Improving Higher Education Course Cognitive Outcomes with a Test-Driven Design Strategy for Instruction

Yongbin Zhang[1], Xiuli Fu[2,*], Li Wei[3], and Chunmei Wang[4]

[1] Department of Mechanical and Electrical Engineering, Beijing Institute of Graphic Communication, China
[2] Department of Information Engineering, Beijing Institute of Petrochemical Technology, China
[3] Department of Economic Management, Zhoukou Normal University, China
[4] School of Journalism and Communication, Beijing Institute of Graphic Communication, China
Email: zhangyongbin@bigc.edu.cn (Y.Z.); fuxiuli@bipt.edu.cn (X.F.); ailili152@163.com (L.W.); wangchunmei@bigc.edu.cn (C.W.)
*Corresponding author

***Abstract*—Course learning outcomes are essential parts of program outcomes in higher education. Improving course learning outcomes is a challenging and ongoing task for instructors. This paper aims to improve course cognitive learning outcomes with a novel instruction design model in higher education. A test-driven instruction design strategy is proposed based on learning science theory and the Outcomes-Based Education (OBE) paradigm. With the test-driven design strategy, an instructor will design test cases for each class or lecture. At the beginning of a lecture, students will take a test. Then the instructor and students have other learning activities. Before the end of the lecture, the students will take the test again. After that, the instructor and the students will discuss the content and solve the problems according to the test results. This test-driven instructional design strategy is aligned with the science of learning and learning theories. The strategy explicitly informs students about learning objectives and reminds teachers to design activities according to learning objectives. The assessment helps instructors and students understand the cognitive outcomes of the classroom. It also helps teachers focus on learning outcomes and understand the challenges that students face. Our empirical study shows that students achieved better course learning outcomes with the test-driven design strategy than with traditional ones.**

***Keywords*—cognitive domain, course outcomes, instructional design, test-driven design, higher education**

I. Introduction

The Outcome-Based Education (OBE) paradigm has attracted global attention from universities and educators. OBE emphasizes the success of each student and student-centered learning, which meets the needs of all stakeholders, including students and their parents, industries, and governments. Spady advocates that outcomes are high-quality demonstrations of what learners can do after the completion of learning [1]. OBE has been widely used in higher education, such as designing curricula [2], evaluating program outcomes [3], and assessing course outcomes [4]. These studies are important because course outcomes are essential components of program outcomes. A course is composed of a series of lectures or classes. Classroom outcomes sustain course outcomes. However, there is a lack of research on classroom learning outcomes. There is a gap between the outcomes of a course and the outcomes of the classrooms. In this article, we present a strategy to improve classroom outcomes. With improved classroom results, students will achieve the expected outcomes of the course more easily, which will facilitate students to achieve program outcomes. A process is provided to demonstrate how to implement our Test-Driven Design (TDD) strategy. The TDD model is applied in classrooms of C programming courses in the spring semester of 2021–2022 at the Beijing Institute of Petrochemical Technology. The results show that the group achieved better course learning outcomes with the TDD model than with traditional methods.

Manuscript received May 1, 2023; revised June 13, 2023; accepted July 26, 2023; published January 24, 2024.

II. Review of Literature

A. Learning Outcomes

Learning outcomes have attracted more attention recently. In higher education, the educational paradigm changes from teacher-centered to student-centered. The emphasis on education also changes from how instructors teach to what students achieve. The idea of Outcome-Based Education (OBE) is supported by various levels of organizations, including governments and institutions. However, there is a lack of a uniform definition of learning outcomes. Different terms, such as learning objectives and learning outcomes, are used interchangeably by some researchers [5]. In this article, we define learning outcomes as what a learner achieves after a learning experience. This definition is aligned with the concept of culminating demonstration in the OBE paradigm [6]. Furthermore, we advocate that learning objectives and learning outcomes are related but not the same. We define learning objectives as expected learning outcomes. The distinction makes it clear that a learner may or may not achieve the expected learning outcome. Therefore, it is important to clarify expected learning outcomes. It is also important to assess whether learners achieve the expected learning outcomes.

