Pedagogical Research
2024, 9(3), em0218
e-ISSN: 2468-4929
https://www.pedagogicalresearch.com

MODESTUM

Research Article **OPEN ACCESS**

Blended pedagogy for computer programming language

Ling Zhang [1*]

[1] School of Software & Internet of Things Engineering, Jiangxi University of Finance and Economics, Nanchang, CHINA
*Corresponding Author: zhangling@jxufe.edu.cn

Citation: Zhang, L. (2024). Blended pedagogy for computer programming language. *Pedagogical Research, 9*(3), em0218. https://doi.org/10.29333/pr/14744

ARTICLE INFO

Received: 18 Mar. 2024
Accepted: 09 Jun. 2024

ABSTRACT

In the face of the challenges posed by the COVID-19 pandemic, the hybrid teaching model has garnered significant attention for its combination of the depth of traditional education with the convenience of distance learning. Focusing on the domain of computer programming language instruction, this study innovatively designs a hybrid teaching strategy aimed at fully exploiting the flexibility of its teaching design and the variety of pedagogical approaches. The strategy integrates face-to-face teaching with online autonomous learning, incorporating project-based teaching methodologies and immediate feedback mechanisms to facilitate active student engagement and deep learning. Through a year-long practice in a C++ programming course, encompassing 68 students, the study empirically validates the effectiveness of the hybrid teaching approach. It not only demonstrates remarkable educational outcomes, enhancing the quality of programming instruction and student satisfaction with their learning experience, but also employs Bayesian analysis to delve into the relationship between learning trajectories and students' sense of self-efficacy. By focusing on key indicators during the learning process, such as the timeliness and quality of online learning, laboratory work, and project assignments, the study then utilizes Bayesian models to directly assess the impact of these learning behavior metrics on students' perceived self-efficacy. The findings reveal that students with outstanding academic achievements exhibit higher levels of self-efficacy, confirming that academic performance can reasonably reflect teaching effectiveness and provide a quantifiable basis for assessing individual learning progress. Consequently, this research not only contributes a novel strategy to computer programming education practice but also offers a valuable reference for the application of hybrid teaching models in other disciplines. Furthermore, it promotes in-depth contemplation on post-pandemic innovations in teaching modes and issues of educational equity, laying a solid foundation for constructing a more adaptive and inclusive future education system.

Keywords: hybrid teaching model, project-based learning approach, instant feedback mechanism, sense of self-efficacy

INTRODUCTION

With the outbreak of the COVID-19 pandemic, global education systems have confronted significant challenges, necessitating a swift shift towards remote education in response to the disruption of in-person teaching (Takona, 2023). In China, online instruction has been widely adopted as an emergency measure, with educators resorting to recording lectures and leveraging open platforms to innovate teaching methodologies, aiming to curb the spread of the virus. However, in this new realm of online education, the majority of teachers find themselves inadequately equipped to effectively manage and organize teaching activities, while learners grapple with communication barriers, a lack of independent problem-solving skills, and insufficient computer literacy (Kizilcec & Schneider, 2015). Although online education requires a continuous process of improvement and adaptation, reliance solely on virtual teaching often falls short of achieving optimal educational outcomes.

Before the pandemic, the merits and demerits of various teaching methods had been extensively examined. Project-based instruction (Ma et al., 2014), with its close alignment to project characteristics, has demonstrated remarkable advantages in cultivating students' entrepreneurial skills. The integration of problem-based learning with traditional teacher-directed curricula has effectively enhanced students' self-confidence, motivation, and practical skills (Brake et al., 2018; Lydia et al., 2024). Conversely, overreliance on teacher-centered approaches often overemphasizing teacher-directed action, neglecting individual student differences and learning efficiency, which can lead to less than satisfactory educational outcomes (Cubric, 2008; Tarimo, 2016). In contrast, student-centered teaching methodologies, which emphasize autonomous learning and encourage independent exploration and knowledge construction, serve to stimulate students' potential (Davidovitch, 2013; Palmer-Abbs et al., 2021).

Post-pandemic, undergraduate programming education confronts shifts in student needs and habits, advocating a transition towards learner-centered pedagogies (Chin & Kozimor, 2023; Sharma et al., 2020). Traditional face-to-face teaching, constrained by the pandemic, has given way to remote learning, which, while flexible, lacks adequate interaction and emotional connectivity, with mere praise feedback proving insufficiently effective (Parks, 2021; Wang et al., 2021). Amidst this context, hybrid teaching models have garnered attention for combining the real-time engagement of in-person instruction with the autonomy of remote learning, offering a multiplicity of teaching strategies (such as diversified feedback) and ensuring continuity of learning amidst the pandemic (Awotunde et al., 2023; Krogh et al., 2009).

By accommodating student needs and transcending the limitations of singular teaching modes, hybrid teaching emerges as an ideal approach to enhance the efficacy of programming education, providing insights for innovation and equity in education (Omona & O'dama, 2024). While the significance of hybrid pedagogical approaches is widely acknowledged, there remains a dearth of research on their specific application within the technically demanding and practice-oriented domain of computer programming language instruction.

This study aims to address this research gap by designing and validating an innovative hybrid teaching model that showcases superiority in instructional design flexibility and diversity of teaching strategies. The model ingeniously integrates instructor-led offline classroom interactions with student-driven online learning environments, incorporating project-based learning and positive feedback mechanisms to effectively stimulate students' proactive inquiry spirit and deep engagement. Furthermore, we conducted a thorough analysis of learning outcomes and introduced Bayesian methods to examine students' self-efficacy, revealing that our teaching method not only achieved favourable instructional outcomes but also aligned with heightened student self-efficacy, thereby further enhancing learning enthusiasm. This research not only provides practical new methods and optimization strategies for programming instruction but also offers valuable references for other disciplines exploring hybrid teaching models. More importantly, it fuels deeper contemplation within the educational community regarding instructional model innovations and issues of educational equity in the new normal post-pandemic era, underpinning the development of a more resilient and inclusive future education system.

