神纳入课程目标、教学内容和实践过程[4]。工匠精神也只有进入专业学习、技能训练和实践任务，才能落实为具体的职业行为[5]。结合程序设计的特点，本文将工匠精神分为规范意识、质量意识、责任意识、协作意识和改进意识五个维度：规范意识体现为遵守代码命名、缩进、注释和结构组织要求；质量意识体现为检查程序逻辑、核对运行结果和开展边界测试；责任意识涉及输入校验、异常处理与数据保护；协作意识要求遵守接口约定、合理分工并共同调试；改进意识

表现为记录问题、修改版本和优化代码结构。

课程思政常见的问题是内容宽泛、融入生硬、评价单一，因此必须把素养要求落到具体教学活动中[6]。程序设计课程还要遵循学生认知和能力发展的节奏，随知识难度与任务要求逐步提高行为标准[7]。教师可据此把五个维度写入项目任务、代码检查和过程评价，并以代码作品、测试记录、调试说明、版本对比和自评互评材料作为观察依据。这样，学生能清楚知道程序完成后还要检查什么、为什么修改，抽象的工匠精神也有了可训练、可检查、可评价的落点。

2.2 五环联动融入机制

程序设计课程的育人改革，需要把课程目标、教学内容、项目实践与评价反馈作为一个整体来设计[8]。据此，本文构建“目标引领—内容映射—项目承载—过程渗透—评价反馈”五环联动机制，如图1所示。

五个环节通过反馈形成闭环，素养目标确定方向，课程内容提供知识基础，项目任务承载实践，教学过程推动行为养成，评价结果用于后续调整。这样，工匠精神成为了学生完成程序设计任务时可遵循的质量标准和行为规范。

3 工匠精神融入课程的实施路径

3.1 重构三维课程目标

课程目标由知识、能力和素养三个层面构成。知识目标主要包括认识Python、基本语法、流程控制语句、组合数据类型、函数、文件操作、异常处理、面向对象等；能力目标聚焦问题分析、程序设计、调试测试和项目开发；素养目标对应规范意识、质量意识、责任意识、协作意识和改进意识，三类目标都在同一教学任务中落实。课程标准、教学设计和项目任务单中的素养目标应采用具体的行为化表述，避免只停留在概括性要求，使课程目标能够落实到教学活动、项目任务和评价标准之中。

3.2 建立课程内容与素养要素的映射关系

各模块的知识特点不同，适合融入的素养内

容也不同。教师应选择与技术任务相关的行为要求。课程模块、主要素养维度和编程行为的对应关系见表1。

这张映射表帮助教师在备课时同时确定教学重点和职业行为要求，再把两者写入代码编写、程序测试和项目实施。对学生而言，规范、质量、责任、协作和改进不是编程之外的附加项，而是判断专业任务完成质量的具体标准。

3.3 以递进式项目强化过程渗透

课程以校园生活和社团服务为情境，把项目分为规范起步、结构优化和责任交付三个阶段。每个阶段都按“示范—实践—检查—改进”推进，使项目难度与行为要求递进提高，如图2所示。学生的关注点由“程序能运行”逐步转向规范实现、测试验证、异常防护和持续优化，相关要求也在反复实

