



Research on Outcome-Based Education Reform in Python Programming Instruction

Quanfu Wang¹, Yang Tao^{2*}, Xuanjiao Lyu¹, Huanhui Liang¹

¹School of Electrical and Computer Engineering, Guangzhou Nanfang College, Guangzhou, China

²College English Education Center, Guangzhou Nanfang College, Guangzhou, China

Email: *taoyangty@163.com

How to cite this paper: Wang, Q.F., Tao, Y., Lyu, X.J. and Liang, H.H. (2025) Research on Outcome-Based Education Reform in Python Programming Instruction. *Open Access Library Journal*, 12: e14235.
<https://doi.org/10.4236/oalib.1114235>

Received: September 8, 2025

Accepted: October 7, 2025

Published: October 10, 2025

Copyright © 2025 by author(s) and Open Access Library 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

Abstract

Under the background of artificial intelligence empowering higher education, this study addresses the issues of outdated content, singular teaching methodologies, one-sided assessment, and insufficient cultivation of student interest in the traditional teaching of the course *Python Programming*. Based on the Outcome-Based Education (OBE) concept, this study constructs a teaching system featuring clearly defined outcomes, backward design, multidimensional assessment, and continuous improvement. With objectives encompassing knowledge, skills, and literacy, this study designs experimental cases such as gaining hands-on experience with edge computing platforms, deploying generative AI, and applying the MediaPipe framework. These cases drive the backward optimization of course content and teaching methods, alongside the establishment of a multi-dimensional assessment mechanism. Results indicate a significant enhancement in students' knowledge mastery, practical skills, and teamwork capabilities. The relevance of teaching content to industry needs was strengthened, and a sustainable mechanism for continuous improvement was formed, providing an actionable pathway for cultivating computer science talents within the context of Emerging Engineering Education.

Subject Areas

Artificial Intelligence

Keywords

Artificial Intelligence, Python Programming, Outcome-Based Education, Backward Design, Emerging Engineering Education

1. Introduction

With the steady advancement of artificial intelligence empowering educational

transformation, the ongoing exploration of independent pathways for cultivating top-tier innovative talents urgently needed by the nation grows increasingly important. In January 2025, the Central Committee of the Communist Party of China and the State Council issued the “Outline of the Plan for Building a Strong Education System (2024-2035)”. This document calls for exploring novel models for national top-tier innovative talent cultivation in strategically critical and emerging fields, aiming to develop first-class core curricula, textbooks, practical projects, and faculty teams [1]. Focusing on constructing a high-quality education system, the plan emphasizes strengthening the cultivation of innovation capabilities and promoting interdisciplinary integration. In July 2025, the General Office of the Ministry of Education, in its “Notice on Organizing and Implementing the Digital Empowerment Initiative for Teacher Development,” highlighted the need to bolster the development of new-type educational infrastructure, deepen the application of technologies like artificial intelligence and big data in teaching and learning, drive the transformation of teaching philosophies, methods, and models, and enhance the quality of education [2]. To augment the comprehensive strength of higher education and forge strategically oriented capabilities, universities have newly established the specialized course *Python Programming*.

As an advanced educational philosophy, Outcomes-Based Education (OBE) has now evolved into a relatively complete theoretical system and practical model. In 1994, Spady W.D., author of the book *Outcome-Based Education: Critical Issues and Answers*, defined OBE as “clearly focusing and organizing the educational system around ensuring that all students demonstrate the essential learnings for success upon finishing school” [3]. The Washington Accord fully adopts the OBE concept and integrates it throughout its engineering education accreditation standards. Drawing on the accreditation philosophy and methods of the American Board for Engineering and Technology (ABET), the Washington Accord establishes twelve graduate attribute requirements, including: Engineering Knowledge, Problem Analysis, Design/Development of Solutions, Investigation, Modern Tool Usage, The Engineer and Society, Environment and Sustainability, Ethics, Individual and Teamwork, Communication, Project Management and Finance, and Lifelong Learning [4]. China became a full signatory member of the Washington Accord in 2016.

