

Exploration on the Teaching Reform of Python Programming based on BOPPPS

Zhehong Zhou

Guangdong University of Science and Technology, Dongguan 523668, China

Abstract: BOPPPS is a new teaching model which is educational goal-oriented and student-centered. Teaching with BOPPPS method can help students establish a learning framework in Python programming course, define their learning goals, lay a solid foundation, understand and master the basic principles and skills of programming, and improve their programming ability and innovative thinking through practice. This teaching method can stimulate students' interest and initiative in learning and improve teaching effect and learning results.

Keywords: Python Programming Course; BOPPPS; Goal-oriented; Student-centered.

1. Introduction

The BOPPPS model was originally developed by the United States Naval Academy and is widely used in the field of education. BOPPPS is an instructional design model, which includes the following five steps: Bridge (introduction), Objective (goal), Preparation (preparation), Presentation (presentation) and Practice (practice). In the teaching of Python programming, the use of BOPPPS teaching can bring the following benefits: (1) introduction (Bridge): through the introduction part, teachers can connect students' previous knowledge with new learning content to help students establish a learning framework. In Python programming, we can stimulate students' interest and initiative by introducing some programming concepts that have been learned or connecting with practical problems that students are familiar with. (2) Objective: defining learning goals is very important for students' learning. In Python programming teaching, teachers can clearly point out learning goals, such as learning to master specific programming syntax, the ability to solve practical problems, and so on. This can help students define their learning goals and stimulate their learning motivation. (3) Preparation: in the preparation stage, teachers can provide the necessary pre-knowledge and skills to lay a solid foundation for students. In Python programming, we can help students understand and master necessary knowledge and skills by explaining basic programming concepts, syntax and common programming tools. (4) Presentation: in the presentation stage, teachers can use a variety of teaching methods, such as explanation, demonstration, demonstration, etc., to impart knowledge and skills to students. In Python programming, we can help students understand and master the basic principles and skills of programming by writing simple sample code and showing actual application scenarios. (5) Practice: practice is the key link of learning. Through practice, students can consolidate the knowledge and skills they have learned and cultivate their ability to solve problems. In Python programming, students can have the opportunity to actually use their knowledge and skills to improve their programming ability and innovative thinking by writing small programs, completing programming tasks and participating in project development. (6) Summary: finally, the teacher will summarize the key contents of this lesson and emphasize the key knowledge points, so that students can better master and

remember.

Many scholars have studied the BOPPPS teaching method. Ma[1] studied the impact of blended learning using the BOPPPS model on students' learning outcomes and perception in the introductory course of health service management in China. The purpose of this study is to evaluate the impact of blended learning based on BOPPPS model on students' learning achievement and curriculum perception, as well as students' attitudes and views on this teaching method. The study uses empirical research methods, including questionnaires, student performance evaluation and interviews. The results show that the blended learning using BOPPPS model has a positive impact on students' learning results in the introductory course of health service management. Zhang[2] studied the instructional design and practice of intensive reading course based on BOPPPS model. The purpose of this study is to explore the instructional design and practice of intensive reading courses based on BOPPPS model, and to evaluate its impact on students' learning effect and learning experience. The study uses empirical research methods, including curriculum design, student performance evaluation and teacher-student feedback. Li[3] puts forward a reform scheme of multi-platform hybrid Python programming teaching model based on BOPPPS model. The purpose of this study is to explore the reform of multi-platform hybrid Python programming teaching model based on BOPPPS model, and to evaluate its impact on students' learning achievement and learning experience. The results show that the reform of multi-platform hybrid Python programming teaching model based on BOPPPS model can effectively improve students' learning achievement and learning experience. Students have a positive attitude towards the reform of the multi-platform hybrid Python programming teaching model based on BOPPPS model, and think that this teaching model will help to improve their learning motivation and programming ability. Luo[4] studied the design of network attack and protection course based on BOPPPS model. The results show that the design of network attack and protection course based on BOPPPS model can effectively improve students' learning achievement and learning experience. Students can better understand and apply the basic concepts and technologies of network attack and protection, and think that this teaching model is helpful to improve students' learning motivation and skills.

Taking the cyberspace security major of the computer College of Guangdong University of Science and Technology as an example, this paper explores the teaching reform of Python programming course based on BOPPPS.

2. Main Contents of Teaching Reform

The main contents of teaching reform are carried out under the guidance of six teaching links in BOPPPS theory.