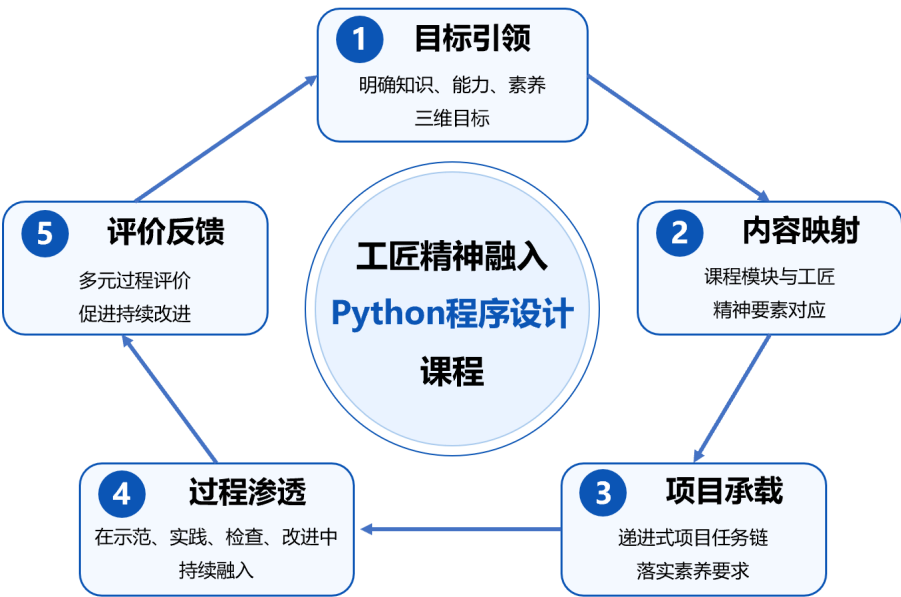


图1. 工匠精神融入《Python程序设计》课程的五环联动机制

表1. 课程模块与工匠精神素养要素映射

课程模块	教学重点	主要素养维度	可观察的编程行为
初识Python	开发环境、程序创建与运行	规范意识、改进意识	按照规范完成环境配置、程序创建和文件保存，主动排查运行错误
基本语法	变量、输入输出、运算符	规范意识、责任意识	使用有意义的变量名，保证数据输入和结果表达准确
流程控制	分支、循环及程序流程	质量意识、改进意识	合理设置判断条件，检查边界情况，排查死循环和逻辑遗漏
组合数据类型	字符串、列表、元组、字典、集合	质量意识、责任意识	根据数据特点选择合适的数据类型，完成数据组织、访问和处理
函数	函数定义、参数、返回值和作用域	规范意识、协作意识	明确函数职责，规范参数和返回值，减少重复代码
文件与异常	文件读写、目录管理、异常处理	责任意识、质量意识	校验输入数据，处理程序异常，保护文件和数据完整性
面向对象	类与对象、封装、继承和多态	协作意识、改进意识	合理分类的职责，提高程序的可扩展性和可维护性

践中转化为编程习惯。

3.4 完善多元全过程评价机制

课程评价同时考查结果和过程，综合学生自评、小组互评和教师评价，并按任务类型调整重点。基础练习侧重代码规范与运行正确性，阶段项目侧重功能完整性、程序结构和测试过程，综合项目考查异常处理、过程改进、团队协作与成果表达。综合项目评价指标及建议权重见表2。

评价权重不是固定的，入门阶段可适当提高功能实现和代码规范的比重；项目复杂度提高后，再增加测试与异常处理、过程改进等维度的比重。评价结果除了计入成绩，还要形成针对性反馈，帮助学生找准问题和改进方向，也为教师调整项目难度、教学活动和评价标准提供依据。

4 典型项目教学案例——“校园社团数据管家”

4.1 明确项目任务与目标

“校园社团数据管家”是文件与异常处理模块

的综合项目。学生需要完成社团名称、负责人、成员人数和活动类型等信息的录入、保存与读取，并处理文件不存在、输入格式错误、字段缺失和保存路径无效等情况。项目要求学生掌握文件读写、路径管理和异常捕获，能够完成数据校验、信息存取与异常处理，在这一过程中重点形成质量意识、责任意识和改进意识。

4.2 组织项目教学实施

项目开始时，学生先完成社团信息的基本读写功能。随后，教师引入文件不存在、成员人数输入非数字、必要字段为空和保存路径无效等故障。学生通过观察程序中断、结果失真或数据受损，认识到程序能运行只是最低要求，完整交付还需要输入校验、异常处理和数据保护。

教师随后比较两个版本：初始版本可以运行；完善版本具备输入校验和异常处理。学生分析两者在稳定性、数据安全和用户体验上的差异，理解异常处理既是保证程序质量的技术要求，也是开发者应承担的责任。