2. Educational Model Based on OBE

2.1. Introduction to the OBE Concept

Within the framework of OBE, learning outcomes are defined in a manner that is both explicit and observable. As Faouzi *et al.* (2003) point out that such outcomes should be assessed through evidence of students’ performance demonstrated after they have engaged in a substantial set of learning experiences [5]. In OBE, learning outcomes refer specifically to the “tangible and measurable performance of student learning” [6]. The OBE concept embodies a “student-centered” educational philosophy. In OBE education, the following five core questions should

be addressed:

- 1) Objective: What do we want students to learn and master?
- 2) Needs: Why do we want students to learn these things?
- 3) Process: How do we help students achieve these learning outcomes?
- 4) Assessment: How do we effectively know that students have achieved these learning outcomes?
- 5) Improvement: How do we ensure that students achieve these learning outcomes effectively?

The OBE-oriented teaching process places central emphasis on outputs, including the intended learning outcomes, the strategies for achieving them, and the methods for assessing them. This teaching process prioritizes student-centeredness and active learning. Guided by ongoing student feedback, it maintains a strong focus on learning results while fostering continuous improvement in both teaching and learning. Student performance is evaluated through comprehensive and ongoing assessments. Grounded in clearly defined learning outcomes and supported by pre-assessment, this model facilitates credit transfer, enabling students to engage in multidisciplinary studies across different institutions. Such flexibility increases opportunities for minor programs and student exchanges. Given its outcome-driven nature, instructors are expected to closely monitor student learning progress, articulate cultivation objectives as well as graduation requirements, and establish measurable targets for their achievement. The teaching plan should explicitly demonstrate its alignment with these graduation requirements. For instructors, delivering “high-quality” instruction involves not only providing targeted support but also systematically evaluating the extent to which each graduation requirement is met.

2.2. Three Core Principles of OBE

1) Student-Centered Educational Philosophy: Instructional design is structured to ensure that students meet clearly defined standards in knowledge, skills, and literacy. The allocation of teaching resources—including faculty expertise and curriculum design—is directed toward guaranteeing the achievement of student learning objectives. Meanwhile, Quality assurance and assessment processes adopt student learning outcomes as the sole measure of effectiveness.

2) Outcome-oriented Educational System: First, cultivation objectives are established based on four key considerations: the needs of national society and educational development; the requirements of industry, industrial sectors, and the workplace; the institution’s positioning and development goals; and the expectations of parents and alumni for students’ future growth. Second, these cultivation objectives inform the formulation of graduation requirements, instructional indicators (teaching objectives), curriculum structures, teaching requirements, and course content. Finally, feedback from teaching evaluations is employed to refine and optimize the graduation requirements.

3) Continuous Improvement Quality Assurance Mechanism: Cultivation object-

tives, graduation requirements, and teaching processes all undergo evaluation. Clear mechanisms and evaluation cycles are established, and the results of these evaluations are systematically applied to drive continuous improvement.

2.3. Five Key Steps for Implementing OBE

1) Define Learning Outcomes: Learning outcomes should be clearly articulated and either directly or indirectly measurable; therefore, they need to be translated into specific performance indicators. The process of defining these outcomes must fully consider the requirements and expectations of educational stakeholders, including government bodies, institutions, employers, students, faculty, and parents. In the OBE framework, defined learning outcomes serve not only as the endpoint of the educational process but also as its starting point.

2) Construct The Curriculum System: Learning outcomes form a competency framework that the curriculum is designed to achieve. A clear and explicit mapping should exist between this framework and the curriculum structure to ensure that, upon completion of their studies, students acquire the intended competencies. Each competency within the framework should be explicitly supported and reinforced by designated courses.

3) Define Teaching Strategies: OBE places particular emphasis on the outputs of the teaching process, along with research-informed teaching models and personalized instructional methods. Personalized teaching requires instructors to accurately assess each student's learning status and to develop a timely understanding of their individual goals, foundational knowledge level, and learning progress. Based on these diverse needs, corresponding instructional plans are designed to provide differentiated learning opportunities.

4) Implement Self-referenced Assessment: In OBE, teaching assessment centers on learning outcomes rather than on teaching content, instructional time, or learning styles. Multi-dimensional and tiered assessment criteria are employed, focusing on the achievement of intended learning outcomes and individual progress, rather than on comparisons among students. Learners are evaluated against specific criteria and assigned proficiency levels (ranging from unskilled to excellent) based on the extent to which they meet the educational requirements. A clear understanding of each student's learning status provides valuable reference for both institutional and instructional improvement.

5) Achieve Culminating Outcomes Through Progressive Stages: Student learning is organized into distinct stages, with specific learning targets defined for each stage. These learning targets advance from foundational to higher levels, ultimately ultimately leading to peak outcomes. This staged approach recognizes that students with different learning abilities may reach the same ultimate goals through varying amounts of time, and also through different pathways and methods.