2.1. Bridge-in

At the beginning of the course, review the programming knowledge that students have learned, such as programming languages or programming concepts that have been learned before, and ask some questions related to Python programming to guide students to think, such as: "which programming language have you learned before?" How is Python different from this language? " At the beginning of each chapter, the Bridge-in operation can be designed as follows: (1) course introduction, introducing students to the basic concepts and characteristics of Python programming language, as well as the wide range of applications of Python in practical applications. (2) Environment setting to help students install and configure Python development environment, such as Python interpreter and integrated development environment (IDE). (3) basic grammar: teach the basic grammar of Python, including variables, data types, operators, conditional statements, loop statements and functions. (4) input and output: teach how to perform input and output operations in Python, including getting user input using the input () function and outputting results using the print () function. (5) data structures: introduce the data structures commonly used in Python, such as lists, tuples, dictionaries and collections, and teach their basic operations and usage. (6) File operation: teach how to read and write files in Python, including opening files, reading file contents, writing files and closing files, etc. (7) functions and modules: introduces the definition and calling methods of functions, and teaches how to use Python's built-in and custom functions. At the same time, it introduces the concept and usage of the module, and guides students to use the existing module to expand the function of Python. (8) error handling: teach how to handle exceptions and errors in Python, including using try-except statements to catch and handle exceptions. (9) programming practice: guide students to carry out some simple programming practices, such as writing small programs to solve practical problems or complete specific tasks, in order to improve their programming ability and practical experience. (10) Project practice: organize students to participate in small-scale project practice, let them apply the Python knowledge and skills they have learned to solve practical problems or complete specific tasks, so as to deepen their understanding and application ability of Python programming.

2.2. Objective

During the Objective phase of the Python course, students will further learn and apply advanced knowledge and skills of Python programming. The goal of this phase is to help students understand the Python programming language in depth and be able to use Python to solve more complex problems and develop applications independently. In the Objective phase, you can cover the following: (1) Advanced syntax and concepts: students will learn more advanced Python syntax and concepts, such as list deduction, generator,

decorator, iterator, and context manager. These concepts will help students write more concise and efficient code. (2) object-oriented programming: students will learn in depth the concepts and principles of object-oriented programming (OOP), and learn to create classes, define properties and methods, and implement inheritance and polymorphism in Python. This will enable students to better organize and manage complex code. (3) Database operation: students will learn how to use Python to interact with the database. They will learn techniques such as connecting to databases, performing SQL queries, and updating operations to process and manage data in real-world projects. (4) Network programming: students will learn Python's network programming libraries and modules, such as socket and requests. They will learn to use Python for network communication and Web development so that they can build web-based applications. (5) GUI programming: students will learn to use Python's GUI programming libraries such as Tkinter, PyQt and wxPython. They will learn to create graphical interface applications to provide users with a friendly interface and interactive experience. (6) data science and data analysis: students will learn the application of Python in the field of data science and data analysis. They will learn to use libraries such as NumPy, Pandas, and Matplotlib for data processing, analysis, and visualization to extract useful information from the data. (7) Web development: students will learn to use Python for Web development. They will learn to create Web applications using frameworks such as Django or Flask so that they can build powerful Web applications. (8) concurrency and multithreading: students will learn concurrent programming and multithreading techniques in Python. They will learn to use threads and processes for concurrent programming to improve program performance and efficiency. (9) debugging and optimization: students will learn to use debugging tools and techniques to debug Python programs. They will also learn some optimization techniques and strategies to improve the performance and efficiency of the program.

2.3. Pre-assessment

In the Pre-assessment stage, the purpose is to evaluate and test the students' mastery of the previous learning content. The goal of this stage is to help teachers understand students' knowledge and skills in Python programming, and provide guidance and adjustment for subsequent teaching, including knowledge testing, programming tasks and project evaluation. Students may need to take a knowledge test to test their understanding of the basics of Python programming. This may include problems with syntax, data types, conditional statements, loops, functions, and so on. Students may need to complete some programming tasks to demonstrate their practical skills in Python programming. These tasks may involve solving simple problems, writing functions, or implementing algorithms. Students may need to participate in a small project evaluation to demonstrate their ability to apply Python programming in real projects. This can help teachers assess students' abilities in project management, code organization, problem solving and teamwork. Through the evaluation of Pre-assessment stage, teachers can understand students' strengths and weaknesses in Python programming, and provide a basis for follow-up teaching plans and personalized guidance. This can ensure the pertinence of teaching contents and methods, and help students learn and apply Python programming better.

2.4. Participatory Learning

In Participatory Learning, students will participate in more interactive and participatory learning. The goal of this stage is to deepen students' understanding and application of Python programming through practice, cooperation and discussion, including practical projects, group cooperation, discussion and sharing, project presentation, feedback and evaluation. Students will participate in practical projects, applying their knowledge and skills in Python programming. These projects can be individual projects or group projects, and students will consolidate and expand their programming skills through practice. Students will form groups with other students to complete projects and tasks. Through teamwork, students can learn to work as a team, communicate and collaborate, and improve their ability to cooperate and solve problems. Students will have the opportunity to participate in discussion and sharing activities. This can include group discussions, class discussions, code reviews, etc., where students can share their ideas, experiences and solutions and learn from the views and methods of others. Students will have the opportunity to demonstrate the projects and results they have completed. This can be a class presentation, a presentation, or an online presentation, where students can show their projects, explain their design ideas, and demonstrate their code. Students will receive feedback and evaluation from teachers and classmates. This can help students understand their advantages and room for improvement, and promote their continuous improvement and development, so as to achieve the training goal of cyberspace security major. Through the study of Participatory Learning stage, students will be able to understand and apply Python programming more deeply, and cultivate their practical ability, teamwork spirit and problem-solving ability. This can make students more confident and independent in practical projects and lay a solid foundation for their future study and career development.