There are various models to classify learning outcomes. Some divide learning into three domains, cognitive, affective, and psychomotor [7]. For each domain, there are different levels from low to high. For example, the revised Bloom taxonomy for the cognitive domain includes 6 levels [8]. Some classify learning outcomes into program, program-specific, and course outcomes [9].

In this article, we focus on the cognitive domain. Currently, while there are plenty of articles on course learning outcomes, there is a lack of research on how to improve course outcomes through classroom instruction. Improving learning in a classroom is a question that teachers around the world care about [10]. Therefore, we focus on how to improve classroom learning with a concrete strategy.

B. *Science of Learning*

The science of learning investigates and explains how people learn. Progress has been achieved in this field over the past decade. Related theories and principles of learning not only explain what happens during learning activities but also guide how teachers instruct. Susan, Michael, and Marsha present seven principles for instruction based on the science of learning, and one of them is that objective-guided learning practice combined with effective feedback improves learning [11]. The principles of learning science have been adopted in education. For example, researchers provide a framework for applications designed to enhance learning [12].

Additionally, the science of learning is evolving and new theories or principles will be provided. For example, the achievement of neuron science and other different fields facilitates the creation of a new science [13].

Research accomplishment in the science of learning provides a sustainable theoretical foundation for developing novel and creative instructional design models.

C. *Instructional Design*

During the past 50 years, a variety of different instructional design models have been developed Research on the instructional design began with the requirement of training military service during the Second World War [14]. Instructional design is the creation of instructional materials consistently and reliably [15]. To improve learning and performance, instructional design involves the life cycle of instruction, such as analysis, design, development, implementation, evaluation, and management [16]. The ADDIE mode, one of the well-known instructional design models, consists of five stages: analysis, design, development, implementation, and evaluation [17]. This simple general model has been widely used in the instructional design of courses and programs [18].

Another widely used model in curriculum design is the UbD framework. The Understanding by Design framework (UbD) represents a process for the design of courses, assessments, and instruction [19]. Similarly to the OBE paradigm, the UbD framework emphasizes 'the ends' and employs the backward approach to design a curriculum based on the expected learning outcomes.

Although the number of instructional design models is increasing, some issues are still common among models. More than four decades ago, Andrews and Goodson found that most models were not based on learning theories and there was a lack of documents explaining model applications after the review of instructional design models [20]. Some issues of instructional design models are still not resolved [21].

In this article, we focus on the instructional design for each classroom, instead of courses or programs. Based on the OBE paradigm and the UbD framework, our instructional design model comprises a planning process for the development and application of test cases.

III. METHODS

The research question is whether our presented test-driven design model for instruction can improve student cognitive learning outcomes in a lecture or a class.

The hypothesis is that the TDD model can improve student lecture cognitive learning outcomes.

First, the TDD model is introduced. Then the TDD model was used in classes to investigate the effect on the outcomes of cognitive learning of the students.

The Statistical Package for the Social Sciences (SPSS version 26) is used as a tool to analyze the data.

A. *Test-Driven Design Model*

Test-Driven was initiated as a practice for developing high-quality software about two decades ago. The traditional software development process has the following sequence: analysis, design, code, and test. This method is also called test-last because a test occurs after coding is completed. To overcome the limitations of traditional development methods, the test-driven method moves the test from the end to the beginning, forming a new cycle: test-design-code-refactor [22]. Research shows that the test-driven method reduced code defects and improved software quality [23]. Although the concept of testing first is not new, testing-driven has been used mainly in software development practice [24].

There are different types of learning outcomes. In this article, we focus on cognitive learning outcomes. Learning outcomes refer to the cognitive learning outcome in our model. During instructional design, course learning outcomes are broken down into unit learning outcomes, which are further divided into learning outcomes for each lecture. Usually, a lecture learning outcome is a basic outcome, which is no longer split. A class session contains one or two learning outcomes from the lecture. With traditional instructional design methods, an instructor will design teaching and learning activities according to the expected learning outcome(s) (or instructional objectives). However, with our instructional design model, teachers will develop test cases for lecture learning outcomes. Test cases are used to measure whether students achieve the expected learning outcomes or not.