3. Current State of Traditional Teaching

Through multiple research methods including questionnaires, student symposiums,

and faculty exchange meetings—an in-depth analysis of the current teaching status of the *Python Programming* course reveals four main issues:

1) Teaching Content: The course content is predominantly centered on fundamental programming knowledge, with limited integration of advanced technologies. It remains largely disconnected from the technological priorities of Emerging Engineering Education, such as edge computing, generative AI, and human pose estimation. This lack of timely and effective incorporation of cutting-edge industry developments results in insufficient student exposure to frontier knowledge, diminished learning interest, and difficulty in cultivating innovative thinking. Some textbook materials are outdated, and the links between chapters lack cohesion, undermining the systematic and holistic structure of the course. As a result, students struggle to build a comprehensive knowledge framework. Instead, they acquire fragmented technical skills. When faced with real-world engineering problems, they often demonstrate weak knowledge transfer abilities, finding it difficult to apply classroom-acquired concepts flexibly to practical problem-solving.

2) Teaching Methodology: Teacher-centered, one-way knowledge transmission remains dominant, and traditional teaching methods place excessive emphasis on theoretical instruction. Classroom interaction remains limited, leaving most students as passive recipients of information. They also have limited opportunities for active thinking and engaged participation. Traditional teaching approach suppresses intellectual engagement and innovative capacity, while depriving students of immersive experiences in model construction, ultimately reducing knowledge conversion efficiency. Although some instructors have attempted to use multimedia teaching aids, these often amount to little more than projecting textbook content onto a screen, lacking dynamism and interactivity. Practical teaching components are relatively weak, with few comprehensive or design-oriented experiments. Consequently, students' practical and innovative abilities remain underdeveloped. Furthermore, the slow development of a repository of representative case studies results in examples that are poorly aligned with real-world applications. This undermines the effectiveness of project-driven practical teaching. The shortage of relevant resources also impedes the cultivation of students' engineering competencies, creating a substantial gap between their abilities and actual industry hiring standards.

3) Teaching Assessment: The current course evaluation system remains largely monolithic, limiting the development of students' comprehensive abilities. In terms of assessment methods, some institutions continue to rely heavily on final written examinations as the primary evaluation tool, with regular grades carrying minimal weight. Excessive emphasis is placed on the rote memorization of theoretical knowledge, while the assessment of practical skills, innovative thinking, teamwork, and other holistic competencies is neglected. This rigid evaluation model confines students within an examination-oriented learning paradigm, diverting their attention from enhancing practical application skills. Moreover, regular grades are often based primarily on attendance and homework completion, lacking

a thorough evaluation of the learning process itself. As a result, it becomes difficult to accurately reflect students' learning attitudes and levels of effort. Such a monolithic assessment system fails to precisely measure actual learning outcomes and inadvertently reinforces the tendency to prioritize theory over practice, thereby perpetuating an inefficient teaching cycle that hampers the cultivation of high-quality graduates with innovative capabilities and engineering literacy.

4) Cultivation of Student Interest and Competencies: Due to limitations in teaching content and methodology, some students lack interest in the course, perceiving it as dull and uninspiring. The prevailing teaching environment, which overemphasizes theoretical instruction at the expense of practical application, makes it challenging for students to rapidly acquire systematic knowledge and to apply theoretical concepts in practice. Furthermore, the abstract nature of theoretical knowledge, combined with insufficient practical components, leads to unengaging classroom atmospheres, reduced instructional effectiveness, and low student participation. In terms of competency development, students' practical skills, innovative thinking abilities, and teamwork capabilities remain underdeveloped, falling short of the comprehensive talent requirements outlined in the context of Emerging Engineering Education.

4. Research Method of Teaching Reform Based on the OBE Concept

This study selected two cohorts of students majoring in Software Engineering (Class of 2020 and Class of 2022) at Guangzhou Nanfang College, comprising a total of 73 participants. All students had completed the course C Language Programming, possessed basic programming skills, and had not systematically studied Python or any prerequisite content related to artificial intelligence.

Three evaluation instruments were employed in this study. The first was the Knowledge Mastery Questionnaire, which covered dimensions such as Python fundamentals, edge computing, and artificial intelligence technologies. The second was the Teamwork Evaluation Questionnaire, focusing on task allocation, communication efficiency, and professional ethics. The third was the Experimental Evaluation Scale, which comprised three primary indicators—environment configuration, system operation, and problem debugging—weighted at 30%, 40%, and 30%, respectively.

