Exploration of Computational Thinking based on Bebras Performance in Webduino Programming by High School students

Jian-Ming Chen[1], Ting-Ting Wu[1]* and Frode Eika Sandnes[2]

[1]School of Technological and Vocational Education
National Yunlin University of Science and Technology, Yunlin 64002, Taiwan, R.O.C.
[2]Oslo Metropolitan University, Oslo, Norway
n10543007@yuntech.org.tw, danytingting@gmail.com, frodes@oslomet.no

Abstract. The 12-year Basic Education Curriculum Guidelines by the Ministry of Education in Taiwan includes learning performances related to computational thinking and programming languages in technology courses. The students will develop other important competence through programming. Learning a programming language should not only involve focus on writing the programs, but should also stimulate students' computational thinking competence and allow them to solve daily problems through information techniques. Situated learning emphasizes students' learning in real scenarios where knowledge is applied as the tool in these real situations. Without such scenarios, the tool has limited value. Likewise, computational thinking competence can be translated as effective problem-solving by the means of information technology. Hence, the thinking process involves analyzing the problems resulting in answers. In addition, the Bebras learning model is based on a concept of informatics which supports comprehension of information science phenomenon and development of computational thinking. This study explored the effects of computational thinking competence on the Bebras test performance. The study targeted senior high school students' who learned program design using a situated learning strategy.

The results confirm the importance of the situated learning strategy when cultivating students' computational thinking competence. Based on homogeneity of two groups of students, the experimental group's posttest score of computational thinking is higher than that of control group. The experimental group were exposed to a situated learning strategy and the control group was not. Significant difference between the two groups shows that the situated learning strategy reinforces computational thinking competence.

Keywords: Situated Learning, Computational Thinking, Bebras.

1 Introduction

The year 2014 was the "Year of Code" in the UK. According to regulation of Department of Education of Britain, children must start learning programming language from five years of age. Besides, program design courses have been included as one of obligatory subjects in junior high schools and elementary schools. Implementation of the policy influences the future competitiveness of a country. In Taiwan, the 12-year Basic Education Curriculum Guidelines list computational thinking and programming languages as competences for technology courses. It is a signal that students should enhance other important abilities through programming. Learning programming languages does not simply refer to superficial writing of programs. The goal is to stimulate students' computational thinking allowing them to solve daily problems by the means of information techniques. By learning program design, young people can master new technology which is highly important in this digital era as they do more than simply interact with the new technology. Through technology, they express their creativity and thoughts to further develop computational thinking competences [1].

Computational thinking is a type of core structure of conceptual evaluation. Programming teaches students to write programs and accomplish tasks through program design instructional tools. At the beginning and the end of the course, students' computational thinking competence is examined [2]. Bebras is an international competition of informatics and computer literacy. In the second Bebras international conference in 2006, there was one brainstorming meeting which proposed different types of questions of tasks that can be applied in the competition. In the conference, more than 120 tasks were developed and around 90 of these were accepted and elaborated further [3]. The Bebras learning model is based on concept of informatics which supports comprehension of information science and development of computational thinking.

Situated learning can effectively lead to knowledge construction. Brown, Collins, and Duguid stated the definitions and importance of situated cognition and situated learning. Thus, situated learning is gradually valued. Knoweldge should be constructed in real activities and learning activities should be combined with culture. Students thus interact with scenarios and result in knowledge. Situated learning emphasizes students' learning in real scenarios. Knowledge is treated as a tool in real scenarios. Without scenarios, these tools have limited value [4]. Likewise, computational thinking competences implies the ability to effectively solve problems using information technology. Thus, the thinking process involves analysis of problems that result in solutions.

Hence, explored effects of computational thinking competences on the Bebras test performance by senior high school students' that learned program design using a situated learning strategy.

2 Literature Review

2.1 Computational Thinking

In 2011, Wing broadly elaborated that question and solution are not simply a matter of mathematical problems. Solution must be analyzed and proved. It can be real problems in daily life which are solved by large-scale complex software system. Therefore, computational thinking, logic thinking and system thinking overlap. Wing thus redefined “computational thinking as reflection process to develop problem and solution and the method to present solution by message processing in problem-solving planning” [5].

Computational thinking resembles core objectives of basic education, reading, writing and arithmetic which should be included as one of core objectives required by computational thinking. In the field of computer science in the U.S. and the UK, computational thinking has been specifically listed as important core competences. Chen studied students’ computational thinking competences through robot programming. Based on the computational thinking standard of the computer science teachers’ association, he developed one computational thinking tool to evaluate grade 5 students. According to the above research, programming language instruction by robots leads to students’ learning challenge and enhances their computational thinking competences [6]. In Leonard’s study of robots and game design, the said researcher enhanced students’ self-efficacy, STEM attitude and computational thinking competences. A study involving 124 secondary school students demonstrated that game design by robots significantly influenced students’ computational thinking competences [7]. Chaudhary et al. explored the effects of LEGO robots on program design and computational thinking experience. They demonstrated that program design instruction reinforced students’ comprehension of skill and knowledge and problem-solving [8]. Yindi analyzed cultivation of students’ computational thinking competences though Visual Basic programming and realized that students’ learning became more active and students were capable of solving real problems with professional knowledge [9]. Parmar implemented program design instruction in virtual environment to evaluate students’ computational thinking competences. According to the findings, learning computer science concept and programming language in context of virtual environment can effectively trigger students’ motivation and interest [10].

