

# 面向深度学习的项目式编程学习优化设计与实践

任永功,林禹竹,多召军

(辽宁师范大学,辽宁 大连 116026)

**【摘 要】**编程课程的深度学习具有一定的领域特殊性,体现为学习者深度应用编程知识解决现实问题的程序设计策略构建与编程自我效能感培养。项目式编程学习不应局限于学习者对于领域知识的理解与配对应用,更需要关注学习者在项目驱动的编程问题解决过程中程序设计策略的构建与迁移。本研究以优化设计项目式编程学习为核心目标,融合编程领域深度学习的特殊性,构建项目式编程学习“四层”过程模型;基于编程方案生成与评估以及程序设计策略构建等过程,梳理编程项目设计原则;结合程序设计支持技术、阶段性结果及启发性问题,结构化地设计促进学习者有效完成程序设计的项目支架。学习者自我感知的程序设计策略水平和编程自我效能感在前测与后测中出现了显著性差异,验证了编程项目设计原则与项目支架设计的有效性。

**【关键词】**深度学习;项目式学习;学习支架;程序设计策略;编程自我效能感

**【中图分类号】**G43 **【文献标识码】**A **【文章编号】**1001-8700(2020)01-0046-08

## 一、问题的提出

编程课程作为计算机相关专业的专业课程,旨在使学生掌握程序设计的基本概念与基本技能,培养学生适应信息社会的解决问题的高阶思维技巧与能力<sup>[1]</sup>。随着新课程教育理念的不断深入,项目式学习作为一种基于建构主义理论的学习模式,以项目的综合性与实践性、自主性与合作性、开放性与创新性为突出特点,以促进学生学习和发展为突出优势被用于编程技术类课程学习。相较于传统教学法,项目式学习对大学生学习成绩和自我效能感提升作用更加显著<sup>[2]</sup>,然而在编程学习过程中,学习者往往因为编程技巧迁移能力不足<sup>[3]</sup>以及编程自我效能感不足<sup>[4]</sup>等而感到困难,使其缺乏程序设计学习的兴趣、自信以及应用复杂概念和技巧来解决编程问题的信念<sup>[5]</sup>。剖析其内在原因,编程项目多是与领域知识高度耦合的配对应用型项目,使学习者仅关注了领域知识的应用规则,忽视了在持续的编程方案生成-评估-实施过程中对于程序设计思维、策略的构建与迁移。

编程课程的深度学习过程具有一定的领域特殊性,其核心内涵体现为:(1)通过项目驱动,使学习者协作探索源于真实情境的编程问题,构建与迁移程序设计策略,逐渐形成应用编程知识与策略解决现实问题的编程思维;(2)在多种引导方式的支持下,增加学习者编程成功的体验,提升学习者的编程自我效能感<sup>[6]</sup>,使学习者敢于应用编程知识与编程思维解决现实问题,进而促进其对于编程知识的深度应用。因此,面向深度学习的项目式编程学习优化设计,应结合编程课程深度学习的领域特殊性,逐层剖析项目式编程学习过程,重构面向深度学习的项目式编程学习过程模型,并据此系统地解析编程项目设计原则,结构化地设计促进学习者有效完成编程项目的学习支架。

## 二、面向深度学习的 PBPL 过程模型构建

项目式编程学习(Project-Based Programming Learning, PBPL)是基于项目式学习模式的编程学习,是学习者在对编程领域知识记忆与简单配对应用的基础上,完成源于真实情境的探索型编程项

**【基金项目】**国家自然科学基金项目“面向异常检测的海量数据流模糊聚类方法研究”(编号:61976109);大连市重点实验室“智慧教育感知与计算”(编号:ZD2019001)。

**【作者简介】**任永功,辽宁师范大学计算机与信息技术学院教授,博士生导师;林禹竹,辽宁师范大学计算机与信息技术学院硕士研究生;多召军(通信作者),辽宁师范大学计算机与信息技术学院,讲师。

目,构建与迁移应用编程领域特殊的策略,提升学习  
者编程自我效能感的过程。本研究采用“剥洋葱”  
的方式,逐层剖析项目式编程学习过程,构建面  
向深度学习的 PBPL 过程模型,如图 1 所示,包括:  
(1)浅表化学习层,主要体现为编程知识理解、记忆  
及简单配对应用;(2)探索型编程项目方案生成与  
评估层,主要体现为学习者在项目式编程学习过  
程中,通过项目需求分析,描述、设计、评估与实施  
项目解决方案,完成较高层次的认知;(3)程序设计策