During the second semester of the 2022-2023 academic year, the traditional teaching model was implemented for the Class of 2020 as a control phase. In contrast, during the second semester of the 2024-2025 academic year, an OBE-based teaching reform was carried out for the Class of 2022, incorporating three experimental case-based teaching activities. MATLAB was used for data processing and analysis to compare changes in students' practical performance before and after the teaching reform. For example, a pretest of the Knowledge Mastery Questionnaire conducted in June 2025 on 32 students from the same major yielded a Cronbach's α coefficient of 0.89, indicating good reliability. In the generative AI

deployment experiment, two frontline instructors independently graded students' assignments based on the Experimental Evaluation Scale, resulting in an intraclass correlation coefficient (ICC) of 0.92, which reflected a high level of inter-rater consistency.

5. Practice of Teaching Reform Based on the OBE Concept

Teaching objectives and activities for the *Python Programming* course should be developed through backward design, beginning with the defined culminating learning outcomes. Course and instructional design then proceed in reverse from these final outcomes to determine the appropriateness of all instructional elements leading to their achievement. The starting point for instruction is not what the instructor intends to teach, but what is required to enable students to achieve the final learning outcomes. Implementing backward design requires adherence to two key principles. First, progression should be determined by working backward from students' expected final outcomes, with course difficulty gradually increasing to guide them toward achievement. Second, emphasis should be placed on significant, foundational, and core outcomes; courses or content deemed less essential can be removed or replaced with more critical elements to more effectively support successful student learning.

By integrating the OBE concept and aligning with the cultivation requirements for computer science professionals under the framework of Emerging Engineering Education, the course design is centered on the knowledge, skills, and literacies students are expected to possess upon graduation. Specifically, these encompass the following:

- 1) Knowledge Objectives: Mastery of fundamental Python language concepts, programming methodologies, and essential algorithms.
- 2) Ability Objectives: Proficiency in Python programming paradigms, the ability to select appropriate AI development environments, and the capacity to solve complex engineering problems within the AI domain.
- 3) Literacy Objectives: Development of professional literacy, teamwork spirit, communication skills, and an awareness of lifelong learning.

The objectives, teaching activities, and assessment items are mapped accordingly, as shown in **Table 1**.

The following section presents three experimental teaching cases designed to ensure that the course objectives effectively support the overarching graduation requirements.

5.1. Understanding Edge Computing Platforms

Edge computing refers to computational processing performed at or near the physical location of users or data sources, enabling localized intelligent data processing services to reduce latency and conserve bandwidth [7]. In December 2024, NVIDIA released the Jetson Orin Nano Super Developer Kit, a compact yet powerful supercomputer that redefines generative AI deployment on small-scale edge

Table 1. Mapping of objectives, teaching activities, and assessment items.

Three-Dimensional Objective Type	Specific Objective Description	Teaching Activities	Assessment Items
Knowledge Objectives	<ol style="list-style-type: none"> 1. Master Python basic syntax and algorithmic logic. 2. Understand hardware characteristics of Jetson Orin Nano. 	<ol style="list-style-type: none"> 1. Theoretical lectures: Python syntax instruction. 2. Laboratory sessions: Edge computing platform setup (Ubuntu system installation, SDK flashing). 	<ol style="list-style-type: none"> 1. Theoretical test (20%). 2. Hardware characteristics description report (15%).
Ability Objectives	<ol style="list-style-type: none"> 1. Independently complete on-device deployment of the DeepSeek-R1 model. 2. Implement human pose estimation using MediaPipe. 	<ol style="list-style-type: none"> 1. Laboratory sessions: Ollama tool configuration and Open WebUI utilization. 2. Laboratory sessions: Python programming (invoking the MediaPipe library). 	<ol style="list-style-type: none"> 1. Practical model deployment evaluation (30%). 2. Human pose estimation program performance evaluation (25%).
Literacy Objectives	<ol style="list-style-type: none"> 1. Complete comprehensive projects through teamwork. 2. Develop an awareness of lifelong learning. 	<ol style="list-style-type: none"> 1. Group project: Collaborative application development integrating edge computing and AI models. 2. Post-class assignment: Writing AI technology course design reports. 	<ol style="list-style-type: none"> 1. Team contribution evaluation (40% peer review + 60% instructor evaluation). 2. Course design report quality evaluation.

computing platforms. Compared to its predecessor, this platform delivers up to 67 INT8 TOPS of AI performance—a 1.7X improvement, and memory bandwidth to 102 GB/s, as illustrated in **Figure 1**.