After developing test cases, teachers continue to design teaching and learning activities as traditional approaches. This model is named Test-Driven Design (TDD) because test cases are created according to the expected learning outcomes of a lecture and test results will affect the following instruction. In a class or a lecture, after introducing the learning objectives, a teacher will assess the students with a test case that would be randomly selected from the test cases created. Then the teacher and students will continue the lecture in traditional ways. Before the end of the class, the students will take the test again. Then the results will be discussed, and errors will be corrected if there are any. The TDD model is depicted in Fig. 1.

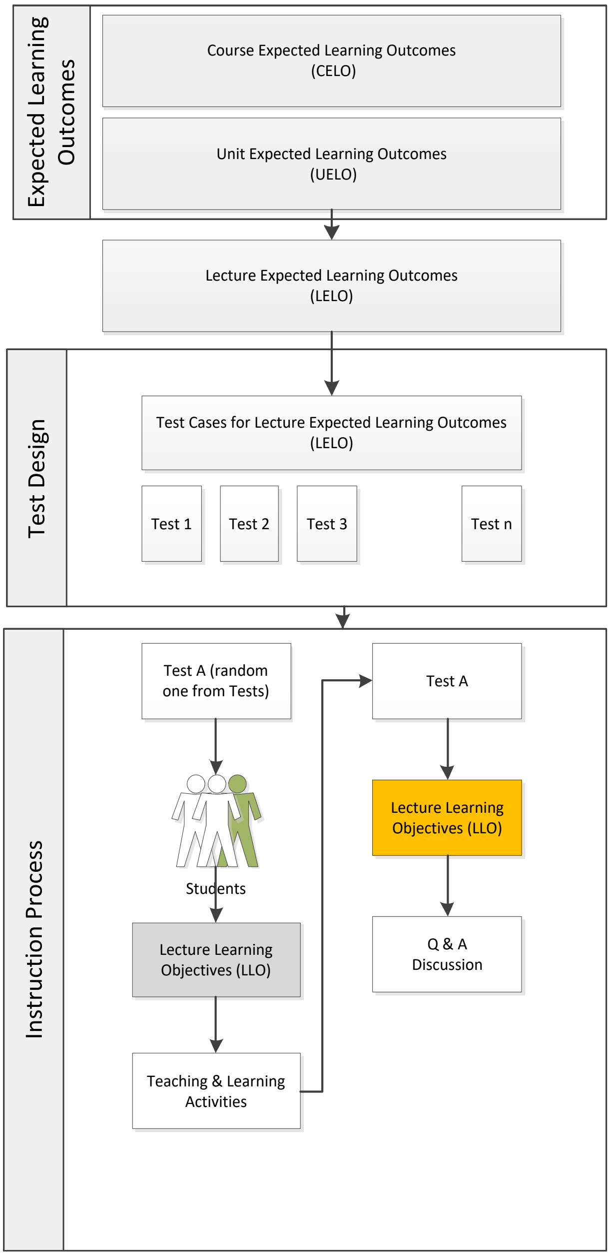

Fig. 1. Test-Driven model for instructional design.

B. Experimental Process

An experiment was designed to test the hypothesis. The C programming language is a compulsory course for undergraduate students majoring in mechanical engineering or automatic engineering. The students took the course in the second semester of their first year. The course lasted 12 weeks with two 2-hour sessions each week. One group consisted of students from mechanical engineering and another group consisted of students from automatic engineering. One of the two groups was randomly selected as the Experimental Group (EG) and the left as the Control Group (CG).

Both groups had a test as a pretest (Test P) in the first week of the course. The test-driven strategy was employed in EG and traditional methods in the CG group for the next 12 weeks. In the 14th week, two weeks after the end of the course, both groups were evaluated with another test as a post-test (Test F). The student grades for Test P and Test F were analyzed. The process is shown in Fig. 2.