略构建与迁移层,主要体现为学习者在完成探索型  
项目的过程中,不仅要实现领域知识的迁移应用,  
更强调具有领域特殊性的程序设计策略构建与迁  
移;(4)学习者编程自我效能感形成层。四个层之  
间是逐层深化的关系,内层的学习与认知过程建  
立在外层活动基础之上,内层形成的编程策略、思  
维及编程成功后的信念也将影响外层的学习与编  
程过程。

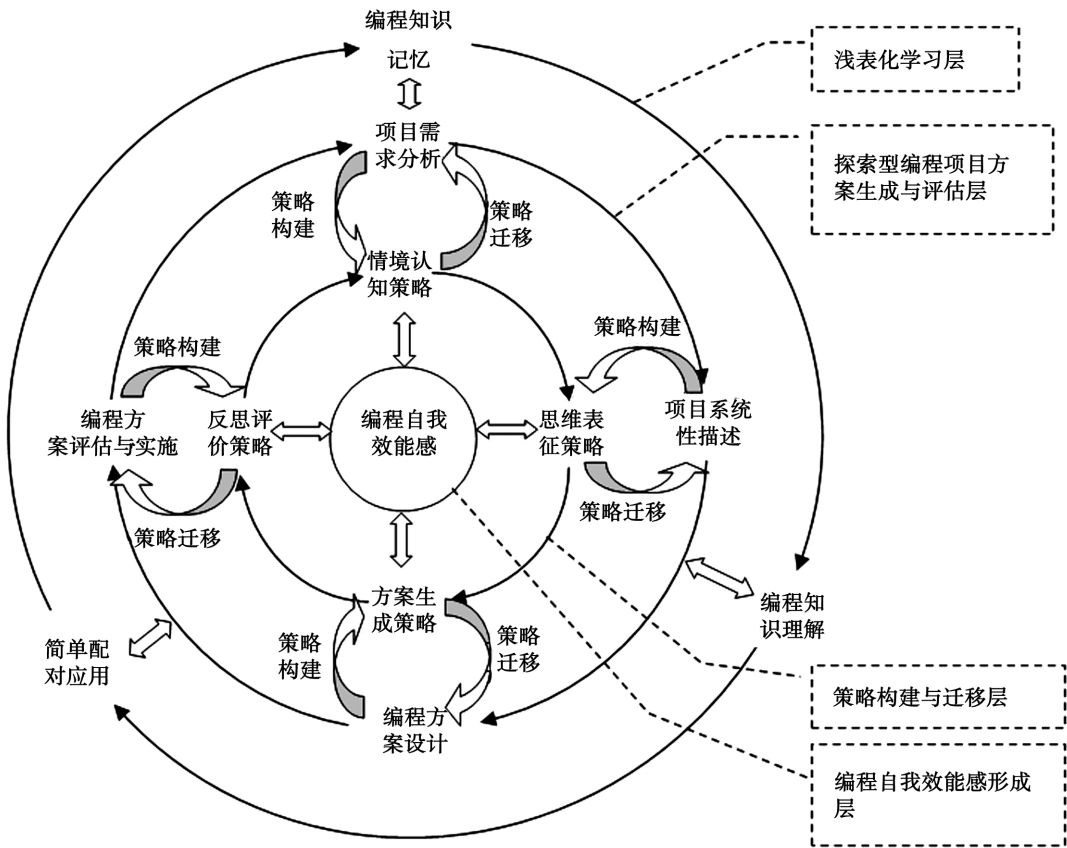


图 1 面向深度学习的 PBPL 过程模型

(一)浅表化学习层

基于布鲁姆的认知目标分类,学习  
者对知识的记忆、理解属于初步的浅层认知,要求学习  
者能够记忆与理解编程基础知识<sup>[7]</sup>。促进学习  
者理解编程领域的规则型知识的简单配对应用过程  
也属于浅表化的学习过程。

(二)探索型编程项目方案生成与评估层

编程项目与编程课程内容之间应该是松耦合  
的,也就是编程项目应该具有开放性、源于真实情  
境和探索型等特征,使学习者在程序设计方案生成  
的过程中体会到逐级递进的步骤累积过程。因此,  
融合软件产品设计流程,将探索型编程项目方案生  
成与评估过程拆分为:(1)项目需求分析;(2)项目  
系统性描述;(3)编程方案设计;(4)编程方案评估

与实施等环节。各个环节之间是递进关系,并且后  
续环节对于前面环节具有一定的反馈指导作用。  
项目需求分析是学习者对项目需求中情境的关键  
信息进行详细地分析,包括项目要求、初始信息、可  
预测的结果及约束条件等信息的准确把握;项目系  
系统性描述是对程序设计中所涉及的专业领域知识、  
实现程序过程时产生的必要环节、完成程序时要达  
到最终效果的集中统一性描述,学习者可借助图示  
形成可视化的表征;编程方案设计是学习者融合领  
域知识将碎片化信息转换为完整方案的过程;编程  
方案评估与实施是学习者对程序设计方案的正确  
性、完整性及可实施性等方面进行自我评估,并在  
集成开发环境下完成程序的编写。