**Figure 1.** Jetson orin nano super developer kit.

The edge computing platform setup is completed by flashing the OS image using NVIDIA's official SDK Manager 2.3.0 within an Ubuntu 20.04 system installed on VMware Workstation Pro 17 virtual machines. Post-installation, the system interface displays (**Figure 2**):

1) Upper section: Device model, CPU utilization, storage status, fan operation, and power consumption.

Students leveraged Open WebUI, an extensible self-hosted AI platform, for experimental operations via its graphical interface, improving system interactivity (Figure 5).

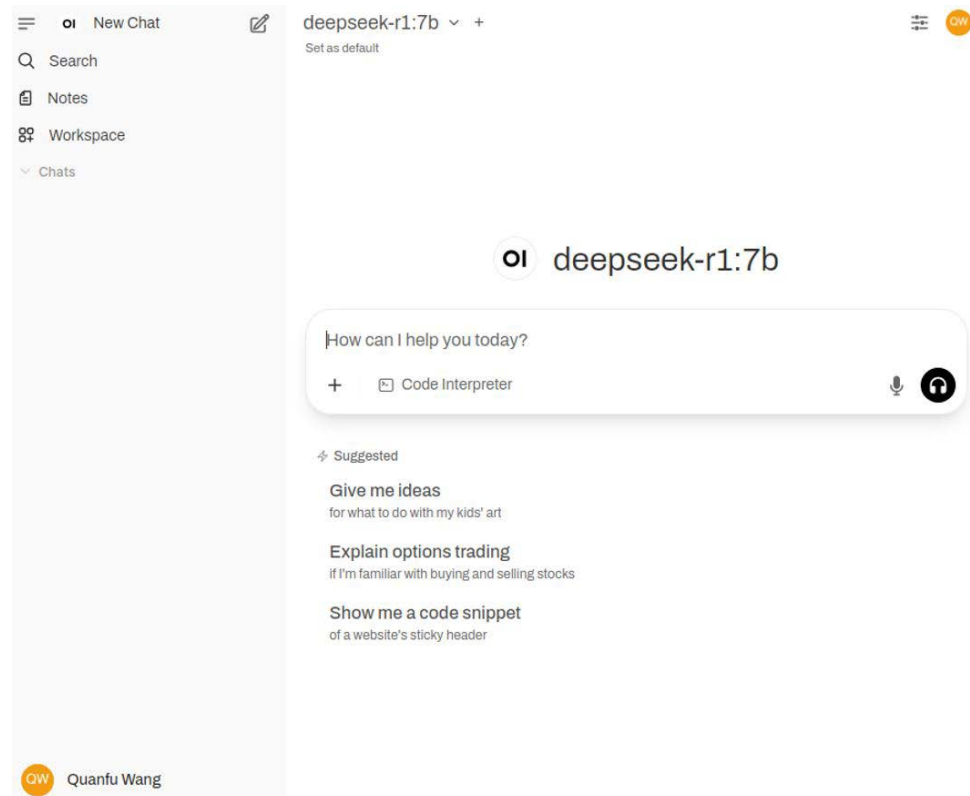


Figure 5. Open WebUI graphical interface.

5.3. MediaPipe Framework Application Experiment

Human pose estimation (also termed body keypoint detection) accurately extracts spatial positions of human joints from images or videos [9]. MediaPipe, an open-source cross-platform framework developed by Google, specializes in building machine learning applications with real-time streaming capabilities. It provides solutions for face recognition, gesture detection, motion tracking, and pose estimation. By integrating a depth camera with the edge platform and implementing experimental Python programs, we achieved real-time human pose estimation functionality (Figure 6).

6. Analysis of Teaching Reform Outcomes

The OBE-based instructional reform of the *Python Programming* course has yielded notable improvements in student competency development, teaching quality, and the optimization of the assessment system. These achievements have been realized through a closed-loop framework that integrates clearly defined outcomes, backward design, diversified assessment, and continuous improvement. The reform's effectiveness is reflected across four key dimensions, as outlined below.