THEORETICAL FRAMEWORK & RESEARCH HYPOTHESES

Existing Research on Programming Language Instruction

In recent years, research on programming language instruction has witnessed significant advancements, particularly concerning teaching methodologies, educational resources, and assessment approaches. Researchers have actively sought teaching models conducive to programming learning, including project-based learning (Kokotsaki et al., 2016), flipped classrooms (Akcayir & Akcayir, 2018; DeLozier & Rhodes, 2017; Takona, 2023), and collaborative learning (Li & Xing, 2020; O'Donnell & Hmelo-Silver, 2013), all aimed at enhancing students' programming skills, logical thinking, and problem-solving abilities. Concurrently, the proliferation of online programming platforms and educational resources has broadened the horizons and possibilities for teaching programming. Moreover, the evaluation of teaching effectiveness in programming has increasingly shifted focus from mere grades to the learning process, emphasizing the measurement of deeper learning outcomes like coding proficiency, algorithm comprehension, and programming mindset (Aivaloglou & Hermans, 2016). In addition, a case study showed that gamified learning in introductory programming enhances student behavior and academic performance significantly (Papadakis & Kalogiannakis, 2019). Following this, Papadakis (2020) research illustrated the efficacy of game development in boosting secondary students' programming enthusiasm and comprehension. Together, these studies highlight gamified teaching's great potential to motivate and enhance capabilities in programming education.

Student-Centered vs. Teacher-Centered Instructional Approaches

Student-centred instructional approaches emphasize the primary position of the student, advocating for self-directed learning, cooperative exploration, and other strategies to ignite intrinsic motivation, develop critical thinking, and nurture creativity (Freire, 1970). Teachers in this model adopt roles as facilitators and advisors, furnishing students with essential support and feedback. On the other hand, teacher-centered methodologies place the teacher at the forefront, underscoring the structured delivery of knowledge and stringent practice, to assure mastery of predetermined learning content (Bruner, 1961). Despite the distinct focuses of these two paradigms, modern educational theories tend towards integrating elements of both, seeking equilibrium that adapts dynamically to the particular educational setting and learners' requirements (Dewey, 1938; Shulman, 1986).

Hybrid Teaching Methods

As a pivotal innovation in the field of education, the evolution of hybrid teaching methodologies mirrors educators' relentless pursuit of optimizing instructional effectiveness and catering to the diverse learning needs of students. Initially, hybrid teaching primarily manifested as a straightforward combination of conventional in-person instruction with distance learning, aiming to address the limitations of single-mode teaching in terms of time, space, and resources (Smith & Fernandez, 2017). Over time, theoretical and practical explorations deepened, leading hybrid teaching to emphasize the dynamism, interactivity, and personalization of the teaching process. By integrating a variety of teaching media, technologies, and strategies, the approach aims to create an efficient learning environment that supports autonomous learning while also providing timely guidance (Johnson, 2013). In recent years, with the advent of advanced technologies such as big data and artificial intelligence, hybrid teaching has embraced a new trend of intelligence and personalization, exemplified by the use of learning analytics systems for real-time monitoring and intervention, and the leveraging of intelligent recommendation systems to tailor personalized learning paths for students (Bedoya Ulla & Franco Perales, 2022).

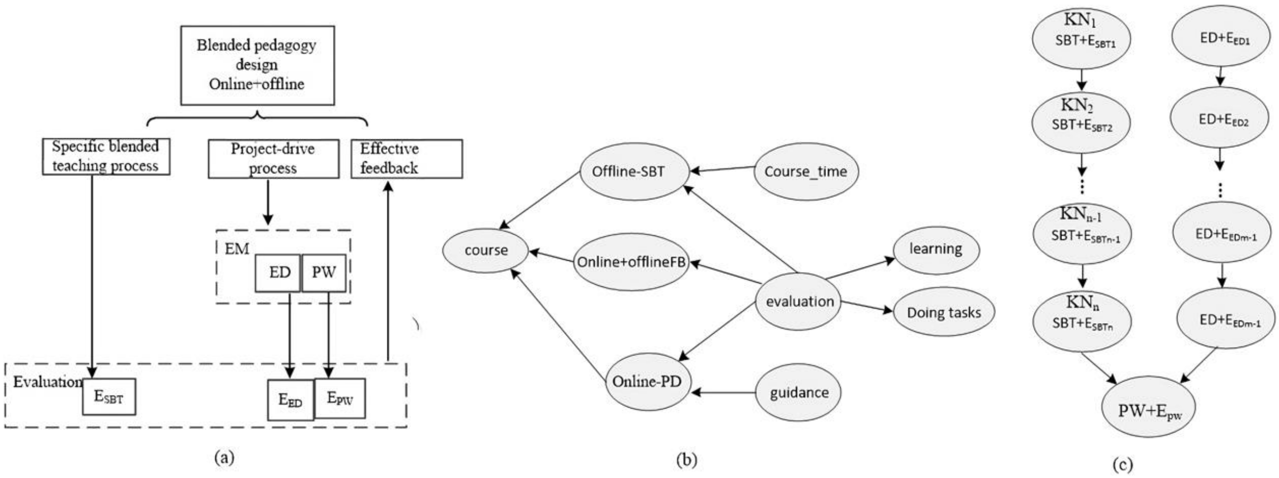

Figure 1. Composition of blended pedagogy: (a) blended pedagogy design, (b) Bayesian network of proposed blended course model, & (c) implementation of a course teaching (Source: Author's own elaboration)

Theoretical Framework & Research Hypotheses of Hybrid Pedagogy

The theoretical foundation of hybrid teaching methods comprehensively draws upon pluralistic education theories such as constructivism, humanism, sociocultural theory, and cognitive load theory. Constructivism underscores learning as an active process of knowledge construction by individuals, necessitating a teaching environment that fosters exploration, collaboration, and problem-solving (Piaget, 1970; Von Glasersfeld, 1989). Humanism emphasizes individual differences and emotional aspects of students, advocating respect for students' autonomy and stimulating learning interest (Maslow, 1954). Sociocultural theory posits that learning occurs within social interactions and cultural practices, highlighting the importance of learning communities and apprenticeship models (Lave & Wenger, 1991). Cognitive load theory cautions against excessive information processing burdens in instructional design, ensuring efficient knowledge acquisition and processing (Sweller, 1988). Grounded in these theories, we posit the following hypotheses.