(三) 程序设计策略构建与迁移层

项目式学习中的程序设计策略是学习者在项目驱动的编程问题解决过程中应用的策略性知识,可分为一般性的元认知策略和编程领域内的特殊性策略。以下主要分析学习者为了完成编程项目在程序设计的不同阶段应用的领域特殊性策略。程序设计过程中的领域特殊性策略可依据程序设计方案生成的过程具体划分为:面向需求分析的情境认知策略、面向程序系统性描述的程序设计思维外化表征策略、面向程序设计方案生成的程序设计方案生成策略及面向方案评估与实施的反思评价策略。情境认知策略体现为学习者通过教师提供的虚拟情境感知所学程序设计知识用于解决现实问题的模式、策略等;思维外化表征策略体现为学习者应用可视化图示支持技术系统地描述程序设计过程需要的多个要素及相互之间关系的技巧与方法等;程序设计方案生成策略是指学习者能够把握项目问题的关键特征,按照一定逻辑思维整理已知条件、目标状态、约束条件及领域知识,以形成有效程序设计方案的思维方式与策略;反思评价策略主要是学习者合理利用程序设计方案生成过程中

的阶段认知结果数据,去评价与反思程序设计方案的方法与策略。

(四) 编程自我效能感形成层

自我效能感是个体对自己完成某个目标或任务的能力的自信程度,包括一般自我效能感和特殊领域自我效能感<sup>[6]</sup>,编程自我效能感是学习者认为自己已具备必要的技能和能力并能够很好地完成编程问题的信念<sup>[2]</sup>。编程知识与编程策略的不断积累使学习者获得正向的学习体验,将增强学生对自身编程学习能力的信心。

三、基于 PBPL 过程模型的项目设计原则解析

传统的项目式学习以任务为驱动、侧重项目完成的结果<sup>[8]</sup>,为了促进学习者构建与迁移程序设计策略,应将关注点转向项目实施过程。基于上述 PBPL 过程模型,探索型项目的适配性直接影响学习者的分析、评价与创造的深层次认知过程及程序设计策略构建与迁移过程。以下基于 PBPL 的编程项目方案生成与评估层以及程序设计策略构建与迁移层,采用细化目标反向映射的思路,系统地梳理编程项目设计原则,如图 2 所示。

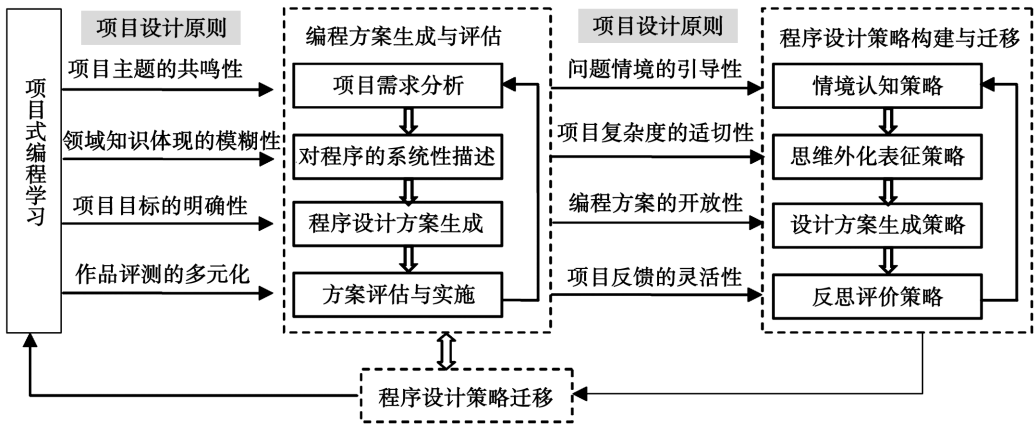


图 2 编程项目设计原则

(一) 映射“编程方案生成与评估”目标的项目设计原则

1. 项目主题的共鸣性

需求分析是开启程序设计的第一步,除了能使学习者准确把握完成项目的关键信息,更应该让学习者感受到完成项目所带来的意义与价值,而项目的环境是促使学习者引起共鸣、深度理解主题的重要因素。Hsu 等曾建立了一个问题的学习环境,帮助学生理解编程概念<sup>[9]</sup>。因此,编程项目的主题应发生在具体的真实世界中,是学习者可能接触到的事物或无法解决的问题,通过相似的情况引发学习

