A Study on the Evaluation of Intermediate Students' Computational Thinking Based on SOLO Classification Theory

Ruoyi Wang[1], Ran Lu*[2], Peiyu Liu[3]

{wry990816@163.com[1], luran@sdnu.edu.cn*[2], Liupy@sdnu.edu.cn[3]}

Shandong Normal University, Jinan China

Abstract. Computational thinking, as a crucial criterion for assessing students' ability to independently solve problems, has long been a core concept of international interest. With the growing demand for IT talent in China, the evaluation of computational thinking among secondary vocational school students has become an urgent issue to address. This article explores how to evaluate the computational thinking of these students by combining the definition of computational thinking with the SOLO classification theory.

Keywords: computational thinking; SOLO classification theory

1 Introduction

Industry 4.0 has heightened the need for systematic thinking, requiring workers to possess advanced skills that encompass independent thinking. In the era of 4.0, thinking ability has emerged as one of the most significant qualities of talented individuals.[1] Computational thinking has become a crucial requirement for talent in the current era of digitalization and informationization. Individuals equipped with computational thinking skills are capable of effectively navigating the significant impact and influence of artificial intelligence on both human life and social structures. [2]

As one of the key components of the future workforce, it is helpful for secondary school students to have appropriate computational thinking for their future work. As of June 2023, a total of 6132 documents were searched in the Web of Science database with the theme of "computational thinking", Few previous studies have pinned the evaluation study of computational thinking of secondary vocational students, as shown in Figure 1.

ICMEIM 2023, September 08-10, Wuhan, People's Republic of China

DOI 10.4108/eai.8-9-2023.2340053

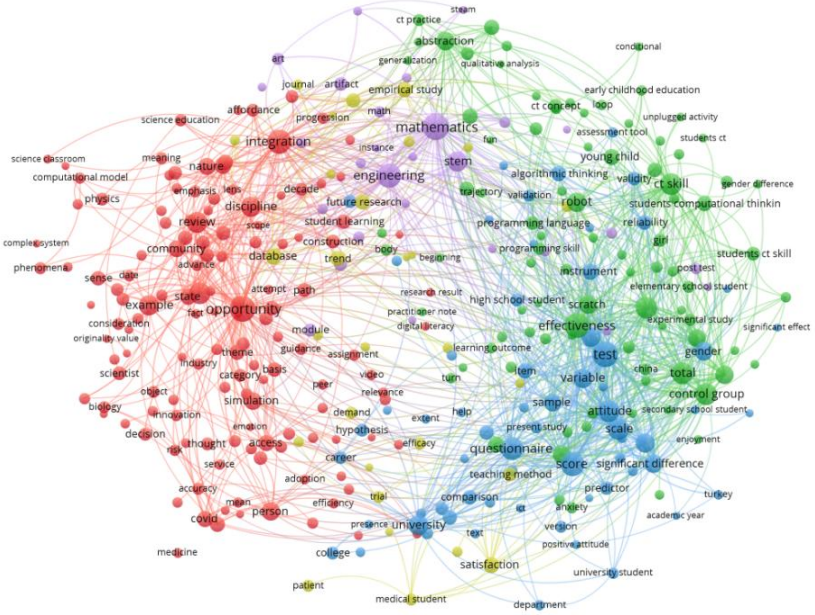

Fig. 1. Web of Science computational thinking core journal literature count

2 Relevant concept

2.1 Computational thinking

Computational thinking education is a core concept garnering significant attention in the global computing field, serving as a foundation for approaching information technology education. It acts as a vital indicator of an individual's ability to independently solve problems. By cultivating computational thinking skills, students gain a deeper understanding of complex problems, enabling them to analyze and tackle challenges more effectively while continuously enhancing their problem-solving abilities.

Many academics and organizations have developed the definition of computational thinking from many viewpoints since Professor Zhou Yizhen first suggested the idea in 2006, as shown in Table 1.

