

# REFORMING THE TRAINING MODE OF THE “PRINCIPLES OF COMPUTER ORGANISATION” COURSE THROUGH MULTIPLE MEASURES

DuanLi Yang\*, Yan Zhang, YuanLi Wang

*College of Information Science and Technology, Hebei Agricultural University, Baoding 071001, Hebei, China.*

*\*Corresponding Author: DuanLi Yang*

**Abstract:** With the goal of building a high-quality “Computer Organization” course, the course team has adopted multiple approaches to enhance teachers’ ability to teach and educate students, and to reform the talent cultivation model. These approaches include improving teachers’ professional competence, promoting engineering education accreditation, adopting a research-oriented teaching model, fostering students’ system development ability, designing innovative experiments, implementing online and offline blended teaching, integrating ideological and political education into the classroom, and reforming assessment methods. This paper describes in detail the implementation process and our reflections on these approaches, and shares them with fellow educators, in the hope of further improving course development and benefiting more students.

**Keywords:** First-class courses; Computer organization and architecture; Teaching reform

## 1 INTRODUCTION

The high-level construction of courses is the key to cultivating high-quality talents. The course of Computer Organization Principles is a national unified examination course for computer-related majors and an important course for related majors such as Computer Science and Technology, Electronic Information Science and Technology, Internet of Things Engineering, Network Engineering, Software Engineering, and Big Data Science and Technology[1]. The most prominent feature of this course is its complex concepts and strong theoretical nature[2]. Although the course team has spent a lot of energy reforming the talent training model over the long period of course construction, it has always been unable to get rid of the pain points of rigid teaching content, insufficient cultivation of innovative hands-on ability, and single assessment method in the training model[3]. The indoctrination-style teaching and the score-based evaluation have resulted in insufficient innovation motivation among students and difficulty in fundamentally improving the teaching quality. To meet the requirements of the "13th Five-Year Plan for National Education Development" and improve teaching quality, this course, tailored to its characteristics, aims to break through old teaching models, explore new methods, and enhance learning outcomes. Addressing relevant pain points, reforming the talent cultivation model hinges on research and practice regarding teachers' teaching and nurturing abilities. This involves reforming teaching methods and students' learning styles, promoting a balance between teaching and learning, shifting learning from "passive" to "active," constructing new teacher-student relationships, knowledge systems, and teaching formats, and cultivating students' lifelong learning abilities. All teachers in the course team conduct teaching summaries at the end of each semester and develop new teaching plans at the beginning of each semester. They have actively reformed and experimented with various teaching methods and approaches, achieving some results. Below, the course team discusses some of the gains since the course's development in eight aspects and takes this opportunity to exchange experiences and shortcomings with colleagues, hoping to further improve and promote a model beneficial to the mutual growth of teachers and students.

## 2 TEACHERS' PROFESSIONAL COMPETENCE IMPROVEMENT

Although the course instructors all have many years of experience teaching "Computer Organization and Architecture" and are very familiar with the knowledge points covered in the course, computer technology is a constantly evolving industry with new technologies and knowledge emerging continuously. Only through continuous learning can advanced knowledge be introduced and students' horizons broadened. With the progress of the times, new methods and approaches are constantly emerging. In order to further expand the teaching content in both depth and breadth, and to introduce advanced teaching concepts and methods into the teaching of "Computer Organization and Architecture," the course instructors used their holidays to study online courses offered by several well-known universities. At the same time, in order to build a first-class course, they also carefully studied hundreds of teaching-related papers. Through this learning, the course team's biggest gains are as follows:

Teaching should emphasize both instruction and learning.

The teaching process should be balanced, with both relaxed and focused instruction. Students should be given ample trust and affirmation to ensure they feel involved.

Emphasizing both teaching and learning, and involving students in the entire teaching process, is a distinctive feature of

the "Computer Organization and Architecture" course at these renowned universities.

In the "Computer Organization and Architecture" courses at these prestigious universities, practical components are a key focus of the teaching. Table 1 shows the arrangements for practical components at various renowned universities.