H1. Adoption of hybrid teaching methods will significantly enhance students' performance in acquiring C++ programming skills, understanding course content, and problem-solving abilities, specifically manifesting in improved grades, increased depth of subject knowledge, and augmented problem-solving capabilities.

H2. Implementation of hybrid teaching methods will significantly elevate students' sense of self-efficacy.

METHODOLOGY

Research Participants & Sample

The study sample comprised the entire cohort of students (n=68) from the software engineering program of a university, admitted in 2021. All participants voluntarily enrolled underwent a year-long C++ course, and actively engaged in completing a series of surveys, contributing data vital for assessing instructional effectiveness.

Teaching Design

Framework of hybrid pedagogy

We devised a hybrid teaching model integrating traditional classroom lectures with online autonomous learning, wherein offline classes prioritize teacher guidance while online components emphasize student autonomy. Enhanced by project-based learning and real-time feedback mechanisms, this model elevates teaching quality. The structure of the hybrid pedagogy, depicted in part a in **Figure 1**, encompasses a specialized blended teaching process (SBT), project-driven learning process (PD), and an efficacious feedback system (FB). The core of the project-driven process involves experimental design (ED) and project work (PW), with respective assessments labeled EED and EPW. Following each assessment task, instructors provide tailored feedback (FB) to students, continually refining instructional outcomes.

Part b in **Figure 1** presents the Bayesian network model of our teaching methodology, which both delineates the hybrid teaching process and serves to evaluate the impact of the designed instructional pathways on students' self-learning efficacy. Within the model, course (C) is comprised of the specific blended teaching component (S), the online-offline integrated feedback mechanism (F), and the online project-driven component (P), collectively encompassing diverse learning modalities such as classroom instruction (course_time), extracurricular mentoring (guidance), and comprehensive assessment (evaluation). It is noteworthy that the course design integrates both in-person classroom teaching and offline learning facilitated through a learning platform. Through a multitude of assessment methods, students can continuously learn via the online platform, devising varied learning trajectories until they achieve their learning objectives. Concurrently, educators can monitor the learning process and outcomes. Part c in **Figure 1** shows implementation of a course teaching.

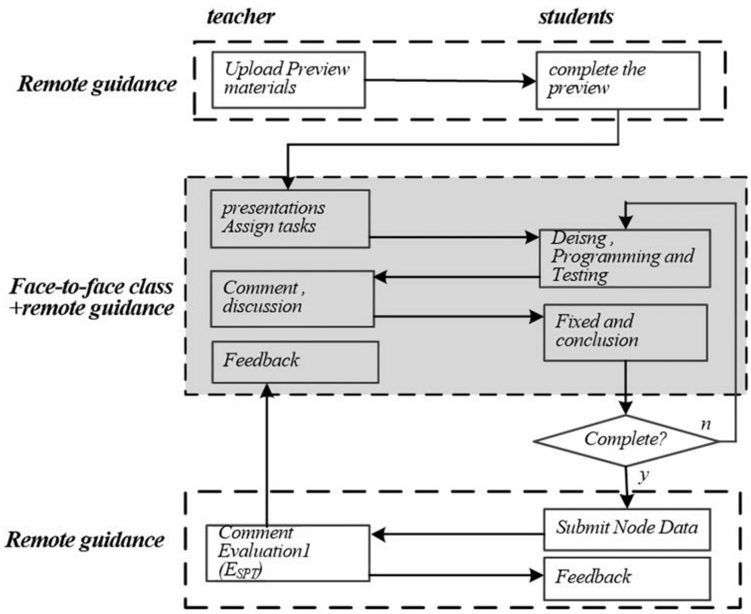

Figure 2. Specific blended teaching process (Source: Author's own elaboration)

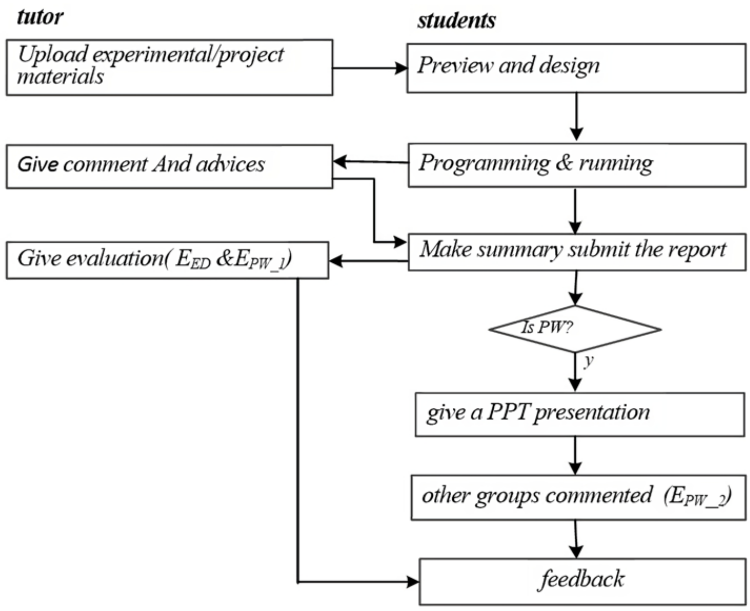

Figure 3. Illustration of project-driven teaching activities & feedback (Source: Author's own elaboration)

Implementation of teaching methodology

The detailed SBT is illustrated in **Figure 2**. It commences with instructors uploading preview materials, followed by students engaging in self-study online and completing short-answer questions. During in-class sessions, teachers systematically explain key concepts and assign tasks related to programming principles, guiding students through the processes of design, coding, and testing to deepen their understanding and mastery of the subject matter. Teachers then review students' programming outputs, facilitate group discussions, prompting students to revise their code, summarize key learning points, and compile them into node data that includes source code, execution results, and individual reflections (ESBT). These node data are submitted via the online platform for teacher evaluation (E1) and serve as components of students' knowledge repositories for future review.

Teachers assess the node data and provide feedback, with the flexibility to offer individualized online or offline tutoring based on students' needs. They also promote peer questioning, answer exchanges, and collaborative suggestions among students. Students are encouraged to iterate through the learning and testing phases until they successfully pass the teacher's assessment.