Table 1. Definitions of computational thinking by different academics or organizations

Academics/ Organizations	Definitions of computational thinking
Zhou Yizhen	Computational thinking is a series of thinking activities that cover the breadth of computer science, such as problem solving, system design, and understanding of human behavior, using the concepts underlying computer science. [3]
Denning	The most important aspects of computational thinking are the understanding of abstraction, the ability to process at different levels of abstraction, algorithmic thinking, and the understanding of the impact caused by, for example, big data. [4]

Dong Rongsheng	Computational thinking is the use of computer science ideas and methods to solve problems, design systems, and understand human behavior. [5]
OECD	The core elements of computational thinking mainly include abstraction, algorithmic thinking, automation, decomposition, debugging, and generalization.
Information Technology Curriculum Standards for Secondary Vocational Schools in China(2020)	Computational thinking refers to a series of thinking activities generated in the process of analyzing and processing information and solving problems by individuals using the ideological methods of information technology. [6]

Through the analysis of the above definitions, there is no consensus on the definition of computational thinking. However, both domestically and internationally, computational thinking is regarded as a thinking process for problem-solving. It involves learners utilizing computers to establish task objectives and discern task logic. For vocational students, whether they enter the workforce after graduation or continue their studies in higher education institutions, possessing a certain level of computational thinking is indispensable and beneficial to their personal development.

2.2 SOLO classification theory

The full name of SOLO is Structure of the Observed Learning Outcome. The theoretical foundation of SOLO is based on Piaget's theory of developmental stages and was proposed by renowned educational psychologist Professor Biggs and his colleagues after extensive research and exploration, and it is a qualitative evaluation method characterized by hierarchical descriptions. [7] According to the structure of students' thinking as reflected in their learning outcomes, the SOLO classification theory categorizes the changes in students' thinking levels into five levels: pre-structure, single-point structure, multi-point structure, correlation structure, and abstract extension structure, as shown in Table 2. [8] [9]

Table 2. The five levels of thinking hierarchy of SOLO categorization theory

Level of thought structure	Concrete expression
pre-structure	Students are unable to gather useful information from the context and integrate it with their existing knowledge, resulting in confused thinking and an inability to effectively respond to relevant problems.
single-point structure	Students are able to obtain some useful information from the context or find few clues within their fragmented relevant knowledge, but can only respond with one possible answer.
multi-point structure	Students access multiple unconnected cues in context or pre-existing cognition and respond with multiple isolated responses.
correlation structure	Students connect valid information extracted from the situation with their original cognition, grasp the structure of knowledge as a whole, and are able to associate multiple events and link them together to respond to more complex interrelationships in their responses.
abstract extension structure	Students can apply their existing knowledge and skills in new situations, summarize concepts and patterns from objective facts, and substantiate anticipated concepts and patterns as facts through reasoning. They are able to provide answers that involve induction and hypotheses.

SOLO classification theory believes that the changes in the structure of thinking shown by learners in answering questions are detectable, so the SOLO classification evaluation theory has great reference significance to the scoring criteria and the preparation of open-ended test questions. In this paper, we take the class "Programming Language Fundamentals" as an example to construct the evaluation system of computational thinking for middle school students.

3 The construction of the evaluation system

According to Professor Zhou Yizhen's definition of computational thinking, Alexander Reoenning divided the problem-solving process of computational thinking into three stages, [10] which is consistent with the three major manifestations of computational thinking that secondary students need to have as required by the abovementioned curriculum standards, as shown in Figure 2.

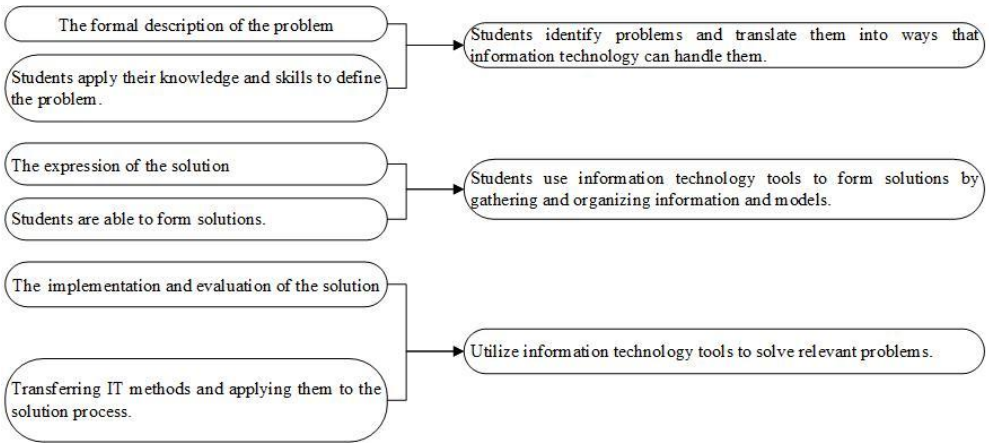

Fig. 2. Correspondence between the three stages and the three manifestations

3.1 Formalization of the problem

The formal expression of a problem is an attempt to formulate and define a problem, which means that students need to use the knowledge and skills they have acquired to define the problem and translate it into a way that information technology can deal with it.