者产生共鸣,从而引发编程动机和编程兴趣。

2. 领域知识体现的模糊性

Kurubacak 研究表明,基于项目的网络学习能够提高学生的批判思维能力<sup>[10]</sup>,有利于学生很好地领会、分析与运用知识,同时也使学生认知结构中知识的逻辑关系更加清晰。项目的系统性描述是基于专业领域知识的,对程序设计中所涉及的所有要素进行集中统一性描述。因此,在设计编程项目时,对于完成项目需要的专业领域知识的体现应具有一定的模糊性。学习者在对专业知识选择的过程中,可以将已知的知识作为中介,通过已知知识

与新知识之间的交互,实现认知结构重组,促进学习者对于领域知识的深度应用。

### 3. 项目目标的明确性

方案的形成过程应是同伴之间相互协作,提取程序设计中的关键信息,并准确应用领域知识、形成编程问题解决方案的过程。编程项目设计应体现出明确的项目目标,从而引导学习者不断修正、持续完善,最终形成自我满意的智慧产品的过程。不断完善编程方案以达到项目目标的过程,也会逐步提升学习者成功编程的信念。

### 4. 作品评测的多元化

为促进学习者评估结果的有效性,评估维度应趋于多元化,不仅评测学习者的最终程序,也应该评价学习者在程序设计过程中的需求分析结果、流程图形式的系统性描述结果以及融合领域知识的程序设计方案等;评估方式应采用多样化手段,如教师评价、组内互评等,灵活的反馈信息使学生在完成编程作品时,能够结合自己的问题改进不足以形成正确的程序设计思维。

## (二)映射“程序设计策略构建与迁移”目标的项目设计原则

### 1. 问题情境的引导性

现实生活中的编程项目的初始条件、目标状态、约束条件等要素是模糊的,需要学习者具备精准提取关键信息的策略与技能。因此,在项目设计时,问题情境不仅要呈现项目需求,也需要具有一定的引导性,以辅助学习者选择与应用其认知结构中恰当的需求分析策略和模式。例如,通过动画、故事视频等资源呈现问题情境,通过表格、图示等工具引导学生选择合适的策略完成项目需求分析。

### 2. 项目复杂度的适切性

项目的系统性描述是学习者基于合适的领域知识,梳理已知条件、目标状态、约束条件等要素之间关系的过程。源于真实情境的探索型项目的实施需要融合多个知识点,涉及知识点的数量越大,项目复杂度就越高,要求学习者问题解决思维结构化表征的能力就越高。当问题解决者能够正确画出问题结构的图示(程序设计的流程图)时,问题的解决绩效将得到改进<sup>[11]</sup>。因此,符合学习者认知水平的编程项目,能够激励学习者应用可视化图示支持技术,系统地描述多个要素之间的相互关系,形成思维外化表征策略。

### 3. 编程方案的开放性

程序方案生成过程是学习者提取项目关键特征并按照一定逻辑思维整理已知条件、目标状态、

约束条件及领域知识的过程,这一过程需要学习者不断调试、反复设计与持续完善项目,学习者在不断修改的过程中也改变了思维方式,增强了策略构建能力。因此,程序设计方案应具有一定的开放性,使学习者在多个备选方案的甄别过程中,体会提取程序设计的关键信息、结构化梳理多种信息与条件、调试与完善程序的方法与策略。

### 4. 项目反馈的灵活性

在编程项目完成过程中,学习者对于需求分析结果、系统性表征结果以及程序设计方案的评价与反思活动是灵活的,并且要求学习者采用恰当的方法与策略。因此,应合理利用程序设计方案生成过程中的阶段性认知结果数据,并在此基础上设计提示性问题,引导学习者自主构建程序设计方案反思与评价的方法与策略。

## 四、基于 PBPL 过程模型的项目支架设计

支架是在线环境下项目化学习的重要干预手段<sup>[12]</sup>,为促进学习者高效地完成编程项目,构建程序设计策略,获得成功的编程体验,需要在项目设计与实施过程中,适时地为学习者提供合适的技术支持、阶段性结果与启发性问题等项目支架。本研究主要从编程方案设计与评估、程序设计策略构建及程序设计策略迁移等三个维度,结构化地设计编程项目支架,程序设计策略构建及程序设计策略迁移维度的项目支架应建立在编程方案设计与评估的基础上,每个维度各个阶段的项目支架间应是密切联系的。