**Table 1** Practical Training Arrangements at Various Renowned Universities

College	Practical requirements
Tsinghua University	The course clearly states its learning objective: to enable students to independently design a complete computer through the course.
Harbin Institute of Technology	The course includes a "Computer Design and Practice" component, which requires students to design computer systems from the lowest-level hardware to the highest-level instructions.
Huazhong University of Science and Technology	The course includes a "Computer Hardware System Design" component, which requires students to start from the most basic logic gates and gradually complete a computer system.
Southeast University	The course "Computer System Design" requires students to have a deep understanding of the underlying hardware structure, design systems, and optimize system performance.

The practical component requires students to have a solid grasp of theoretical knowledge. It not only improves their practical and engineering application abilities, but also cultivates their self-learning, independent thinking, information retrieval, teamwork, and initiative abilities in the process of designing systems and solving problems. It also enhances their self-confidence and stimulates their innovative awareness.

Practical teaching employs real hardware design methods, allowing students to apply theoretical knowledge to practical situations. This fosters a sense of satisfaction and confidence from the results, helping to stimulate students' interest in learning. However, due to the long development cycle, the depth and breadth of knowledge involved, the complexity of the content, and the need for a strong foundation of prior knowledge, the practical implementation process is quite challenging. But we cannot give up because of these difficulties. After discussion, the course team decided to adopt a "step-by-step – focusing on key points – multi-level" strategy. "Step-by-step" means dividing practical teaching into several stages; "focusing on key points" means designing key modules; and "multi-level" refers to the progressively increasing difficulty and reusability of the results.

The specific implementation process is as follows:

Phase 1: "Subtle Influence" --- Integrating practical design concepts into teaching, introducing some ideas and knowledge involved in the design implementation process into the classroom, and guiding students to conduct analysis and simple design.

Phase 2: "Exemplary Leadership" --- Organize a group of outstanding students to carry out design practice. Through the exemplary role of these students, eliminate the fear of difficulties of other students and inspire their self-confidence. At the same time, these students can also serve as the teacher's capable assistants to help guide other students.

Phase 3: "Summary and Adjustment" --- Summarize the achievements of this phase and further improve the course. In this phase, the course team teachers conduct detailed exchanges and summaries of the work carried out in the early stage and formulate a plan for large-scale promotion.

Phase 4: "Promotion and Implementation" --- Large-scale implementation phase, organizing students to carry out practical teaching based on the experience accumulated in the early stage, designing and implementing a whole from the underlying hardware to the upper-level machine instruction execution.

### 3 CURRICULUM DEVELOPMENT BASED ON ENGINEERING EDUCATION ACCREDITATION

The professional certification of engineering education emphasizes the output orientation of teaching objectives, and the training objectives should clearly indicate the requirements that students need to meet upon graduation[4]. The requirements for talent training are reflected in the three dimensions of knowledge + ability + innovation. Based on this, the course team has reorganized the course content and formulated more appropriate teaching objectives, content, requirements, methods and assessment methods.

The course objectives and specific content are shown in Table 2:

**Table 2** Course Objectives and Specific Content

Target	Specific content
Theoretical Section Teaching Objectives Objective 1: Master the systematic knowledge and apply it to the reasoning and analysis process of complex engineering problems in the application field.	<ul style="list-style-type: none"> <li>➤ Master the calculation methods, understand the nature and judgment of overflow, master the hardware operation theory, and be able to analyze system performance and select data types and calculation methods in development;</li> <li>➤ Having a grasp of the complex and diverse range of storage devices, one can select the right one based on performance indicators;</li> <li>➤ Master expansion circuits; be able to analyze systems based on capacity requirements, select chips, determine decoding, and design circuits; be able to design and analyze main memory and cache address mappings.</li> <li>➤ Master the concepts of instruction cycle and microprogram, and</li> </ul>