Intermediate students need to understand the idea of structured program design in the course "Programming Language Fundamentals", master the concepts related to the C language as well as understand the basic ideas and methods of simple program design in C. That is to say, when faced with the relevant problems, the intermediate students can express them in the terminology related to the C language, and be able to understand the meaning of the problems. Intermediate students have the phenomenon of low interest in learning and no enthusiasm for learning, and they will show the phenomenon of avoiding questions without answering them, etc. This paper attributes this behavior to the pre-structural level, as shown in Table 3.

Table 3. "Formalized description of problems" evaluation scale.

SOLO Classification Theory Levels	Student Learning Behavior Exhibits
pre-structure	Students may not understand the problem or avoid it because they lack relevant concepts and problem-solving abilities. .
single-point structure	Students can have a basic understanding of the problem but struggle to identify which information is irrelevant, and unable to fullly translate problems into a form that is easily processed by a computer.
multi-point structure	Students can understand the problem accurately, and retain useful information.
correlation structure	Students are able to organize relevant information, and accurately and concisely represent the problems.
abstract extension structure	Students can decompose problems into sub-problems based on their existing knowledge. They establish connections between these sub-problems, forming a problem-solving approach.

3.2 Expression of solutions

The expression of a solution means that it is expressed in a non-fuzzy way so that it can be manipulated by a computer and at the same time this expression can support computer programming, which means that students use information technology tools to form a solution to a problem by collecting and organizing all kinds of information, models, and so on. Transforming the form of a problem is the foundation of computational thinking, the basic work of computer-based problem solving. [11] In this process, students need to transform the problem into a form that can be easily processed by computers. For the C language course, students need to master the connotation of commonly used symbols and rules of computation, and at the same time, they should be able to draw a flowchart for the problem, , as shown in Table 4.

According to the understanding of the problem in the previous stage, students need to know what concepts or algorithms should be applied to solve the problem, and if multiple concepts are involved, students should have certain integration ability.

Table 4. "Expression of solutions" evaluation scale.

SOLO Classification Theory Levels	Student Learning Behavior Exhibits
pre-structure	Students do not have the basic knowledge of the C language to be able to draw appropriate flowcharts and formulate solutions to problems.
single-point structure	Students are able to select data and algorithms based on their understanding of the problem, but are unable to develop a complete solution.
multi-point structure	Students are able to organize information using icons, text, or images and create flowcharts, but they are unable to successfully run the program and obtain results.
correlation structure	Students will be able to understand the problem and draw a flowchart, and will be able to model a solution based on the problem.

abstract extension structure	Students are able to translate real-world problems into computer-understandable forms and judge the reasonableness of the model based on the results of actual operations, use different treatments for different situations, and design algorithms that can be successfully applied to the problem-solving process.

3.3 Program implementation and evaluation

The implementation and evaluation of the solution means that the computer directly displays the results of the implementation of the solution in its own way and evaluates the results of the implementation and refines the solution, i.e., the student is able to migrate information technology methods and apply them to the solution process, solving the problem in question in accordance with the methods and techniques used in the first two phases. Although the execution and evaluation of the program is mainly done by the computer, the whole process of applying computational thinking may not be finished after the program is executed, and the students need to execute it several times so as to perfect the program. When carrying out this stage, secondary students need to incorporate what they have learned and debug the computer. Since most of the development direction of secondary students is oriented to employment, it is necessary to assess the students' understanding and application of the whole process of problem solving in this stage, and to know whether they can transfer the solution to similar problems in life, as shown in Table 5.

Table 5. "Program implementation and evaluation" evaluation scale.

SOLO Classification Theory Levels	Student Learning Behavior Exhibits
pre-structure	Students are unable to understand the programming that has been given and cannot run the program code successfully.
single-point structure	Students are able to run the code without problems, but can not resolve the error message on their own.
multi-point structure	Students can perform repetitions of the solution and summarize the basic ideas and methods of the solution.
correlation structure	Based on the overall solution, students are able to make connections between information, data, algorithms, etc., and know how to apply them to the solution session.
abstract extension structure	Students can abstract the logic of problem-solving thinking and optimize the solution to the overall problem in the context of different life applications and prior knowledge and experience, and transfer analogies to other areas of problem solving. [12]

The SOLO categorical theory levels were mapped to the level of computational thinking possessed by the students based on their performance of learning behaviors, as shown in the Fig.3 below.