### (一)“编程方案设计与评估”维度的项目支架

“编程方案设计与评估”维度的项目支架,是在“项目需求分析”“项目系统性描述”“编程方案生成”“方案评估与实施”过程中,将高交互性的要素分解,降低其复杂程度,使其成为多个不同层级的认知图式要素<sup>[8]</sup>。“项目需求分析”阶段的情境支持技术主要是辅助学习者准确分析真实情境中的关键信息,例如,为学习者提供含有表头信息和简单案例的需求分析表格,学习者可以结合案例和表头信息的关键词完成编程项目的需求分析。“项目系统性描述”阶段的结构化表征技术主要是辅助学习者将其内在的较为混乱的信息,按照程序的逻辑结构重新梳理并进行可视化表征。例如,为学习者提供散乱的带有部分关键信息的流程图组件,要求学习者按照自己的思路将散乱的组件合成为完成的流程图。“编程方案生成”阶段的领域知识支架主要是辅助学习者结合领域知识,将程序流程图转

译为程序代码的程序设计方案,该阶段学习支架主要针对编程基础知识的应用规则等方面为学习者提供支持。“方案评估与实施”阶段的学习支架主要是辅助学习者对前面几个阶段形成的程序设计方案的可行性及完整性等内容进行评估,该阶段的学习支架应以为学习者提供评估的参考模板或方案为主,如教师提供的方案和同伴的程序设计方案等参考数据支持。

(二)“程序设计策略构建”维度的项目支架

程序设计策略构建是建立在编程方案设计、实施与评估基础上的学习者内在的认知过程,程序设计策略构建维度的项目支架主要结合学习者的阶段性结果数据,在每个环节中设计合适的启发性问题,引导学习者对编程问题解决过程进行反思、评价与归因,确保学习者在合理的支架下更好地完成活动。该维度的项目支架设计主要包括:促进情境认知策略构建的需求分析结果与启发性问题,促进思维外化表征策略构建的系统性描述结果与启发性问题,促进编程方案生成策略构建的程序设计方案与启发性问题,促进反思与评价策略构建的多重结果融合后的数据与启发性问题。

(三)“程序设计策略迁移”维度的项目支架

程序设计策略迁移发生的关键是学习者能将新的编程问题的情境、程序设计思路与已完成的某个编程问题建立联系。“程序设计策略迁移”维度

的项目支架的主要作用是引导学习者积极寻找认知结构中与新项目相关的问题,并进行合理的领域知识比较,发现两个学习项目之间的相同规则,实现编程思路及技巧的迁移。

五、教学实践案例

(一)教学对象与教学内容

本研究以某师范大学教育技术学专业本科一年级的 29 名学生为教学对象,以大学生编程学习的入门课程“C 语言程序设计”为教学内容,要求学生在掌握程序设计基本概念的基础上,建立正确的程序设计思想,培养对于编程课程学习的兴趣和积极的动机。

(二)教学活动

本研究以团队自主研发的移动网络学习社区<sup>[13]</sup>作为技术平台开展教学活动,选取“循环结构设计”为主要内容开展为期一个月的教学实验。下面以“for 循环程序设计”为例,阐述编程项目的设计与引导活动的设计。为促进学习者建构与迁移程序设计策略,要求学习者完成两个探索型项目,如表 1 所示,通过完成“项目一”,使学习者深度应用核心知识点,构建不同阶段的程序设计策略;“项目二”的难度有所提升,旨在促进学习者迁移应用通过“项目一”构建的程序设计策略。

表 1 “for 循环”编程项目与引导活动设计

项目描述	主要教学活动		设计意义
项目一： “xz 同学在参加教育实习时，为了更好地了解班级情况，决定在 40 名同学中抽取 15 名进行访谈，请大家帮助 xz 同学随机生成 15 个学号，并保存在数组里。”	项目需求分析	①查看项目情境,填写项目需求分析表格,初步确定项目目标(随机生成 15 个 1~40 之间的不相同的整数)、初始信息(生成数范围:1~40 之间,生成数个数:15 个数)、约束条件(15 次随机生成的数不可以相同); ②分享需求分析表格,查看与完善同伴的表格; ③结合同伴分享的需求分析结果和自己的需求分析结果,反思同伴协同完善需求分析表格的过程,认真回答“你是如何加工项目描述信息并完成需求分析表格的?有什么特别之处?”	①描述项目情境时强调几个数字,引导学习者准确填写需求分析表格; ②通过真实的问题情境,激发学习者对编程项目学习的兴趣; ③通过图表引导,使学习者了解需求分析阶段的主要工作,以及在启发性问题的引导下分析有效完成项目的技巧与策略。
	项目系统性描述	①基于上述的需求分析结果,初步确定完成项目需要的专业领域知识(随机生成整数的方法、循环语句等); ②批判性地使用专业领域知识,对比 while,do...while 及 for 循环语句结构的特点,发现本案例应用 for 语句更简单; ③结合专业知识,初步确定完成项目的步骤,并画出流程图;分享流程图,查看与完善同伴的流程图; ④结合最终的流程图,反思系统性描述的过程,并回答“你是如何对比分析 while,do...while 及 for 的区别?画流程图的过程中,如何确定判断框、执行框等组件的先后顺序?”	①项目情境中对于专业领域知识的体现较为模糊,使学习者批判性地使用专业知识,清晰知识的逻辑关系; ②通过复杂度略高的项目使学习者认识到通过画流程图梳理思路及组织相关问题完成步骤的方法与优势; ③在启发性问题的引导下,思考使用流程图外化表征程序设计思维的方法与技巧。