Project-driven process (PD) and effective feedback mechanism are depicted in **Figure 3**. Following the acquisition of each knowledge node, students engage in practical activities, generating node data from their experiences. To reinforce practical skills and the comprehensive application of professional knowledge, we have incorporated project-based teaching activities, comprising experimental design (ED) throughout the learning journey and project work (PW) after the course. ED is designed to verify acquired knowledge and enhance practical operation skills, whereas PW aims to hone the ability to apply professional knowledge in an integrated manner.

Table 1. Variables collected after course

Variable	Definition
SBT_{T1}-SBT_{T10}	1: In-class submission of data; 2: Submission within the same day; & 3: Submission within the same week for node data.
SBT_{Qu1}-SBT_{Qu10}	Excellent: Includes complete source code, execution results, and summary; Good: Features source code and execution results; Poor: Incomplete lab report, with homework, not fully attempted.
ED_{T1}- ED_{T5}	1: In-class data submission; 2: Same-day data submission; & 3: Within-week node data submission.
ED_{Qu1}-ED_{Qu5}	Excellent: Incorporates all functions with rich human-computer interaction; Good: Basic functions are completed but not thoroughly refined; Poor: Functions remain uncompleted.
PD_{Te1}- PD_{Te5}	Project student evaluation: Categorized into five tiers: 90-100, 80-90, 70-80, 60-70, Below 60
PD_{Stu1}- PD_{Stu5}	Project student assessment: Divided into five ranges:90-100, 80-90, 70-80, 60-70, Below 60
Q_1	When learning a node, what percentage of the knowledge point do you believe you have mastered? 1: 80%-100%, 2: 60%-80%, 3: Below 60%.
Q_2	Do you find it difficult to complete each node data within the class time? 1: Very easy 2: Easy 3: Complex 4: Very complex
Q_3	Do you perceive the setup of PW as challenging?1: Very easy 2: Easy 3: Difficult 4: Very difficult
Q_4	Using specific blended learning approach, were you able to acquire knowledge necessary to complete project? 1: Yes & 2: No
Q5	Which skills have you gained from this course? 1: Programming skills 2: Practical skills 3: Team collaboration skills
ED_{Qu_old1}-ED_{Qu_old5}	Experiment with data using traditional classroom teaching methods
PD_{Te_old1}-PD_{Te_old5}	Project data using traditional classroom teaching methods, with teacher evaluations

Teachers issue experimental/project materials after which students conduct preliminary studies and design frameworks. Teachers provide critiques on initial proposals, suggest improvements, and offer programming guidance. Students then proceed to complete coding and execution tasks, submitting experiment/project reports via the online platform. Teachers review these submissions, yielding evaluations marked as EED or EPW. During the PW phase, students collaborate in groups, with each group preparing a PowerPoint presentation. Other groups offer evaluations, contributing to the formation of EPW_2.

Data Collection

Utilizing the online teaching platform, we systematically gathered objective learning data generated during SBT and PD stages, including node data, records of experiment/project report submissions, teacher grading, and peer evaluations. After the course, questionnaires were administered to students to assess their acceptance of the teaching method and their perceived self-efficacy, capturing their subjective experiences and evaluations of the teaching effectiveness. Specifically, the collected data encompassed: Submission times for the 10 node assignments during SBT (SBT_{T1}-SBT_{T10}) and objective quality metrics of these assignments (SBT_{Qu1}-SBT_{Qu10}), Submission times for five experimental reports (ED_{T1}- ED_{T5}) and corresponding quality data (ED_{Qu1}-ED_{Qu5}), Evaluations from teachers on five projects (PD_{Te1}-PD_{Te5}) and peer evaluations (PD_{Stu1}-PD_{Stu5}). Subjective responses regarding teaching effectiveness were captured through questionnaire items Q1-Q5. Additionally, data from experiments using traditional teaching methods served as a comparison, including quality data from five experimental reports (ED_{Qu_old1}-ED_{Qu_old5}) and project evaluations (PD_{Te_old1}-PD_{Te_old5}). Detailed descriptions of these variables are provided in **Table 1**.

Data Analysis

Initially, we analyze the data from SBT, ED, and PD sessions to intuitively assessment the impact of node configurations on enhancing students' learning outcomes. Subsequently, we employ t-tests to examine the learning achievements under traditional methods versus those attained through the project-driven hybrid education approach proposed herein for both ED and PD segments, aiming to derive relatively precise outcomes. Finally, we utilize Bayesian methods to evaluate students' learning pathways, investigating from which route–ESB, ED, or PD–their sense of self-efficacy predominantly arises, with model outputs informing this exploration. By synthesizing these statistical analyses' findings, we conduct a comprehensive assessment of the overall effect of the hybrid teaching methodology on C++ course learning outcomes for software engineering students.

RESULTS

Exploratory Data Analysis of E_{SPT}, E_{ED}, & E_{PW}

The preliminary statistical results of 10 node datasets are summarized in **Table 2**. An initial examination reveals several key points. First, all 68 students completed and submitted their node data, with the majority managing to finish their assignments within class time and achieving excellent ratings on most occasions. Second, the submission timing and quality of work are correlated with the complexity of the node data. For instance, in the ninth assignment, which was relatively demanding, out of the 33 students who submitted on time, 25 received an 'excellent' rating, suggesting that only a small proportion of students possessed the capability for independent framework design and code writing at this stage. Conversely, in the tenth session, where the assignment was less challenging, among the 52 students who submitted punctually, a notable 56 obtained 'excellent' grades.

Table 3 presents the submission timelines and quality evaluations for the five experiments. Experiment 1 through experiment 4 were relatively straightforward, with nearly all students managing to submit their reports on schedule. A vast majority of them excelled in their assignments, attaining the highest grade category. Conversely, in the fifth, more comprehensive experiment, which integrated various knowledge areas and was thus more challenging, only 43 students completed their work on time, among whom 37 achieved the 'excellent' grade.