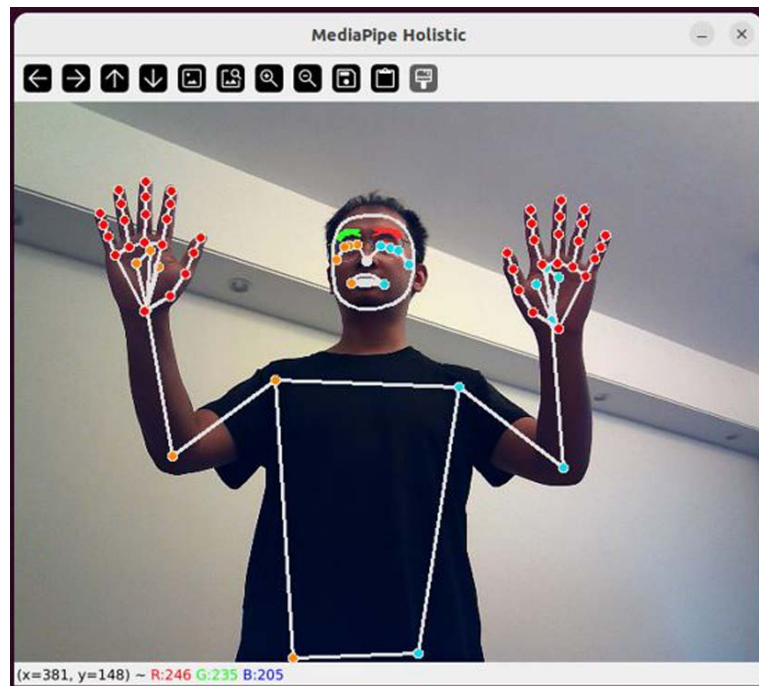


Figure 6. Human pose estimation interface.

6.1. Significant Enhancement in Learning Outcomes and Objective Attainment

The reform systematically reinforced students' core competencies by targeting the attainment of three interrelated objectives: knowledge, skills, and literacies.

1) Knowledge Attainment: Students demonstrated substantially improved mastery of Python fundamentals and algorithmic logic. Their exposure to frontier technologies expanded considerably, particularly edge computing, generative AI, and human pose estimation. A large proportion (92%) of students were able to accurately describe both the hardware specifications of the Jetson Orin Nano Super Developer Kit and the core functionalities of the MediaPipe framework.

2) Skill Development: In the generative AI deployment experiment, 87% of students successfully completed independent installation and configuration of the Ollama toolkit for on-device deployment of the DeepSeek-R1 model. In the MediaPipe framework experiment, 91% of students designed and implemented functional human pose estimation programs, effectively translating programming knowledge into real-world applications.

3) Literacy Cultivation: Through collaborative projects integrating edge computing platforms with AI models, students strengthened their abilities in collaborative problem-solving and task decomposition. Post-project reflections further indicated progress in workload allocation, consensus building, and team-based problem resolution—competencies that directly align with professional collaboration and the principles of lifelong learning.

Before the reform, the completion rate of similar experiments by students in the same major was only 62%. After the reform, the completion rate of core

experiments exceeded 85%, representing an improvement of 37%, which demonstrates the significant impact of the reform on students' ability development.

6.2. Enhanced Alignment of Teaching Methods and Content

The integration of edge computing platforms, Ollama, MediaPipe and related tools has significantly strengthened the course's alignment with emerging industry technologies. Case studies now possess greater timeliness and practical relevance, which has noticeably increased student engagement with interdisciplinary fields at the intersection of Python, AI, and edge computing. The transition from unidirectional knowledge delivery to project-driven pedagogy has further enhanced students' capacity for knowledge transfer. When addressing complex challenges such as selecting and deploying AI models for specific application scenarios, students are now able to design more rational solutions, demonstrating stronger engineering competencies.

6.3. Further Optimization of the Assessment System

The introduction of a diversified assessment framework has effectively mitigated the constraints of traditional exam-oriented evaluation. By adopting a balanced model of "formative assessment (50%) + summative assessment (50%)," the evaluation scope extends beyond theoretical recall to include practical implementation, innovative application, and team-based contributions. Tiered evaluation criteria designed for different stages of learning accommodate individualized development needs more effectively: students at risk of underperformance show improved achievement, while advanced learners demonstrate greater capacity for innovation.