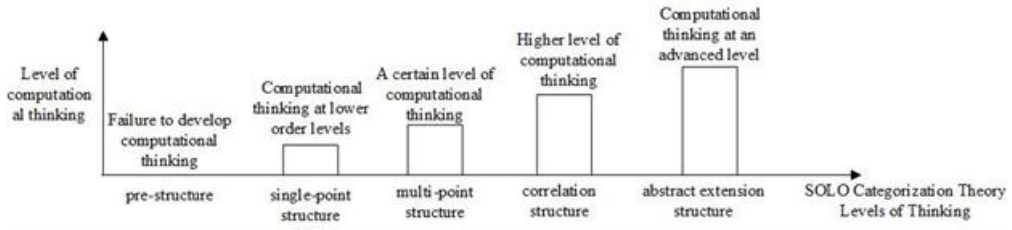

Fig. 3. Computational Thinking Levels Comparison Chart

4 Conclusion

After the formation of the evaluation scale, this study consulted the opinions of five frontline vocational school teachers. Through the organization and summary of the interview results, it was concluded that the vocational school computational thinking evaluation based on the SOLO classification theory can help teachers assess students' computational thinking after the completion of the curriculum. This assessment enables teachers to adjust teaching strategies and methods to better cultivate students' computational thinking, help them master professional knowledge, improve their professional skills, and lay a solid foundation for future employment or further education. However, the development of thinking is continuous and requires long-term cultivation. In addition to establishing evaluation criteria, it is also necessary to assess students' computational thinking levels through other forms.

As an important part of future workers, vocational school students need to not only master relevant professional skills but also possess certain computational thinking abilities. It is necessary to continue conducting in-depth research on how to more accurately assess students' computational thinking levels and establish a systematic and comprehensive evaluation system.

References

[1] Tang R. L. (2015) . Countermeasures research on cultivation of technical skill talents in the era of industry 4.0. Journal of Hubei Adult Education College, (6), 22-25, 38. doi: 10.16019/j.cnki.cn42-1578/g4.2015.06.006.

[2] Wang X. N. (2020). Research on the Cultivation of Computational Thinking in Higher Vocational Computer Basic Course. Computer Knowledge and Technology, 16(10), 148-149. doi: CNKI:SUN:DNZS.0.2020-10-064.

[3] WING J. M. (2006). Computational thinking. Communications of the ACM, (3), 33-35. doi: 10.1145/1118178.1118215.

[4] Denning, P. J. (2009). The profession of IT Beyond computational thinking. Communications of the ACM, 52(6), 28-30. doi:10.1145/1516046.1516054.

[5] Dong R. S. (2002). Methodology of computer science and technology. Posts & Telecom Press, Beijing, 9, 200. doi: 10.3969/j.issn.1002-137X.2002.01.001.

[6] Wang J. (2020). Cultivating information technology core literacy to create high-quality professional talents adapted to the needs of the times--Thinking about the implementation of

information technology curriculum standards in secondary schools. China Vocational and Technical Education, (17), 19-23.
https://wenku.baidu.com/view/e7d94ac865ec102de2bd960590c69ec3d4bbdbd3?fr=xueshu&_wkts_=1690196365077
[7] Wu Y. C. (2008). The application of SOLO taxonomy in teaching evaluation. Journal of South China Normal University (Social Science Edition), (03), 95-99+160. doi: CNKI:SUN:HNSB.0.2008-03-014.
[8] Cai, Y. H. (2006). SOLO categorization theory and its application in teaching[J]. Teacher Education Research, (01), 34-40. doi: 10.3969/j.issn.1672-5905.2006.01.008.
[9] John B. Biggs, Kevin F. Collis. (2010). Evaluation of learning quality: SOL0 classification theory (Structure of Observable Learning Outcomes). Translated by Gao Lingbiao and Zhang Hongyan. Beijing: People's Education Press, 28, 32-33. ISBN: 9787107228858.
[10] Nie Y. P. (2018). Research on the construction of evaluation indexes of computational thinking in junior middle school information technology courses. Shandong:Qufu Normal University. doi: CNKI:CDMD:2.1018.245245.
[11] Cheng S. X. (2006). Educational evaluation methodology and technology. Beijing:Beijing Normal University Press, 278-279. ISBN: 7-303-06841-4.
[12] Lin L. (2023). Teaching and evaluating integrated programming teaching design based on the SOLO classification theory--Taking the lesson of "intelligent desk lamp" as an example. New Course Introduction, (5), 63-66.
https://kns.cnki.net/kcms/detail/detail.aspx?FileName=XKCD202305017&DbName=CJFN2023