Experimental Section Teaching Objectives	Objective 2: To be able to comprehensively utilize theoretical and technical means, consider user needs from a system perspective, and provide overall solutions to complex engineering problems in the field of computer applications.	<p>be able to design microinstructions.</p> <ul style="list-style-type: none"> <li>● In software and hardware design, we can consider user needs from a system perspective, select appropriate data representation methods, design software programs and hardware systems that satisfy the theory of locality, arrange program instructions reasonably, reduce pipeline-related problems, avoid system efficiency degradation caused by flow interruption, optimize variable definitions, reduce memory access processes, and save execution time.             <ul style="list-style-type: none"> <li>➢ Master the combinational functions of the 4-bit 74LS181</li> <li>➢ Master the functions, pinout function symbols, and pin assignments of the 8-bit 74LS299.                 <ul style="list-style-type: none"> <li>➢ Master the functions of SRAM 6116</li> </ul> </li> </ul> </li> <li>➢ Master the composition, principles, and operating mechanisms of microprogrammed controller hardware, as well as the dynamic working process of the entire machine, and be able to apply this knowledge to the analysis of complex engineering problems in application fields.</li> <li>● Design an arithmetic unit operation experiment with overflow results; design an arithmetic unit experiment that can perform multiplication and division operations using the 74LS299 chip; design an experiment to write and read data into unused space of the 6116 memory; be able to design experimental schemes, solve problems, and complete experiments.</li> </ul>
	Objective 1: To master computer systems knowledge and be able to apply it to the reasoning and analysis processes of complex engineering problems in computer application fields.	
	Objective 2: To be able to design experimental schemes, complete experiments, and solve experimental problems using scientific methods.	

#### Course teaching methods:

(1) Theoretical Course Objective 1: The teaching focus is on mastering the structure and working principles of the five major computer hardware components. The course employs methods such as in-depth classroom lectures, physical demonstrations, and multimedia courseware animation simulations to stimulate students' learning interest and enable them to master fundamental knowledge of computer system architecture, including the arithmetic unit's operation process, memory read/write process, CPU instruction fetching and execution process, and bus systems. Research-based project cases are assigned, and through case analysis and discussion, students are guided to master relevant knowledge and methods, and how to apply this knowledge to system design and development. Students are required to collect relevant resources using information technology outside of class, and the results are assessed through assignments to achieve the requirements of Course Objective 1.

(2) Course Objective 2 for Theoretical Courses: For topics such as the principle of locality, the principle of pipeline interruption, memory data read/write rules, and the implementation principle of cache memory, the lectures will focus on explaining from a systemic perspective how the composition and working principles of computer hardware constrain program execution efficiency, and how programs can be designed to be consistent with the computer hardware structure to improve code quality. The lectures will also incorporate application cases of considerable complexity to conduct functional analysis, enabling students to understand how to apply theoretical knowledge to solve practical problems. Appropriately challenging practice problems will be assigned, requiring students to independently research relevant materials, solve problems through self-study, and complete assignments for assessment, thus achieving the requirements of Course Objective 2.

(3) Experimental Course Objective 1: Experiments are conducted in groups of 1 to 3 students. Students are required to independently complete the relevant experiments according to the experimental objectives. Through the experiments, students will master relevant theoretical knowledge and cultivate good habits in hardware experimental operation, providing knowledge reserves for the development of application systems. For complex comprehensive experiments, students are required to integrate theory into the design and develop their ability to learn independently, think critically, discover, analyze, and solve problems.

(4) Course Objective 2 for Experimental Class: The experimental class adopts a combination of centralized explanation and individual guidance. Common problems discovered in the experiment are explained in class, while individual questions raised by students are addressed with targeted guidance. During the experiment, students are guided to pay attention to the details and are guided to explore solutions, thus cultivating their initiative in learning.

Course assessment and evaluation methods: (1) Theoretical courses: The learning outcomes of the course are judged based on the achievement of the course objectives and whether the relevant ideas and methods in the course can be used to solve engineering practice problems.

Course assessments consist of regular assessments and a final assessment, with a weighting of 3:7.

Regular assessment: Regular assessment focuses on process and periodic evaluation, with classroom performance and homework accounting for 4:6. Classroom performance is assessed through in-class quizzes by the teacher, and grades are given based on the test results. Homework is assigned after each (or several) learning units, with a time limit for completion, and grades are given based on the quality of completion. Students are required to take at least 5 self-tests after class.

Final exam: Based on the course objectives, a variety of questions will be designed to assess students' understanding of the five major components of computer hardware and the concepts and structures of buses. The exam will focus on

students' mastery of relevant theoretical knowledge and their ability to solve related engineering problems. Each exam will have a corresponding grading standard designed based on the exam questions. The final exam will be closed-book, with the written exam score out of 100, weighted at 0.7 and included in the final overall grade, as shown in Table 3.