项目描述	主要教学活动		设计意义
	编程方案设计	①结合专业知识(for 循环语句、if 条件语句等),将流程图中的步骤转换为程序代码形成程序流程图; ②分享程序流程图,在同伴之间协作优化流程图的过程中,解决程序方案中复杂问题(为了避免随机生成数的重复,需要嵌套 for 循环语句,遍历已生成数的集合,并通过 if 语句判断是否重复),小组间相互探讨,不断修改,思路汇总,找到程序的修改方案,有效地解决问题; ③结合程序流程图,反思程序设计方案生成过程,并回答“如何把握关键信息将流程图转译为程序流程图?”	①使学生进一步认识 for 循环语句的用途(循环多次执行相同语句、遍历数据集); ②在明确的项目目标引导下,学习者相互协作,持续完善程序设计方案; ③在启发性问题引导下,使学习者在协作优化程序方案的过程中体会程序设计方案生成的策略。
	评估与实施	①结合教师展示的标准程序方案及评价标准,对程序设计方案进行自我评估及同伴评估; ②组织信息,撰写与检查程序代码; ③回顾完成项目过程,及时反思教师的评阅意见和同伴的有效建议,对产生的问题内部归因。	①多维度的评估信息引导学习者对程序设计方案的可性性与正确性进行评估; ②及时、灵活反馈给学习者使其进行自我反思与评估。
项目二:“xz 同学兼职图书管理员,现有通过扫码录入书号的 76 本书,为方便将书放回书架需要将书号进行排序,请大家为他设计一个排序程序。”	①对比两个项目的项目目标,思考如何应用项目一的知识解决项目二的问题,发现项目一是数的循环执行和数据集遍历,项目二则是在项目一的基础上进行数据的交换排序; ②对比项目一和项目二的项目情境,结合项目一需求分析过程完成项目二的需求分析表格; ③找出相关联的部分,项目情境虽然不同,但都是循环执行相同代码的问题,可在项目一的基础上完成项目二; ④根据项目要求,小组协作完成项目二的方案设计; ⑤学生评价小组成员作品后发现问题并提出疑问,通过教师解答,发现编写思路的不同和解决方法的不同,了解编程方式是多种多样的,知识点之间存在必然联系。		①项目二复杂度略高于项目一,使学习者在掌握使用 for 循环实现数据集遍历的基础上,学会使用 for 循环语句实现交换排序; ②项目二是对项目一的拓展与延伸,引导学习者对比完成两个编程项目需要的问题空间,促进迁移发生; ③项目方案具有一定的开放性,使学习者找到两个项目之间的相同规则,实现编程思路、图示及技巧的迁移。

（三）数据收集

1. 学习者感知的程序设计策略构建水平调查量表设计

为测量大学生自我感知的程序设计策略水平,本研究从情境认知策略水平、思维外化表征策略水平、程序设计方案生成策略水平和反思评价策略水平等四个维度设计 20 个题项,每个题项采用 1 ~ 5 点(非常不符合 ~ 非常符合)评分。实验前测与实验后测的总体内部一致性信度 Cronbach ’ s α 为 0.915。

2. 学习者程序设计自我效能感水平调查问卷设计

本研究改编 Kong 等人设计的编程效能感(Programming Self – efficacy) 调查问卷和创意自我效能感(Creative Self – efficacy) 调查问卷<sup>[14]</sup>,构建学习者程序设计自我效能感水平调查量表,测量学习者程序设计自我效能感水平。量表共设计了 9 个题

项,每个题项采用 1 ~ 5 点(非常不符合 ~ 非常符合)评分。实验前测与实验后测的总体内部一致性信度 Cronbach ’ s α 为 0.863。

（四）数据分析与讨论

采用问卷调查的方法对学习者在教学实验前后的自我感知程序设计策略构建水平和学习者编程自我效能感水平两个方面的数据进行分析与讨论。

1. 学习者自我感知程序设计策略构建水平分析

本研究采用学习者感知的程序设计策略构建水平调查量表测量实验前后学习者感知的程序设计策略构建水平,并对前后测数据进行配对 T 检验,以分析面向深度学习的项目式编程学习优化设计对学习自我感知的程序设计策略构建的影响效果。结果如表 2 所示,学习者自我感知的程序设计策略构建的总体水平,在基于项目优化设计开展学习活动的前测与后测中出现了显著性差异(T =