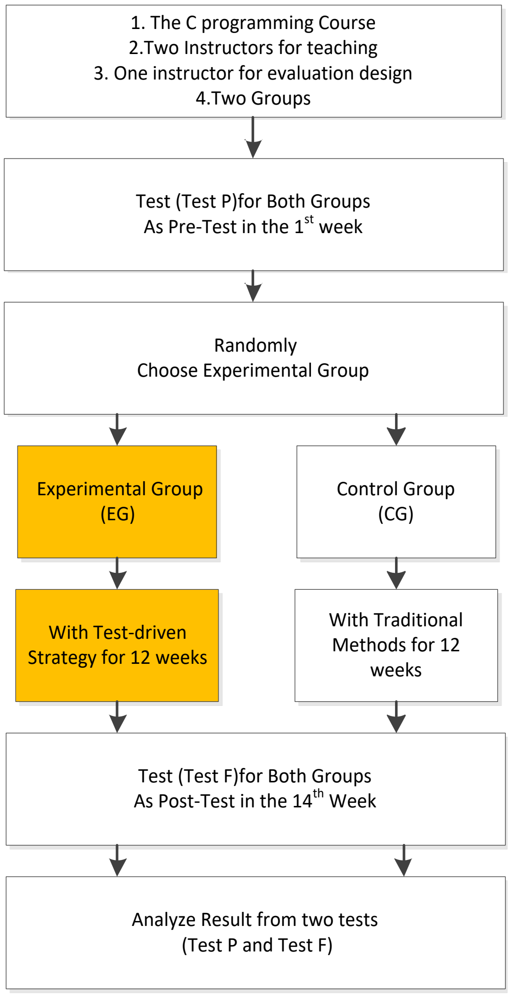

Fig. 2. The process for the experiment.

C. Participants

There were 53 undergraduates (female = 17, male = 36) from automatic engineering in group one and 54 students (female = 17, male = 37) from mechanical engineering in group two. Group two was randomly chosen as the

Experimental Group (EG) and group one as the Control Group (CG).

Instructor A, with a Master's degree in mechanical engineering, taught the Experimental Group (EG). Instructor B, with a doctorate in automatic engineering, was responsible for the Control Group (CG). Instructor C, an independent teacher with a doctorate in computer science, was in charge of the design of tests for both groups but did not participate in teaching.

In the first week, before the course, a test designed by Instructor C was assigned to both groups. The test includes three questions about C programming. The first was multi-choice about valid variable names. The second question was about loop statements and required students to write the value of a variable. The third, related to conditional statements, asked students to write the value of a variable.

During the following 12 weeks, Instructor A taught EG with the test-driven strategy and created all test cases for the class. Instructor B taught CG with methods used before. Both groups had 2-hour lectures twice a week.

Both groups finished the course in the 12th week. Then, all students had a week to prepare for the final test, which was designed by Instructor C. In the 14th week, all students took the final test.

IV. Results

For the first test (Test P), only several students in both groups correctly answered the questions. It is understandable because most students had not learned a programming language before. For each group, we counted how many students correctly answered the questions. Table I shows the number of students in two groups who answered correctly or incorrectly for each question.

TABLE I. Number of Correct and Wrong Answers for Each Question

Group	Question	Number of correct answers	Number of wrong answers
CG	1st question	7	46
CG	2nd question	3	50
CG	3rd question	1	52
EG	1st question	6	48
EG	2nd question	2	52
EG	3rd question	1	53

For the first question, 7 students in the CG answered correctly and 6 did in the EG. The Pearson Chi-Square test is used to analyze the differences between the two groups. The results show that there is no significant difference (p = 0.518 > 0.05) between the two groups.

For the second question, Fisher's test is used because the number of correct answers is 2 and 3 respectively. Both are less than 5. There are no significant differences (p = 0.678 > 0.05) between the two groups in answering the second question.

Similarly to the second question, there is no significant difference (p = 0.556 > 0.05) between the two groups for the third question.

We can conclude that there is no significant difference in the knowledge of the C programming language between the CG and the EG.

For the final test (Test F), the normality distributions of grades were checked with the Shapiro-Wilk Test in SPSS 26. Statistics were shown in Table II.

TABLE II. Normality Test with Shapiro-Wilk

Group	Mean	Std. Deviation	Statistic	Df	Significance
CG	58.5	16.2	0.992	53	0.969
EG	65.5	18.1	0.931	54	0.004

The students in EG had a higher average grade than the students in CG. The p-value of CG is 0.969, which indicates that the grades of CG distribute normally. However, the p-value of EG is 0.004, which means that the grades of EG did not distribute normally.

Therefore, a nonparametric test, the Mann-Whitney Test, was employed to compare the means of the two groups. And the one-tailed p-value is 0.013 (p < 0.05), which indicates that there is a statistical difference between the means of the two groups.