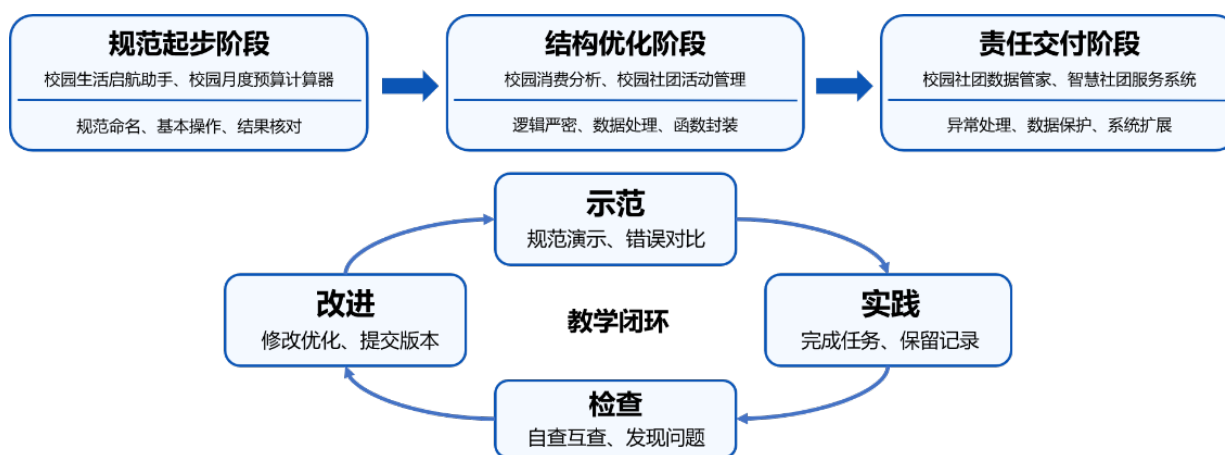


图2. 递进式项目任务链与“示范—实践—检查—改进”教学闭环

表2. 综合项目评价指标及建议权重

评价维度	建议权重	主要观测点
功能实现	35%	需求完成度、运行正确性、结果准确性
代码规范与结构	20%	命名、缩进、注释、重复控制、模块划分
测试与异常处理	20%	正常测试、边界测试、异常处理、数据完整性
过程改进	15%	调试记录、修改说明、版本优化、反思深度
协作与表达	10%	任务分工、沟通配合、项目文档、成果展示

在实践环节，学生补充输入校验、异常提示和文件管理，再按检查清单自查。学生根据检查结果修改程序，提交初始版本、测试记录、修改说明和完善版本。项目依次安排故障情境、版本对比、程序完善、代码检查和修改优化，把质量意识、责任意识和改进意识落实到每一步，具体流程见图3。

4.3 强化项目评价与价值引领

项目评价包括功能实现、代码规范、输入校验、异常处理、测试验证、版本改进和协作表现。初始版本是否完全正确只占一部分，评价更关注学生能否主动识别风险、分析原因并根据反馈有效修改。把数据保护、异常处理和质量改进写入任务与评分标准，促使学生把发现问题和持续改进视为学习成果，逐步形成重规范、重质量、负责任、持续改进的职业习惯。

5 成效评价

5.1 实施对象与评价方法

教学实践于2025-2026学年第二学期开展，实施对象为人工智能工程技术专业2025级1班，共30名学生；对照班为计算机应用技术专业2025级1班，共30名学生。实施班采用“五环联动”机制开展教学，对照班按照原有教学方式完成相同的课程

内容与项目任务，并采用统一评分标准。

本文结合作品分析、阶段比较和班级比较。分析材料包括实施班30名学生的前、后期项目作品及平台记录，主要观察代码规范、边界测试、异常处理、按时提交和版本修改；另对实施班与对照班的阶段项目和综合项目平均成绩作描述性比较。为避免只凭项目成绩判断教学成效，本文从行为、作品、结果和过程四方面组织多源证据，如图4所示。四类证据相互补充，可从不同角度分析学生编程行为和项目质量的变化。

5.2 实施结果与分析

对实施班30名学生的前期项目作品（校园社团活动管理器）、后期项目作品（校园社团数据管家）进行比较后发现，代码规范、边界测试和异常处理三项指标均有提高，如图5所示。代码规范达标率由53.3%升至86.7%，主动开展边界测试的比例由46.7%升至76.7%，异常处理完成率由33.3%升至73.3%。异常处理的增幅最大，提示故障情境和异常处理任务有助于学生关注程序运行风险，增强质量意识与责任意识。