**Table 3** Assessment and Evaluation Methods for Achieving Course Objectives and Comparison of Grades

Course Objectives	Assessment Content	Assessment process			Grade ratio /A*30%+B*70%
		Regular assessment / A (30%)		Final Exam /B(70%)	
		A = a1 * 40% + a2 * 60%	Classroom grades /a1(40%)	Homework /a2(60%)	
Target 1	Data representation and operation methods in computers, overflow principles and judgment methods, working principle of arithmetic unit, working principle of memory, word extension of memory, address mapping of main memory and cache, microprogram related concepts and control systems.	60	60	60	60
Target 2	Data format, instruction representation and addressing modes, memory working principle, locality principle, pipeline working principle, and the entire instruction cycle operation process from instruction fetching to instruction execution by the controller.	40	40	40	40

**Experimental Class:**

The experimental score consists of two parts: the experimental results and the experimental report.

Experiment results were verified on-site, and students were required to present their findings in response to various questions. All experiments included thought-provoking and in-depth inquiry-based questions, requiring students to carefully explore and discuss the results after class.

The experimental indicators are shown in Table 4:

The evaluation criteria for experimental grades consist of the standardization of experimental procedures, the correctness of results, and the group presentations in each experimental class. The scoring indicators for experimental grades include: (1) standardization of experimental procedures; (2) correctness of experimental results; and (3) correct answers to relevant questions during the presentations.

**Table 4** Evaluation Criteria for Experimental Results

Score	Evaluation Indicators
90-100	All indicators at all levels met the standards.
80-89	Indicator (1) is not up to standard, while other indicators are up to standard.
70-79	Indicator (3) is not up to standard, while other indicators are up to standard.
60-69	Indicator (2) is not up to standard, while other indicators are up to standard.
Less than 60 points	Not meeting the above criteria

The evaluation index points for the experimental report are shown in Table 5:

The experimental report is evaluated based on indicators such as completeness, accuracy, and standardization. The evaluation criteria for the experimental report include: (1) the experimental report has a complete structure and meets the writing requirements; (2) the experimental results are complete; and (3) the summary is substantial. Among them, indicator (2) is the key indicator, and (1) and (3) are non-key indicators. The score is evaluated based on the above indicators.

**Table 5** Evaluation Criteria for Experiment Reports

Score	Evaluation Indicators
90-100	All indicators at all levels met the standards.
80-89	Key indicators met the standards, but non-key indicators (1) failed to meet the standards.
70-79	Key indicators met the standards, but non-key indicators (3) did not meet the standards.
60-69	Both key and non-key indicators failed to meet the standards.
Less than 60 points	Not meeting the above criteria

**4 IMPLEMENTATION OF INQUIRY-BASED TEACHING MODEL**

The research-based teaching model can change the traditional indoctrination teaching to project-based inquiry teaching, establish the teaching concept of "student-centered", and stimulate students' learning initiative[5]. The course group adopts innovative project-driven teaching and sets up different projects and requirements according to students with different abilities. The projects are mainly divided into two categories: in-class research topics and open large projects after class. The in-class research topics set up 3 to 4 comprehensive training projects of moderate difficulty. These

projects include both basic knowledge points of the course and cutting-edge knowledge. Students plan their time independently, consult relevant literature, cooperate in groups, analyze and design projects, determine project development plans, and finally submit project results. Teachers and students then conduct discussions. The formulation of these projects is based on the principle that students can master the basic knowledge of the course while improving their academic research ability. The projects take students' in-depth understanding of relevant classroom theoretical knowledge as the basic requirement, while also cultivating their ability to learn independently, analyze problems, cooperate in teams and innovate. The four main in-class projects are shown in Table 6:

**Table 6** In-class Research Projects

Project Content
Project 1: Using two 74181 chips, design and implement an arithmetic logic unit (ALU) capable of performing 16-bit binary data operations. This topic mainly covers parallel carry, function control signals, and timing concepts.
Project 2: Memory Design. This project focuses on designing memory spaces that meet address allocation requirements using SRAM and ROM. Key topics include chip select signal generation circuit design, address line and data line circuit design, and read/write control signal design.
Project 3: The Relationship Between Programs, Instruction Cycles, and Microprograms. Microprogrammed controller and microprogram design: Based on the given arithmetic logic unit (ALU), controller, memory hardware circuitry, microinstruction format, and instruction cycle flowchart, design and implement microinstructions.
Project 4: Design several machine instructions, and based on the computer hardware architecture and microinstruction format, write the corresponding microinstruction sequence, and debug it on the computer. This topic mainly covers the following knowledge points: hardware and software relationships, and the low-level execution process of machine instructions.