Based on the results of two tests, we concluded that at the beginning of the c programming course, students from CG and EG had similar knowledge or skills in C programming. At the end of the course, students in the EG had a significantly higher average grade than students in the CG did.

V. Discussion

Explicit learning objects motivate students to learn. The expected learning outcomes of a lecture help students understand what they should be able to do at the end of the lecture.

The TDD model helps instructors align their instruction with the learning outcomes. During the process of test case design, teachers will fully understand the objectives of a lecture, which facilitates the development of the teaching and learning activities or learning resources. The emphasis on learning outcomes is aligned with the OBE paradigm.

Students take a test twice in a class. The first-time test helps students understand not only the expected learning outcomes but also the context in which students learn how and when to apply what they learn.

The second time, the students know their performance and the gap between the expected outcomes and their actual achievement. It will also help the instructor to get a clear picture of the student learning outcomes, which helps the instructor reflect on teaching and learning. Additionally, the instructor provides students with effective instant feedback on student performance. The specified feedback practice guided by goals is in line with the principles of how people learn [11].

Therefore, the test-driven design model incorporates the critical tenets of the science of learning. According to Gagne's learning theory, learning conditions include attracting the attention of learners, knowing the learning objectives, providing effective feedback, and measuring the learning outcomes [25]. The TDD model helps instructors teach by complying with some of the conditions. For example, it explicitly informs students of the learning objectives at the beginning of a class. The test helps students to see the context for the application of new knowledge. All of this helps students understand what they will learn and why. The test before the end of the class will assess what they have learned, helping teachers and students know the learning outcomes. Then the following discussion will provide students with instant feedback, which corrects students' misunderstandings and fosters deep learning.

However, there are some limitations of this research. Firstly, the sample sizes of students and teachers are small. There were 53 students in the control group and 54 in the experimental group. And three instructors participated in this study. We cannot tell whether bigger-size groups will achieve the same benefits with different instructors. Secondly, the presented model was applied in only the C programming course. It is not clear whether the TDD model can improve learning outcomes in other courses. Therefore, we will try this model in more courses in the future. In addition, this study did not investigate the participants' opinions about the model. We know neither what the instructors think about the model nor what the students think about the model.

Therefore, more work is required to investigate the reliability and validation of the TDD model. In the future, we will apply this model in more courses with different size classes. Also, we will survey opinions from students and instructors about the TDD model.

VI. Conclusion

A Test-Driven Design (TDD) model is presented to improve course learning outcomes in higher education. Based on learning theories and related principles, the TDD model guides teachers in designing instruction by starting with creating test cases according to learning outcomes. The TDD helps instructors keep assessments, learning and teaching activities, and learning outcomes in line. An experimental application of the TDD reveals that the TDD improved student learning outcomes in the C programming courses.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Yongbin Zhang constructed the test-driven strategy and also participated in the course instruction. Xiuli Fu designed the experiment and wrote the draft of this research. Li Wei analyzed the data; Chunmei Wang was responsible for proofreading and applied the test-driven strategy in the course. All authors had approved the final version.

Funding

The study is supported by the Beijing Municipal Education Commission (Grant No. 2021149), the Beijing Higher Education Academy (Grant No. MS2022398), and the Beijing Institute of Graphic Communication (Grant No. 20231001).

References

[1] W. G. Spady, "Choosing outcomes of significance," *Educational Leadership*, vol. 51, no. 6, pp. 18–22, 1994.

[2] D. Soares, P. Carvalho, and D. Dias, "Designing learning outcomes in design higher education curricula," *International Journal of Art & Design Education*, vol. 39, no. 2, pp. 392–404, 2020.

[3] M. M. Syeed, A. Shihavuddin, M. F. Uddin, M. Hasan, and R. H. Khan, "Outcome Based Education (OBE): Defining the process and practice for engineering education," *IEEE Access*, vol. 10, pp. 119170–119192, 2022.

[4] K. Premalatha, "Course and program outcomes assessment methods in outcome-based education: A review," *Journal of Education*, vol. 199, no. 3, pp. 111–127, 2019.