6.4. Initial Formation of a Continuous Improvement Mechanism

The systematic collection of student feedback and industry practitioner input has enabled dynamic optimization of instructional practices. To resolve challenges associated with environment configuration in generative AI deployment experiments, supplementary tutorial cases and pre-configured system images were introduced, leading to higher experimental success rates. In response to industry demand for competencies that integrate "Python + AI + edge computing," a new instructional module on "edge-cloud synergy" was incorporated, further reinforcing the curriculum's alignment with professional requirements.

7. Conclusions

Centering on students' culminating learning outcomes, this reform addresses the limitations of traditional instruction—such as outdated content, unidimensional teaching methods, and fragmented assessment—through backward curriculum design, integration of cutting-edge technical cases, and diversified evaluation mechanisms. These measures collectively advance a fundamental shift from knowledge transmission to competency cultivation. Empirical evidence indicates that

embedding the OBE philosophy into course design substantially enhances students' engineering practice capabilities and innovative thinking, thereby offering a viable pathway for cultivating computer science professionals within the framework of Emerging Engineering Education.

This study has three main limitations. First, the sample source was relatively homogeneous, consisting solely of students from Guangzhou Nanfang College; therefore, the generalizability of the results requires further validation across multiple institutions. Second, the follow-up period was relatively short, tracking only up to one month after the reform concluded, without examining students' actual ability to apply the technology in employment settings. Third, some data (such as teamwork ability) relied on student self-reports, which may contain subjective bias; future research could incorporate evaluations from enterprise mentors for further validation.

Looking ahead, further course enhancements will focus on three priorities:

- 1) Continuously monitoring technological advancements in AI and edge computing to dynamically update teaching case repositories;
- 2) Deepening industry-academia collaboration by embedding authentic engineering projects as pedagogical drivers;
- 3) Refining the course competency matrix to ensure seamless articulation between Python Programming and subsequent modules.

Through iterative optimization, this reform aims to establish a high-quality curriculum system attuned to the demands of the AI era, thereby laying a robust foundation for nurturing the innovative talents urgently needed in national development.

Fund Project

This paper is funded by the university-level projects of Guangzhou Nanfang College (project numbers: 2024XK005) and Association of Fundamental Computing Education in Chinese Universities, AFCEC (project number: 2024-AFCEC-306).

Conflicts of Interest

The authors declare no conflicts of interest.

References

- [1] Communist Party of China (CPC) Central Committee and the State Council (2025) The Master Plan on Building China into a Leading Country in Education (2024-2035). https://english.www.gov.cn/policies/latestreleases/202501/20/c-ontent_WS678d85c6c6d0868f4e8eef83.html
- [2] Ministry of Education of the People's Republic of China (2025) Notice on Organizing and Implementing the Digital Empowerment Initiative for Teacher Development. http://www.moe.gov.cn/srcsite/A10/s7034/202507/t20250704_1196586.html
- [3] Spady, W.D. (1994) Outcome-Based Education: Critical Issues and Answers. American Association of School Administrators.

-
- <https://files.eric.ed.gov/fulltext/ED380910.pdf>
- [4] International Engineering Alliance (2014) The Washington Accord Is an International Agreement between Organisations Responsible for Accrediting Professional Engineering Degree Programmes. <https://www.internationalengineeringalliance.org/accords/washington-accord>
- [5] Bouslama, F., Lansari, A., Mahmoud Al-Rawi, A. and Abonamah, A.A. (2003) A Novel Outcome-Based Educational Model and Its Effect on Student Learning, Curriculum Development, and Assessment. *Journal of Information Technology Education: Research*, **2**, 203-214. <https://doi.org/10.28945/323>
- [6] Kaliannan, M. and Chandran, S.D. (2012) Empowering Students through Outcome-Based Education (OBE). *Research in Education*, **87**, 50-63. <https://doi.org/10.7227/rie.87.1.4>
- [7] Wang, X., Han, Y., Leung, V.C.M., Niyato, D., Yan, X. and Chen, X. (2020) Convergence of Edge Computing and Deep Learning: A Comprehensive Survey. *IEEE Communications Surveys & Tutorials*, **22**, 869-904. <https://doi.org/10.1109/comst.2020.2970550>
- [8] Zhang, H. (2025) How Deepseek-R1 Was Created? *Journal of Shenzhen University Science and Engineering*, **42**, 226-232. <https://doi.org/10.3724/sp.j.1249.2025.02226>
- [9] Sigal, L. (2021) Human Pose Estimation. In: Sigal, L., Ed., *Computer Vision*, Springer International Publishing, 1-20. https://doi.org/10.1007/978-3-030-03243-2_584-1