Table 2. Node data (ESPT) submission time & quality statistics

Var	In-class submission	Submission within day	Submission within week	Var	Excellent	Good	Poor
SBT_{T1}	38	15	15	SBT_{Qu1}	45	10	13
SBT_{T2}	40	12	16	SBT_{Qu2}	50	10	8
SBT_{T3}	41	17	10	SBT_{Qu3}	50	9	9
SBT_{T4}	51	12	5	SBT_{Qu4}	53	11	4
SBT_{T5}	50	16	2	SBT_{Qu5}	48	10	10
SBT_{T6}	48	15	5	SBT_{Qu6}	47	9	12
SBT_{T7}	46	17	5	SBT_{Qu7}	52	8	8
SBT_{T8}	59	9	0	SBT_{Qu8}	56	7	5
SBT_{T9}	33	23	12	SBT_{Qu9}	25	30	13
SBT_{T10}	52	10	6	SBT_{Qu10}	56	6	6

Table 3. Experiments data (EED) submission time & quality statistics

Var	In-class submission	Submission within day	Submission within week	Var	Excellent	Good	Poor
EDT1	**51**	**13**	**3**	EDQu1	**53**	**10**	**5**
EDT2	**45**	**15**	**8**	EDQu2	**55**	**12**	**1**
EDT3	**42**	**16**	**10**	EDQu3	**55**	**8**	**5**
EDT4	**46**	**17**	**5**	EDQu4	**55**	**8**	**5**
EDT5	**25**	**21**	**22**	EDQu5	**37**	**21**	**10**

Table 4. Statistic results on comprehensive project (EPW)

Var	90-100	80-90	70-80	60-70	<60	Var	90-100	80-90	70-80	60-70	<60
PD_{Te1}	4	8	10	8	0	PD_{Stu1}	5	10	10	5	0
PD_{Te2}	3	8	12	7	0	PD_{Stu2}	8	12	6	3	1
PD_{Te3}	5	9	10	6	0	PD_{Stu3}	10	12	7	1	0
PD_{Te4}	3	10	11	5	1	PD_{Stu4}	8	10	10	2	0
PD_{Te5}	3	10	12	5	0	PD_{Stu2}	8	12	9	1	0

Comprehensive project (EPW) entails not only in-class planning but also post-class discussions and execution by team members. Team members must strategize their individual responsibilities concerning design, coding, debugging, and testing tasks. Completion of the comprehensive project necessitates teamwork, with each member accountable for their assigned module. Assessment of the comprehensive project comprises two parts: one from the instructor and another from fellow team members.

Statistical outcomes in **Table 4** illustrate that projects demonstrating superior design and well-written reports receive higher scores from the teacher's evaluation, considering classroom time and student performance. From the student perspective, teams that present their projects confidently score higher, while those with members less adept at public speaking tend to receive lower marks.

Thus, by adopting a node-based teaching approach complemented with project-driven learning, implementing a blended mode of offline and online instruction, providing timely and effective feedback, and administering appropriate assessments at each teaching juncture, the preliminary exploration of these data suggests that this teaching methodology effectively stimulates student motivation, enhances their enthusiasm for learning, enables the majority of students to submit assignments on schedule, and achieves favorable learning outcomes.

Comparison of Traditional Teaching & Proposed Project-Driven Hybrid Method

To investigate whether the new teaching approach significantly improves student performance, we employed t-tests to compare the educational outcomes in the C++ course, identical experiments, and projects between traditional teaching methods and the novel project-driven hybrid approach presented herein. The results revealed a remarkable advantage of the proposed teaching methodology in both experimental grades and project scores.

In all five experiments (ED_{Qu1} through ED_{Qu5}), the average scores of students under the new teaching method exceeded those of the traditional method, with the mean differences ranging from -0.053 to -0.574, indicating a significant improvement in students' experimental grades. The p-values for the left-tail tests in all experiments were less than 0.001, reaching a highly significant level, robustly validating the hypothesis that the new teaching method surpasses the traditional one. Effect sizes, as reflected by the t-statistics, ranged from -3.387 to -8.682, demonstrating moderate to large effects. Specifically, Experiment 5 (ED_{Qu5}) exhibited the largest effect, where the new teaching method had the most pronounced impact on enhancing experimental scores.

In the five comparable projects (PD_{Stu1} through PD_{Stu5}), students' average scores under the new teaching method were also higher than those under the traditional method, with mean differences ranging from -0.235 to -0.382, further confirming the superiority of the new teaching method in boosting project scores. Similarly, the p-values for the left-tailed tests in all projects were less than 0.001, attaining a highly significant level, affirming the new teaching method's advantage over the traditional one in terms of project scores. The t-statistics, ranging from -4.074 to -6.440, denote moderate to large significant disparities between the new and traditional teaching methods in project scores. Notably, the third project (PDS_{tu3}) showed the greatest effect, where the new teaching method had the most substantial impact on enhancing students' scores.

Table 5. Comparison of teaching effectiveness differences

Var-var	Mean	Standard error	Standard deviation	95% confidence interval	t-statistic	Left-tailed p-value
ED_{Qu1} ED_{Qu_old1}	-0.191	0.056	0.465	[-0.304, 0.079]	-3.387	<0.001
ED_{Qu2} ED_{Qu_old2}	-0.574	0.076	0.630	[-0.726,-0.042]	-7.501	<0.001
ED_{Qu3} ED_{Qu_old3}	-0.485	0.090	0.743	[-0.665, -0.304]	-5.386	<0.001
ED_{Qu4} ED_{Qu_old4}	-0.059	0.097	0.796	[-0.781, -0.396]	-6.093	<0.001
ED_{Qu5} ED_{Qu_old5}	-0.053	0.061	0.503	[-0.651, -0.408]	-8.682	<0.001
PD_{Stu1} PD_{Te_old1}	-0.294	0.056	0.459	[-0.405, -0.183]	-5.284	<0.001
PD_{Stu2} PD_{Te_old2}	-0.280	0.055	0.452	[-0.389, -0.170]	-5.097	<0.001
PD_{Stu3} PD_{Te_old3}	-0.382	0.059	0.490	[-0.501 -0.264]	-6.440	<0.001
PD_{Stu4} PD_{Te_old4}	-0.235	0.052	0.427	[-0339, -0.132]	-4.540	<0.001
PD_{Stu2} PD_{Te_old5}	-0.265	0.065	0.536	[-0.039, -0.135]	-4.074	<0.001