[5] J. Allan, "Learning outcomes in higher education," *Studies in Higher Education*, vol. 21, no. 1, pp. 93–108, 1996.

[6] W. G. Spady and K. J. Marshall, "Beyond traditional outcome-based education," *Educational Leadership*, vol. 49, no. 2, pp. 67–72, 1991.

[7] K. M. Enneking, G. R. Breitenstein, A. F. Coleman, J. H. Reeves, Y. Wang, and N. P. Grove, "The evaluation of a hybrid, general chemistry laboratory curriculum: Impact on students' cognitive, affective, and psychomotor learning," *Journal of Chemical Education*, vol. 96, no. 6, pp. 1058–1067, 2019.

[8] D. R. Krathwohl, "A revision of bloom's taxonomy: An overview," *Theory into Practice*, vol. 41, no. 4, pp. 212–218, 2002.

[9] H. Vidyanagar, "Programme outcomes programme specific outcomes course outcomes," doctoral dissertation, Osmania University, 2015.

[10] J. S. Twyman and W. L. Heward, "How to improve student learning in every classroom now," *International Journal of Educational Research*, vol. 87, pp. 78–90, 2018.

[11] L. Yang, "The role of feedback orientation in converting external feedback to learning opportunities for implementing assessment-as-learning in the context of feedback," in *Assessment as Learning*, Routledge, 2021, pp. 53–75.

[12] E. Molloy, D. Boud, and M. Henderson, "Developing a learning-centred framework for feedback literacy," *Assessment & Evaluation in Higher Education*, vol. 45, no. 4, pp. 527–540, 2020.

[13] G. W. Lindsay, "Attention in psychology, neuroscience, and machine learning," *Frontiers in Computational Neuroscience*, vol. 14, p. 29, 2020.

[14] R. Bodily, H. Leary, and R. E. West, "Research trends in instructional design and technology journals," *British Journal of Educational Technology*, vol. 50, no. 1, pp. 64–79, 2019.

[15] J. K. McDonald and S. C. Yanchar, "Towards a view of originary theory in instructional design," *Educational Technology Research and Development*, vol. 68, pp. 633–651, 2020.

[16] C. M. Budoya, M. M. Kissaka, and J. S. Mtebe, "Instructional design enabled agile method using ADDIE model and feature driven development process," *International Journal of Education and Development Using Information and Communication Technology*, vol. 15, no. 1, p. n1, 2019.

[17] E. Widyastuti, "Using the ADDIE model to develop learning material for actuarial mathematics," in *Journal of Physics: Conference Series*, IOP Publishing, 2019, vol. 1188, no. 1, 012052.

[18] M. A. Stapa and N. Mohammad, "The use of Addie model for designing blended learning application at vocational colleges in Malaysia," *Asia-Pacific Journal of Information Technology and Multimedia*, vol. 8, no. 1, pp. 49–62, 2019.

[19] J. McTighe and G. Wiggins, *Understanding by Design Framework*, Alexandria, VA: Association for Supervision and Curriculum Development, 2012.

[20] D. H. Andrews and L. A. Goodson, "A comparative analysis of models of instructional design," *Journal of Instructional Development*, vol. 3, no. 4, pp. 2–16, 1980.

[21] J. R. Bajracharya, "Instructional design and models: ASSURE and Kemp," *Journal of Education and Research*, vol. 9, no. 2, pp. 1–9, 2019.

[22] V. S. Bhadauria, R. Mahapatra, and S. P. Nerur, "Performance outcomes of test-driven development: An experimental investigation," *Journal of the Association for Information Systems*, vol. 21, no. 4, pp. 1045–1071, 2020.
[23] I. Karac, B. Turhan, and N. Juristo, "A controlled experiment with novice developers on the impact of task description granularity on software quality in test-driven development," *IEEE Transactions on Software Engineering*, vol. 47, no. 7, pp. 1315–1330, 2019.
[24] A. Santos, *et al.*, "A family of experiments on test-driven development," *Empirical Software Engineering*, vol. 26, pp. 1–53, 2021.
[25] M. J. Dolasinski and J. Reynolds, "Microlearning: A new learning model," *Journal of Hospitality & Tourism Research*, vol. 44, no. 3, pp. 551–561, 2020.