The implementation of in-class projects enhances learning interest and fosters a sense of participation, self-affirmation, and happiness. Students have been learning in the traditional teacher-lecturing, student-listening style for over a decade. Even a small change in teaching methods can bring unexpected results, bringing surprises and satisfaction to teachers. They sincerely marvel at how capable their students can be by giving them the freedom to act. Through teaching students from the School of Information Science and Technology and the School of Modern Science and Technology over the past few years, it has been found that students from the School of Modern Science and Technology have a stronger willingness to participate, more active thinking, and produce the most impressive results. Some students not only presented relevant knowledge but also used that knowledge to introduce the history and development of related industries, analyzing employment prospects, salaries, and potential employers from an employment perspective. They also offered career advice to their classmates, suggesting which courses to focus on, what extracurricular learning to do, and which industry and company recruitment information to pay attention to. Their clear and logical thinking is truly admirable.

of conducting after-school large-scale projects is to further guide capable and interested students through practical projects, and to further stimulate the integration of theoretical knowledge and independent innovation ability through participation in project development. The main after-school large-scale projects are shown in Table 7:

**Table 7** Post-Class Research Projects

Project Content
Project 1: Design and Development of a Three-Tier On-Site Inspection-Review-Approval System
Project 2: Remote Monitoring System for Greenhouse Fertilization
Project 3: Remote Real-time Monitoring System for Aquaculture Environment.
Project 4: IoT-based Smart Health Remote Monitoring and Early Warning System
Project 5: Intelligent Monitoring, Early Warning and Control System for Chicken Coop Environment
Project 6: Indoor Fire Alarm System
Project 7: Design and Implementation of a Precise Hazard Detection System Based on Intelligent Algorithms

These projects enable students to not only learn knowledge relevant to their course but also actively explore other related knowledge, broadening their horizons and expanding the breadth and depth of their knowledge. During project implementation, teachers should maintain an overall grasp of the project, communicate frequently with students, and guide them in analyzing difficulties and developing solutions. In this model, the teacher's role is more like that of a project leader, providing guidance, leading students into the project, helping them determine methods, and guiding their exploration. Throughout the project, students and teachers shift their original roles to become project leaders and members. Through project practice, both teachers and students change their original roles, facilitating smoother communication and enhancing students' exploratory and practical spirit.

The biggest takeaway from the after-school projects was that some students' learning and development potential and abilities, as well as their interpersonal and communication skills, exceeded teachers' expectations, demonstrating high IQ and EQ. Initially, it was hoped that most students would participate in these projects, but it was found that students had different self-assessments, abilities, and expectations. Therefore, only a small number of students actively participated in the project development and persevered. It was precisely these students' efforts that yielded fruitful results. Several students in the project group were admitted to Beijing Jiaotong University through the recommendation system, several were admitted to prestigious graduate programs, and several found their ideal jobs. The students admitted through the recommendation system not only had excellent academic performance but also received bonuses

due to their project participation and awards. Those admitted to graduate programs gained an advantage in interviews due to their practical project experience. Many students also gained knowledge and experience through project participation and found their ideal jobs. Although only a small number of students persisted with the after-school projects, this may have a significant impact on their future life trajectories and is of extraordinary significance to their families. Meanwhile, some projects also feature continuous development, which subtly fosters mentorship and a sustained positive impact among students. For example, the chicken coop environmental monitoring system project initially developed only a monitoring system, then a monitoring and control system, and later a control system with harmful gas monitoring. Currently, an intelligent algorithm-based chicken coop harmful gas detection system is under development. This continuous project allows previous students to act as mentors for the next batch of students, encouraging the former and eliminating any sense of apprehension for the latter, resulting in significant achievements.