2.5. Post-assessment

Post-assessment aims to comprehensively evaluate and test the knowledge and skills acquired by students throughout the course of study. The goal of this phase is to help teachers assess students' overall level of Python programming and provide feedback and guidance to students. It can be carried out from the following aspects: (1) Comprehensive knowledge test: students may need to take a comprehensive knowledge test to test their comprehensive mastery of Python programming. This may involve syntax, data structure, algorithm, object-oriented programming, database operation, network programming, GUI programming, data science, data analysis and so on. (2) Integrated programming tasks: students may need to complete some comprehensive programming tasks to demonstrate their comprehensive ability in Python programming. These tasks may involve solving complex problems, designing and implementing complete applications or systems, and so on. (3) Project evaluation: students may need to participate in a comprehensive project evaluation to demonstrate their ability to apply Python programming in practical projects. This can help teachers assess students' abilities in project management, code organization, problem solving and teamwork. Through the Post-assessment assessment, teachers can fully understand the ability and potential of students in Python programming, and provide students with personalized feedback and guidance. This can help students to further

consolidate what they have learned, discover and solve their weaknesses, and prepare for further study and career development. At the same time, teachers can also adjust and improve the curriculum through the evaluation results to improve the teaching effect and students' learning experience.

2.6. Summary in BOPPPS

The Summary phase of Python programming is usually a summary and review phase at the end of the course. The purpose of this stage is to help students review and consolidate the knowledge and skills they have learned in the course, and reflect and summarize the whole learning process. In the Summary phase, teachers and students can carry out the following activities together: (1) knowledge review: students will review the main knowledge points and concepts learned in the course. This can be done through class discussion, group sharing, exercises, etc. Teachers can provide a comprehensive knowledge review material to help students review and consolidate what they have learned. (2) skill consolidation: students will have the opportunity to consolidate their Python programming skills through practice and practice. This can include writing small projects, solving programming problems, completing programming tasks, and so on. Through practical coding exercises, students can deepen their understanding and application of what they have learned. (3) Summary and reflection: students will have the opportunity to summarize and reflect on the whole learning process. They can review their learning achievements, challenges and problems solved in the course. This helps students to recognize their own growth and progress and provide guidance for future learning. In short, the Summary phase is a summary and review of Python programming, which aims to consolidate students' knowledge and skills and help them reflect and further develop.

3. Summary

BOPPPS teaching method requires teachers to set teaching goals clearly before teaching. For Python programming courses, clear teaching objectives can help students understand the knowledge and skills they will learn and stimulate their learning motivation. The BOPPPS teaching method emphasizes objective evaluation, which evaluates students' learning achievements through tests, assignments and projects. In the Python programming course, this means that students can demonstrate their understanding and application skills by writing and debugging Python code. The BOPPPS teaching method requires pre-assessment before teaching in order to understand the students' previous knowledge and skills. In the course of Python programming, pre-evaluation can help teachers understand the programming basis of students and make personalized teaching arrangements according to the different needs of students. BOPPPS teaching method provides a variety of teaching strategies, including group cooperation, case studies, discussion and so on. In the Python programming course, these teaching strategies can help students better understand and apply programming concepts, and develop the ability of problem solving and teamwork. To sum up, BOPPPS teaching method is very important to Python programming courses, which can help teachers organize and deliver teaching content more effectively and promote students' learning and development.

References

- [1] Ma X, Ma X, Li L, et al. "Effect of blended learning with BOPPPS model on Chinese student outcomes and perceptions in an introduction course of health services management." *Advances in Physiology Education* 45.2 (2021): 409-417.
- [2] Zhang, Liang. "Teaching design and practice of intensive reading course based on BOPPPS." *Journal of Language Teaching and Research* 11.3 (2020): 503-508.
- [3] Li, Min, et al. "Reform of the multi-platform blended teaching model of Python programming based on BOPPPS." 2021 16th International Conference on Computer Science & Education (ICCSE). IEEE, (2021): 398-401.
- [4] Luo, Haibo. "The Course Design of Network Attack and Prevention Under the BOPPPS." 2022 6th International Seminar on Education, Management and Social Sciences (ISEMSS 2022). Atlantis Press, (2022).