Table 6. Impact of node assignment evaluations on students' perceived level of mastery

Variable	Mean	Standard deviation	MCSE	Median	95% confidence interval	
Age	-0.0291	0.1051	0.0093	-0.0323	-0.2276	0.1773
Gender	-0.0833	0.3234	0.0140	-0.0944	-0.7090	0.5633
Node submission quality	0.9199	0.2398	.0146	0.9063	0.4880	1.4303

Table 7. Impact of node assignment evaluations on students' perceived difficulty

Variable	Mean	Standard deviation	MCSE	Median	95% confidence interval	
Age	-0.0980	0.1133	0.0051	-0.0960	-0.3209	0..1361
Gender	-0.1186	0.3702	0.0179	-0.0964	-0.8636	0.6029
Node submission quality	1.0039	0.2570	0.0138	0.9817	0.5705	1.5778

Table 8. Impact of teacher & student evaluations on students' perceived difficulty (q3) for projects

Variable	Mean	Standard deviation	MCSE	Median	95% confidence interval	
Age	-0.0104	0.1154	0.0048	-0.0090	-0.2403	0.2108
Gender	-0.0169	0.4352	0.0229	-0.0175	-0.8278	0.8211
Teacher evaluation	0.0169	0.4352	0.0229	0.0175	-0.8278	0.8211
Student evaluation	0.8350	0.4484	0.0215	0.8192	0.0427	1.7774

In summary, the new teaching method demonstrates a statistically significant advantage over the traditional method across two pivotal learning outcome indicators: experimental grades and project scores. Furthermore, the consistently large t-statistics underscore the substantial positive influence of the new teaching method on both experimental and project performance. Consequently, for the same cohort of students engaged in identical experiments and projects, the new teaching method has effectively enhanced students' mastery of the subject matter. This evidence suggests that, under current conditions, the new teaching method holds a significant advantage over the traditional approach, providing robust grounds for further exploring its effectiveness in different contexts and justifying its broader implementation. **Table 5** shows comparison of teaching effectiveness differences.

Bayesian Analysis of Learning Paths & Student Self-Efficacy

Upon completion of all knowledge points, we harvested students' learning data from the online platform for evaluation, extracting characteristics of their learning behaviors to inform adjustments to teaching methodologies. To further enhance learning outcomes, we also incorporated questionnaires during the course addressing classroom learning, experiments, and comprehensive projects, to gather feedback on learning experiences and evaluate knowledge mastery, thereby providing material for effective improvement of teaching effectiveness.

To better evaluate the impact of online and offline learning outcomes on students' sense of self-efficacy, we first conducted a factor analysis on the submission times and assignment qualities of the ten experimental nodes, as well as experiment and project data. This allowed us to obtain variables representing the submission time (Tq) and quality (Qq) of online class node data, the submission time (Te) and quality (Qe) combining online and offline experiments, and the quality (Qp) of offline project assignments. Subsequently, we employed Bayesian modelling to analyze their influence on questions q1 and q5 related to self-efficacy. The results are presented below.

Table 6 and **Table 7** respectively undertake Bayesian analysis focusing on the mastery level (q1) and perceived difficulty (q2) of experiments. It becomes evident that students who received better evaluations on node assignments perceive themselves as having mastered more knowledge and found the corresponding topics easier. This indirectly confirms that, for an individual student, using grades as a measure of teaching effectiveness is indeed reasonable.

Table 8 and **Table 9** present the results of the self-efficacy analysis based on project data, specifically addressing the perceived difficulty (q3) and the capacity for completion (q4), with results akin to those shown in Tables 5 and 6. The findings indicate that students receiving higher peer evaluations tend to perceive tasks as easier, and concurrently, peer evaluations of projects significantly influence students' self-assessments of their completion capabilities. Conversely, teacher evaluations show no significant impact. Moreover, Table X illustrates that teacher evaluations focus more on the quality of the completed work itself,

Table 9. Impact of teacher & student evaluations on students' perceived ability to complete tasks (q4) in projects

Variable	Mean	Standard deviation	MCSE	Median	95% confidence interval	
Age	-0.0501	0.0827	0.0171	-0.0471	-0.2277	0.0927
Gender	-0.4427	0.3035	0.0142	-0.4315	-1.0519	0.1179
Teacher evaluation	-0.5401	0.3964	0.3965	-0.5474	-1.3361	0.2238
Student evaluation	1.2852	0.4099	0..0244	1.2810	0.4840	2.1316

Table 10. Effect of teacher and student evaluations on students' perceived capability (q5) in projects

Variable	Mean	Standard deviation	MCSE	Median	95% confidence interval	
Age	0.0573	0.0593	0.0162	-0.0638	-0.0529	0.1562
Gender	0.0675	0.3124	0.0465	0.0770	-0.5388	0.6348
Node evaluation	-2.3326	0.4609	0.0078	-2.3649	-3.1824	-1.3599
Experiment evaluation	0.8272	0.4045	0.0756	0.8461	0.04324	1.5663
Teacher assessment	-0.9998	0.2049	0.0437	-0.9974	-1.3990	-0.6403
Student assessment	0.4090	0.2971	0.0572	0.4052	-0.1577	1.0327

whereas students consider their performance during presentations, suggesting that grades do affect students' self-efficacy to a certain extent. Additionally, self-efficacy is also related to students' analytical abilities and communication skills. These findings align with previous research.

Table 10 presents the results regarding students' self-perceived enhancement of their abilities in the course, where ability is quantified as the sum of skills acquired by each student, with a range from 1 to 6; a higher value indicates a greater acquisition of skills. The findings indicate that both experimental performance and teacher evaluations have the most significant impact on students' self-assessments but in contrasting ways. Students with higher experiment scores tend to perceive that they have gained more abilities. Conversely, those who received higher evaluations from teachers paradoxically reported lower self-perceived acquisition of abilities. Meanwhile, the node data (potentially referring to specific stages or milestones within the course) shows no significant effect on students' perceived skill enhancement.