1、Systematic competency development and a shift in teaching philosophy

The design of the knowledge system of the course "Computer Organization and Architecture" is mostly based on the perspective of computer designers rather than the perspective of users[6] However, for most students, after graduation, they often engage in software development work rather than professional CPU hardware designers or operating system development engineers. It is precisely for this reason that some students are always confused about the value and significance of learning this course. In order to make students understand the necessity of opening the course, the course team has reorganized the course knowledge and introduced the idea of system view into the teaching process, so that students can realize that as a system composed of two major parts, software and hardware, the software must ultimately be analyzed and executed by the real hardware. Only by having a full understanding of the composition and working principle of computer hardware, and writing software that is most consistent with the hardware structure, can high-quality code be written, and become an expert in code maintenance [7-8], low-level indicator satisfaction, code compatibility and universality. It is also based on the idea of system view that the course team has introduced the teaching concept of teaching computer hardware composition from the perspective of programmers, and integrated the relevant knowledge of courses such as C language, operating system, and compiler principles into the teaching, so that students can have a deeper understanding of the relevant knowledge of the prerequisite courses from the lowest hardware level of the computer, and lay the foundation for subsequent courses. Students reported that teaching relevant knowledge from a programmer's perspective not only made the course content easier to understand but also made it more practical. Students could verify relevant theories through programs, easily build a complete knowledge system and overall knowledge framework for computer systems, and acquire the ability to write high-performance programs. The relevant knowledge points in other courses are shown in Table 8.

**Table 8** Knowledge Points Related to Other Courses

Other courses	Related knowledge points
Operating System Course	The CPU supports the execution of the OS, the interrupt hardware mechanism supports the operation of OS interrupt handlers, I/O devices support the interaction between the OS and the user, and storage devices provide the storage environment for the OS.
C Programming Course	Why define various data types? How are arithmetic and logical operations implemented? How does the principle of locality affect execution efficiency? What is the relationship between pointers and memory? What is the relationship between statements and instructions?
Compiler Principles Course	The functions of the compiler, how the compiler compiles variables, function calls, data-related and control-related processing, and static and dynamic pipeline scheduling are all related to compiler optimization.

**5 INNOVATIVE EXPERIMENTS CONDUCTED**

Traditional experiments are primarily verification-based, where students conduct experiments using model machines by manually connecting wires and employing system keypad input devices or binary toggle switches. While verification experiments are easy to operate and master, they lack innovation, hindering students' critical thinking and preventing them from deeply understanding the system's internal workings, thus limiting opportunities for independent innovation and development. To stimulate students' creativity, the teaching team has independently developed unique experimental projects that students can complete according to their abilities, building their confidence and stimulating their interest in learning.

Featured experimental projects include:

(1) Directly controlling output devices using switch input devices: This experiment allows students to intuitively experience how hardware is controlled, how inputs and outputs are connected via a bus, the function of control signals, the function of the bus, timing, and other concepts. This is a basic experiment that all students are required to complete. Although the experiment is simple, the results are intuitive, allowing students to not only learn knowledge but also experience the fun of hardware experiments. Furthermore, each cohort of students implements the experimental content in various different ways, which amazes the course instructors.

(2) Model computer composition principle experiment: Assemble a complete computer, design instructions such as ADD, SUB, LOAD, MOV, STO, JMP, etc., design microprogram flow according to instruction cycle, and debug on the computer. This experiment involves many concepts and requires complex content, which is very easy to make mistakes. Therefore, students are required to complete some or all of the experiments according to their actual situation.

Due to differences in basic knowledge and varying learning abilities among students, patience and timely encouragement are essential for those who struggle. With teacher guidance, almost all students can achieve the desired

results and actively act as peer instructors, helping others. This encourages the former and prevents the latter from feeling intimidated. Timely praise and affirmation are crucial. After completing experiments, teachers should provide timely encouragement and recognition, allowing each student to experience a sense of accomplishment, self-affirmation, and increased confidence, thus stimulating their interest in conducting further experiments.

The benefits of implementing innovative experiments: Praise and recognition are crucial, as they more easily stimulate learning interest. After class, students are often seen spending their free time in the lab, truly working tirelessly, arriving early and leaving late, never leaving until the lab is closed. They even schedule weekend experiments with teachers, fostering a positive learning atmosphere where students compete and learn from each other. The implementation of innovative experimental teaching has given me many insights. As teachers, we must be adept at discovering students' potential, acting as their mentors; genuine dedication will reap rewards.

## **6 BLENDED LEARNING**

By making full use of Rain Classroom to carry out blended online and offline teaching, online attendance, sharing of courseware, assignments, experimental content, and discussion content can be achieved. Students can make full use of high-quality resources for self-study and communicate with teachers at any time through the network, truly realizing interaction anytime and anywhere.