- 10.79,  $P < 0.01$ )。表明针对项目式编程学习的阶段性特征设计的相关项目,能够促进学习者构建与迁移程序设计策略。

“情境认知策略”维度在基于项目优化设计开展学习活动的前测与后测中出现了显著性差异( $T = -15.98, P < 0.01$ ),表明:(1)在引起学生共鸣的真实情境项目主题和相关表格与图示的引导下,学习者能够有效地完成程序需求分析;(2)阶段性结果数据和启发性问题,对于引导学习者构建真实情境中关键信息的提取与加工技巧及策略有积极的促进作用。

“思维外化表征策略”维度( $T = -6.68, P < 0.01$ )和“设计方案生成策略”维度( $-10.91 = -$

10.91,  $P < 0.01$ )得分均值在前测和后测中存在极显著性差异,表明:(1)目标明确、复杂度略高的项目,能够激发学习者持续完善项目解决方案;(2)项目描述中较模糊的专业知识体现和相对开放的项目方案设计,结合启发性问题,能够引导学习者反复运用绘制流程图工具梳理程序设计思维与完成程序设计方案,并积极体会与掌握基于协同意义建构工具的协作设计程序设计方案的技巧与策略。

“反思评价策略”维度( $T = -9.23, P < 0.01$ )的前后测数据存在显著性差异,说明多元化评估与灵活的反馈对学习者的反思项目完成过程及项目方案自我评价起到一定的促进作用。

表 2 学习者感知的程序设计策略构建水平分析结果

策略构建维度	变量	均值	标准差	自由度	T 值	显著性
总体水平	前测	3.22	.29	19	-10.79	0.002
	后测	4.21	.16			
情境认知策略	前测	2.99	.28	4	-15.98	0.000
	后测	4.19	.26			
思维外化表征策略	前测	3.48	.21	4	-6.68	0.003
	后测	4.25	.15			
设计方案生成策略	前测	3.48	.17	4	-10.91	0.000
	后测	4.40	.05			
反思评价策略	前测	2.95	.22	4	-9.23	0.001
	后测	4.01	.18			

2. 学习者编程自我效能感水平分析

本研究采用学习者编程自我效能感调查量表测量实验前后学习者程序设计自我效能感水平,并对前后测数据进行配对 T 检验,以分析面向深度学习的项目式编程学习优化设计对学习者的程序设计自我效能感的影响效果。结果如表 3 所示,学习者编程自我效能感水平,在基于项目设计的学习活动的前测与后测中出现了显著性差异( $T = -7.84, P < 0.01$ ),说明特征清晰的项目能够激发学习者的程序设计兴趣与动机,并持续优化与完善程序设计方案,这个过程也增加了学习者成功编程的信念,表明成功地干预计算机编程学习能提高学生的编

程自我效能感<sup>[15]</sup>。  
创意自我效能是学习者相信自己能够产生新颖的想法与解决方法<sup>[16]</sup>。前后测数据进行配对 T 检验结果如表 3 所示,“创意自我效能感”维度得分均值在前测和后测中存在极显著性差异( $T = -7.35, P < 0.01$ ),说明“明确的编程目标”、“开放的程序设计方案”及“符合学习者最近发展区的项目难度”设计,能够引导学习者不断优化与完善程序设计思路与程序设计方案,这个过程也提升了学习者产生新颖想法与解决问题的自信心,即提升了学习者的创意自我效能感。

表 3 学习者编程自我效能感分析结果

	变量	均值	标准差	自由度	T 值	显著性
编程自我效能感	前测	2.26	.32	4	-7.84	0.001
	后测	3.23	.54			
创意自我效能感	前测	3.28	.57	3	-7.35	0.005
	前测	3.28	.57			

## 六、结语

项目式学习以项目的综合性与实践性、开放性与创新性等突出特点,被用于编程技术类学习以促进学生学习和发展。然而,在项目式编程学习中,忽视了在持续的编程方案生成-评估-实施过程中对于程序设计思维、策略的构建与迁移。本研究将促进学习者构建编程策略和培养学习者的编程自我效能感作为编程特殊领域完成深度学习的目标,深入分析项目式编程学习过程,梳理深度编程学习过程反向映射的项目设计原则,并融合编程支持技术与引导活动设计项目支架。分析教学实践前后学习者自我感知的编程策略构建效果和编程自我效能感数据,结果出现显著性差异,说明研究思路与设计的有效性。后续将通过学习行为数据观察学习者在项目式编程学习过程中的行为与心理变化,开展进一步的微观研究。

### 【参考文献】

[1]Chen G,Shen J,Barth - cohen L, et al. Assessing Elementary Students' Computational Thinking in Everyday Reasoning and Robotics Programming[J]. Computers & Education, 2017, 109: 162 - 175.