DISCUSSION & LIMITATIONS

This paper illustrates a new teaching method for computer programming courses using C++ as a case study, highlighting the effectiveness of blended learning in cultivating flexible learning strategies, enhancing academic performance, and boosting student satisfaction. The COVID-19 pandemic necessitated many universities to adopt remote online education as a means to curb transmission (Anthony et al., 2019). While some institutions retained conventional in-person teaching, blended teaching methodologies gained prominence to curtail online teaching durations and augment instructional effectiveness (Tadlaoui & Chekou, 2021).

The proposed blended teaching paradigm integrates a structured blend of teaching procedures, project-based learning, and efficient feedback. Face-to-face SBT sessions facilitate vital communication channels between educators and learners. Feedback on new course content (KN) is imparted in class to optimize learning outcomes.

Bayesian learning introduces multiple learning pathways for students, enabling them to progress through knowledge nodes towards defined objectives, with EED, EPR, and ESPT serving as online assessments. EED fosters fundamental theory comprehension, EPR enhances student motivation and hands-on skills (Brake et al., 2018), and ESPT cultivates entrepreneurial competencies (Ma et al., 2014). Effective feedback on student assignments further stimulates student initiative (Wang et al., 2021). As anticipated, evaluations in EP and ED revealed improved learning outcomes and satisfaction levels compared to traditional teaching methods. Based on a year-long C++ programming course, this blended teaching approach emerges as ideal for instructing computer programming languages.

Reasons for this success potentially lie in the provision of more flexible and varied learning and assessment formats, coupled with a reduced need for in-person teaching hours, stimulation of autonomous and continuous learning, and enhanced learning outcomes. Computer programming demands more than theoretical knowledge alone (Tadlaoui & Chekou, 2021). The Bayesian learning model proposed here offers students additional learning options and facilitates ongoing improvement in programming skills through a diverse array of evaluative measures.

The primary challenge with this teaching method is ensuring prompt communication and guidance when students encounter difficulties during online study, either from teachers or peers. Furthermore, it imposes new demands on educators, who must not only teach technical content but also instruct on the assessment methodologies within the learning platform. In other words, teachers must also facilitate interactive sessions, deliver timely and constructive feedback, evaluate through reflective practices, and develop various digital visual e-learning materials. These aspects represent ongoing areas for improvement in the future.

CONCLUSIONS & FUTURE RESEARCH DIRECTIONS

This study meticulously planned and successfully executed an innovative hybrid teaching model, infusing fresh momentum into the realm of computer programming language education, particularly in the practice of C++ programming courses.

Throughout a year's educational implementation, we not only validated the outstanding effectiveness of this teaching strategy, significantly improving teaching quality and student satisfaction, but also, through the adoption of Bayesian analysis, revealed a profound positive correlation between learning paths and the enhancement of student self-confidence. This further confirmed academic achievement as a crucial indicator in assessing the effectiveness of teaching quality, laying a solid empirical foundation for refining personalized learning trajectories in future endeavors. These research outcomes not only reinforce the role of hybrid teaching methodologies in advancing programming education but also equip educators with practical strategies and theoretical foundations for implementation.

Looking ahead, the frontier of research will pivot toward the deep integration of technology and personalized learning. With rapid technological advancements, future explorations should concentrate on how artificial intelligence, especially machine learning techniques, can more intricately tailor growth trajectories for individual learners. The objective is to establish a system capable of intelligently allocating educational resources based on students' learning dynamics and competency levels, thereby maximizing learning efficiency and personal potential development, ushering in a new era of personalized education.

Funding: No funding source is reported for this study.

Ethical statement: The author stated that this study included an online questionnaire. The author further stated that the study adheres to ethical guidelines, ensuring participant consent and data privacy were strictly maintained.

Declaration of interest: No conflict of interest is declared by the author.

Data sharing statement: Data supporting the findings and conclusions are available upon request from the author.

REFERENCES

Aivaloglou, E., & Hermans, F. (2016). How kids code and how we know: An exploratory study on the Scratch repository. In *Proceedings of the 2016 ACM Conference on International Computing Education Research* (pp. 53-61). https://doi.org/10.1145/2960310.2960325

Akcayir, G., & Akcayir, M. (2018). The flipped classroom: A review of its advantages and challenges. *Computers & Education, 126*, 334-345. https://doi.org/10.1016/j.compedu.2018.07.021

Anthony, B., Kamaludin, A., Romli, A., Raffei, A. F. M., Nincarean A/L Eh Phon, D., Abdullah, A., Ming, G. L., Shukor, N. A., Nordin, M. S., & Baba, S. (2019). Exploring the role of blended learning for teaching and learning effectiveness in institutions of higher learning: An empirical investigation. *Education and Information Technologies, 24*, 3433-3466. https://doi.org/10.1007/s10639-019-09941-z

Awotunde, J. B., Ayo, E. F., Ajamu, G. J., Jimoh, T. B., & Ajagbe, S. A. (2023). The influence of industry 4.0 and 5.0 for distance learning education in times of pandemic for a modern society. In *Advances in distance learning in times of pandemic* (pp. 177-214). Chapman and Hall/CRC. https://doi.org/10.1201/9781003322252-8

Bedoya Ulla, M., & Franco Perales, W. (2022). Hybrid teaching: Conceptualization through practice for the post-COVID-19 pandemic education. *Frontiers in Education, 7*, Article 924594. https://doi.org/10.3389/feduc.2022.924594

Brake, N. A., Jao, M., & Su, D. (2018). Integration of micro design-build-test projects in instructor-centered courses to increase student confidence. *Journal of Professional Issues in Engineering Education and Practice, 144*(2), Article 05018002. https://doi.org/10.1061/(ASCE)EI.1943-5541.0000355

Bruner, J. S. (1961). The act of discovery. *Harvard Educational Review, 31*, 21-32.