过程材料中，30名学生有28人按时提交项目任务，按时提交率为93.3%；25人提交了较完整的修改版本，占83.3%。多数学生能够按时完成任务，并根据反馈修改程序。图6显示，实施班阶段项目



图3. “校园社团数据管家”项目实施流程示意图



图4. 工匠精神融入课程的多源成效评价证据链

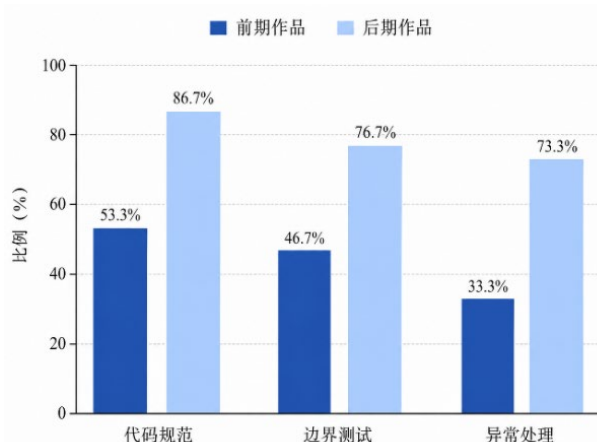


图5. 实施班学生编程行为指标前后变化

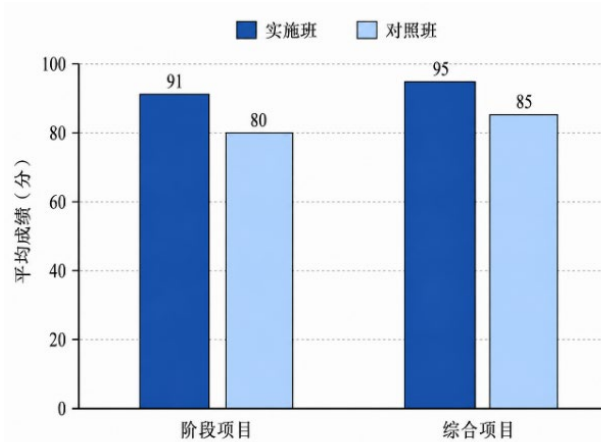


图6. 实施班与对照班项目平均成绩比较

和综合项目平均成绩分别为91分、95分，对照班分别为80分、85分，实施班两个项目的平均成绩均高于对照班。

总体来看，“五环联动”机制实施后，学生的编程行为和项目表现有所改善。但由于班级规模有限，两个班专业不同，评价资料类型也不够丰富，这些结果只能描述教学变化，不能证明班级差异由该机制单独造成。后续研究还需扩大样本，并结合问卷和访谈进一步验证。

6 结语

本文把工匠精神转化为规范、质量、责任、协作和改进五类可观察的编程行为，构建“目标引领—内容映射—项目承载—过程渗透—评价反馈”

五环联动机制，并以递进式项目和过程评价落实到教学。实施班在代码规范、边界测试和异常处理方面均有改善，阶段项目和综合项目平均成绩高于对照班。现有结论仍受班级规模、专业差异和评价资料类型限制，需要在更大范围的教学实践中继续检验。

参考文献

- [1]全国人民代表大会常务委员会.中华人民共和国职业教育法[Z].2022-04-20.
- [2]国务院.国家职业教育改革实施方案[Z].国发〔2019〕4号,2019-01-24.
- [3]教育部.高等学校课程思政建设指导纲要[Z].教高〔2020〕3号,2020-05-28.
- [4]关世春.职业教育课程思政融入路径研究[J].中国职业技术

- 教育,2021(35):44-49.
- [5]刘凤国.工匠精神融入职业教育课程思政建设的理论基础、现实困惑和路径探析[J].辽宁师专学报:社会科学版,2024(4).
- [6]孟子敏,李莉.课程思政教学实践中的若干问题及改进路径[J].中国大学教学,2022(3):51-57.
- [7]白蕾,闫魁颖.阶梯式课程思政在“程序设计基础”中的探索与实践[J].教育教学论坛,2025(49):156-159.
- [8]宋晓霞,武伟,李勇,等.“三位一体”的程序设计基础课程教学实践[J].计算机教育,2025(5):69-74.