## **7 IDEOLOGICAL AND POLITICAL EDUCATION ENTERS THE CLASSROOM**

Integrating ideological and political education into the classroom not only cultivates correct thinking methods but also guides the establishment of a correct worldview, values, and outlook on life. From the relationship between computer hardware and software to the dependence of software on hardware, and how to optimize software from a low-level perspective when writing high-quality code, students are inspired to understand computer composition from a systems perspective, cultivating their ability to analyze and solve problems from a macro and systems-wide perspective. Through the design of the five major components of computer hardware and the instruction system, students grasp the concept of hardware-software equivalence, learn how to optimize system performance while fully considering cost-effectiveness, and develop their ability to apply dialectical materialism. Using malicious software and viruses in computers to warn students and guide them to establish correct values. Through ideological and political cases such as the Dawning Galaxy supercomputer, the debate over the preservation of Chinese characters, and the construction of the Nanjing Yangtze River Bridge, students' sense of national responsibility and patriotism is guided.

## **8 Reform Of Assessment Methods**

Assessment methods are an important part of evaluating teaching effectiveness and are also a crucial component of first-class courses. Relying solely on written exams for theoretical courses has its limitations. To provide a more accurate assessment of teaching quality and student performance, within the framework of engineering education accreditation course assessment methods, for the final assessment of the theoretical portion of the course, we adopted a combination of oral defense and written examination, considering the fundamental nature of Computer Organization Principles, which involves a great deal of theoretical understanding. This approach eliminates the phenomenon of students rote memorization before exams without focusing on comprehension and in-depth mastery. The theoretical knowledge points covered in the defense are distributed to students in advance. During the assessment, questions are randomly selected, and students answer them on the spot. Because students are unaware of their specific defense topics, all questions must be mastered, preventing the previous practice of students asking teachers to highlight key points before exams, emphasizing rote memorization over understanding. Furthermore, this novel and flexible approach stimulates student interest, fostering a strong learning atmosphere of mutual discussion and questioning before the exam. Students generally reported significant gains from the experience. Because this course also involves some computational and design-related content, the final assessment, in addition to a defense format to test the underlying principles, also includes computational problems to objectively assess students' calculation and design abilities. Lab grades are awarded throughout the entire course, emphasizing the completion of each experiment to avoid the difficulty in accurately assessing the final practical exam, which is susceptible to subjective influences.

## **9 CONCLUSION**

Improving teachers' teaching and nurturing abilities and reforming talent cultivation models hinges on reforming teachers' teaching methods and students' learning styles, promoting a shift towards "active learning." Currently, the Computer Organization Principles course group has achieved some success. The group has meticulously reviewed and optimized the teaching content, introduced various teaching methods, and received positive feedback from both teachers and students, achieving certain results and providing experience for colleagues in developing other courses.

## **COMPETING INTERESTS**

The authors have no relevant financial or non-financial interests to disclose.

## REFERENCES

- [1] Tan Zhihu. Computer Organization Principles. Beijing: Posts & Telecom Press, 2021.
- [2] Tang Shuofei. Computer Organization and Architecture (2nd Edition). Beijing: Higher Education Press, 2015.
- [3] Sun Baodan, Jiang Junhui, Lin Dan. Exploration and practice of teaching "Computer Organization and Architecture" course under the background of new engineering disciplines. Education and Teaching Forum, 2026(02): 1-4.
- [4] Liu Hao, Qian Zhenjiang, Xie Conghua. Research on blended teaching reform of "Computer Organization and Architecture" course. Education and Teaching Forum, 2025(44): 132-135.
- [5] Liu Junjie, Wang Junbo, Song Qiang, et al. Blended teaching reform and practice of Computer Organization Principles course. Electronic Quality, 2025(04): 119-124.
- [6] Liu Caihua. Exploration of Teaching Reform of Computer Organization Principles Course Based on OBE Concept. Western Quality Education, 2024, 10(13): 123-127.
- [7] Stanley T. A student-centered approach to teaching computer organization and architecture. Journal of Computing Sciences in Colleges, 2021, 36(5): 72-80.
- [8] Manjikian, N. Integration of FPGA-based computer organization and architecture courses into a four-year computer engineering curriculum // Proceedings of the 2013 IEEE International Conference on Microelectronic Systems Education (MSE). IEEE, 2013: 45-48.