[2]Frymier A B, Shulman G, Houser M. The Development of a Learner Empowerment Measure[J]. Communication Education, 1996, 45(3): 181 - 199.

[3]Chao P. Exploring Students' Computational Practice, Design and Performance of Problem - solving Through a Visual Programming Environment[J]. Computers & Education, 2016, 95: 202 - 215.

[4]Sáez - lópez J,Román - gonzález M,Vázquez - cano E. Visual Programming Languages Integrated Across the Curriculum in Elementary School: a Two Year Case Study Using "scratch" in Five Schools[J].

Computers & Education, 2016, 97: 129 - 141.

[5]Yukselturk E,Altioek S. An Investigation of the Effects of Programming with Scratch on the Preservice It Teachers' Self - efficacy Perceptions and Attitudes Towards Computer Programming[J]. British Journal of Educational Technology, 2017, 48(3): 789 - 801.

[6]朱阁,卢守楠,郑加丽. 计算机编程类课程持续学习意图模型与实证研究[J]. 高等工程教育研究,2019(04):194 - 200.

[7]何克抗. 深度学习:网络时代学习方式的变革[J]. 教育研究,2018,39(05):111 - 115.

[8]申静洁,赵呈领,周端云. 培养学生创新能力:基于项目学习理论的创客课程设计研究[J]. 现代远距离教育,2019(02):43 - 51.

[9]Hsu T C,Hwang G J. Effects of a Structured Resource - based WebIssue - Quest Approach on Students' Learning Performances in Computer Programming Courses [J]. Educational Technology & Society , 2017(3): 82 - 94.

[10]Kurubacak G. Building Knowledge Networks through Project - Based Online Learning: A Study of Developing Critical Thinking Skills-Via Reusable Learning Objects [J]. Computers in Human Behavior, 2007, 23(6): 2668 ~2695.

[11]多召军,赵蔚,李玉斌,等. 问题解决学习视角下基于网络学习空间的混合式学习设计[J]. 电化教育研究, 2018, 39(2): 32 - 38.

[12]李梅. 在线环境下项目化学习支架探究[J]. 现代远距离教育, 2019 (01): 3 - 9.

[13]多召军,赵蔚,任永功. 移动网络学习社区构建新范式:大学生自我调节学习效能感培养视角[J]. 现代远距离教育, 2019 (01): 10 - 17.

[14]Kong S, Chiu MM, Lai M. A Study of Primary School Students' Interest, Collaboration Attitude, and Programming Empowerment in Computational Thinking Education[J]. Computers & Education, 2018, 127(2018):178 - 179.

[15]Denner J,Werner L,Campe S, et al. Pair Programming: Under What Conditions Is It Advantageous for Middle School Students? [J]. Journal of Research on Technology in Education, 2014, 46(3): 277 - 296.

[16]Brennan Karen. New frameworks for studying and assessing the development of computational thinking[C]. 2012 Annual meeting of the American educational research association (AERA'12), Canada.

## Optimization Design and Practice of Project - based Programming Learning for Deep Learning

REN Yonggong, LIN Yuzhu, DUO Zhaojun

(School of Computer and Information Technology, Liaoning Normal University, Dalian Liaoning 116000)

**Abstract:** Programming courses based on deep learning have certain domain specificity. It is embodied in the deeper application of programming knowledge to solve the real problem of programming strategy construction and programming self - efficacy. Project - based programming learning should not be limited to learners' understanding and matching application of domain knowledge, but also need to pay attention to learners' programming strategy construction and migration in project - driven programming problem solving process. First, the "four layers" process model of project - based programming learning was designed to optimize design PBPL, combining the particularity of deep learning in the field of programming. Next we traced programming project design principles from the dimensions of "programming plan generation and evaluation" and "programming strategy construction and migration". Finally we designed the bracket model to facilitate the learners to effectively complete the programming project combining the programming support technology, the phased results and the heuristic issues, and structurally. The results of practical research show that the learner's self - perceived programming strategy level and programming self - efficacy have significant differences in pre - test and post - test, which verifies the effectiveness of programming project design principles and project scaffold design.

**Key words:** Deep Learning; Project - based Learning; Programming Strategy; Programming Self - efficacy