Chin, J., & Kozimor, M. L. (2023). *Emerging stronger: Pedagogical lessons from the pandemic*. Taylor & Francis. https://doi.org/10.4324/9781003316336

Cubric, M. (2008). Agile learning & teaching with wikis: Building a pattern. In *Proceedings of the 4th International Symposium on Wikis* (pp. 1-2). https://doi.org/10.1145/1822258.1822296

Davidovitch, N. (2013). Learning-centered teaching and backward course design–From transferring knowledge to teaching skills. *Journal of International Education Research, 9*(4), Article 329. https://doi.org/10.19030/jier.v9i4.8084

DeLozier, S. J., & Rhodes, M. G. (2017). Flipped classrooms: A review of key ideas and recommendations for practice. *Educational Psychology Review, 29*(1), 141-151. https://doi.org/10.1007/s10648-015-9356-9

Dewey, J. (1938). *Experience and education*. Macmillan Company.

Freire, P. (1970). *Pedagogy of the oppressed*. Seabury Press.

Johnson, D. (2013). *Media franchising: Creative license and collaboration in the culture industries*. NYU Press.

Kizilcec, R. F., & Schneider, E. (2015). Motivation as a lens to understand online learners: Toward data-driven design with the OLEI scale. *ACM Transactions on Computer-Human Interaction, 22*(2), 1-24. https://doi.org/10.1145/2699735

Kokotsaki, D., Menzies, V., & Wiggins, A. (2016). Project-based learning: A review of the literature. *Improving Schools, 19*(3), 267-277. https://doi.org/10.1177/1365480216659733

Krogh, J., Saltin, B., Gluud, C., & Nordentoft, M. (2009). The DEMO trial: A randomized, parallel-group, observer-blinded clinical trial of strength versus aerobic versus relaxation training for patients with mild to moderate depression. *Journal of Clinical Psychiatry, 70*(6), Article 790. https://doi.org/10.4088/JCP.08m04241

Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge University Press. https://doi.org/10.1017/CBO9780511815355

Li, J., & Xing, J. (2020). Why is collaborative agglomeration of innovation so important for improving regional innovation capabilities? A perspective based on collaborative agglomeration of industry-university-research institution. *Complexity, 2020*, 1-21. https://doi.org/10.1155/2020/7049606

Lydia, K., Lily, Y., & Oleksandr, G. (2024). Microbiota and the control of blood-tissue barriers: Problem-based-learning of students at the medical university. In *Proceedings of the 13th International Scientific and Practical Conference "Information and Innovative Technologies in the Development of Society"*.

Ma, K., Teng, H., Du, L., & Zhang, K. (2014). Project-driven learning-by-doing method for teaching software engineering using virtualization technology. *International Journal of Emerging Technologies in Learning, 9*. https://doi.org/10.3991/ijet.v9i9.4006

Maslow, A. H. (1954). The instinctoid nature of basic needs. *Journal of Personality, 22*(3), 326-347. https://doi.org/10.1111/j.1467-6494.1954.tb01136.x

O'Donnell, A. M., & Hmelo-Silver, C. E. (2013). Introduction: What is collaborative learning?: An overview. In C. Hmelo-Silver, C. A. Chinn, C. Chan, & A. O'Donnell (Eds.), *The international handbook of collaborative learning* (pp. 1-15). Routledge. https://doi.org/10.4324/9780203837290

Omona, K., & O'dama, M. K. (2024). *Global perspectives on micro-learning and micro-credentials in higher education*. IGI Global. https://doi.org/10.4018/979-8-3693-0343-6

Palmer-Abbs, M., Cottrill, C., & Farrington, J. (2021). The digital lottery: The impact of next generation broadband on rural small and micro businesses in the Northeast of Scotland. *Journal of Rural Studies, 81*, 99-115. https://doi.org/10.1016/j.jrurstud.2020.08.049

Papadakis, S. (2020). Evaluating a game-development approach to teach introductory programming concepts in secondary education. *International Journal of Technology Enhanced Learning, 12*(2), 127-145. https://doi.org/10.1504/IJTEL.2020.106282

Papadakis, S., & Kalogiannakis, M. (2019). Evaluating the effectiveness of a game-based learning approach in modifying students' behavioural outcomes and competence, in an introductory programming course. A case study in Greece. *International Journal of Teaching and Case Studies, 10*(3), 235-250. https://doi.org/10.1504/IJTCS.2019.102760

Parks, T. (2021). *Technological impacts during transition from face-to-face teaching to synchronous during the pandemic* [PhD thesis, Capella University].

Piaget, J. (1970). *Genetic epistemology*. Columbia University Press. https://doi.org/10.1177/000276427001300320

Sharma, M., Biros, D., Ayyalasomayajula, S., & Dalal, N. (2020). Teaching programming to the post-millennial generation: Pedagogic considerations for an is course. *Journal of Information Systems Education, 31*(2), 96-105.

Shulman, L. S. (1986). Those who understand: Knowledge growth in teaching. *Educational Researcher, 15*(2), 4-14. https://doi.org/10.3102/0013189X015002004

Smith, W. C., & Fernandez, F. (2017). Education, skills, and wage gaps in Canada and the United States. *International Migration, 55*(3), 57-73. https://doi.org/10.1111/imig.12328

Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive Science, 12*(2), 257-285. https://doi.org/10.1207/s15516709cog1202_4

Tadlaoui, M. A., & Chekou, M. (2021). A blended learning approach for teaching python programming language: Towards a post pandemic pedagogy. *International Journal of Advanced Computer Research, 11*(52), Article 13. https://doi.org/10.19101/IJACR.2020.1048120

Takona, J. P. (2023). Flipping the script: Post-COVID-19 opportunities, challenges, and threats amidst the digital revolution in higher education. *International Journal on Social and Education Sciences, 5*(4), 847-863. https://doi.org/10.46328/ijonses.569

Tarimo, W. T. (2016). *Computer-supported agile teaching* [PhD thesis, Brandeis University].

Von Glasersfeld, E. (1989). Cognition, construction of knowledge, and teaching. *Synthese, 80*, 121-140. https://doi.org/10.1007/BF00869951

Wang, H., Tlili, A., Lehman, J. D., Lu, H., & Huang, R. (2021). Investigating feedback implemented by instructors to support online competency-based learning (CBL): A multiple case study. *International Journal of Educational Technology in Higher Education, 18*, 1-21. https://doi.org/10.1186/s41239-021-00241-6