According to previous literature, computational thinking is not a type of programming language. It can be applied to explain and describe calculation, abstraction and information terms. Hemmendinger argued that the objective of computational thinking is to teach students to as economists, physicists and artists and realize how to solve, create and discover problems by arithmetic [11].

2.2 Situated Learning

The cultivation of situated learning has become more important in recent years. Professional talents should cooperate with each other and solve problems in different environments and situations [12], [13]. Lave and Wenger defined situated learning as follows: “direct observation in the field and learning of various operations in exchange and interaction” [14]. Situated learning was first proposed by Brown, Collins and Duguid in 1989. It is an instructional model developed upon theory of modern cognitive psychology. The basic assumption is to allow students to learn in real, or simulated, situations. They simulate real environment in the instruction of course to allow students to be engaged in instructional situations and learn more efficiently.

Stein defined four key factors of situated learning environments, namely Content: task and process which learners must execute; Scenario: situated and environmental clues which surround the learners and meaningful support in creative process; Community: learners will create and negotiate the situations with them; and Participation: learners work together and with experts and in one social group, they solve problems related to daily life environments. Learning becomes a social process. It depends on interaction with others in the same situation. The interactions should be similar to those of a practical environment [15]. Chen and Lin applied situated learning of a digital game to instruction of Chinese poetry. Using Chinese poetry of the Tang Dynasty as an example, they developed animations to simulate poets' obstacles in writing and assisted with Chinese poetry learning of junior high school students in Taiwan [16]. According to the above research, situated learning positively influenced students' perception. In College of Education of University of Zaragoza, Bravo-Alvarez et al. applied situated learning in two obligatory Early Childhood Education subjects. According to the experiment, students were highly satisfied with the learning experience of situated learning and it enhanced students' judgement competences [17]. Lin et al. introduced situated learning in research on English drama education of 5th graders in elementary school. Interviews with teachers and analysis of in-class video recording showed that the instruction reinforced students' motivation to participate in group learning activities [18].

According to the previous literature, situated learning is an effective instructional tool for students to comprehend concepts in school. By situated learning, students recognize correlation and importance of course content and more actively participate in learning. More importantly, they apply knowledge in reality. Hence, computer science education should break through the past boundary on information technology use, emphasize content of computer science, strengthen students' comprehension of computer science content, cultivate their logic thinking and problem-solving competence, introduce learning in daily lives to trigger students' learning interest and satisfy basic living needs of the digital era.

2.3 Computational Thinking and Bebras

Bebras is an international competition of informatics and computer literacy. It was launched in Lithuania in 2003. Early planning of task construction and preparation of implementation of Bebras lasted for nearly one year. The first competition was in held in October 2004. Objective of this organization is to turn Bebras into one international competition. In May, 2005, Olympiad of informatics was held in Lithuania [19]. During the spring of 2006, Lithuania held two international Bebras conferences and founded the Bebras international organization committee [20]. The main purpose of the conference is to construct a series of questions for future Bebras competitions.

Students use computer and technology every day. Though the assistance of technology, some students further develop their problem-solving comprehension. However, when applying technology, students need thinking skills to solve daily problems. The best measure is to develop the computational thinking competences. Students' daily problem-solving competences is the most significant issue according to educators and decision makers [21, 22]. When teaching informatics and computer science literacy through problem solving, it is important to stimulate students' learning through interesting questions. Therefore, one should propose questions from by various fields of science and daily life. The most common questions of computer science are application of daily lives and general questions of history, language, art and even mathematics. It is extremely important to select questions that students can solve with the same opportunities using different operating systems or computers.

In the second international Bebras conference in 2006, a brainstorming meeting was held to propose ideas on different types of questions applied in the competition. More than 120 tasks were developed at the conference and around 90 of these were accepted and elaborated further. The issues are categorized below [3], including general logic thinking, information and communication technology in daily lives, real and technology problems, information comprehension, calculation and program design, basic mathematics of computer science, history, to mention a few.

According to previous literature, the Bebras learning model focuses on concept of informatics which supports comprehension of information science phenomenon and development of computational thinking. Therefore, this study cultivates students' computational thinking competence through situated learning regarding students' competence of technology, logic thinking and problem solving. Among others, students' computational thinking competence is evaluated through the Bebras questions.

3 Methodology

This study explored effects of computational thinking competences on Bebras test performance by senior high school students' who learned program design using a situated learning strategy.

3.1 Participants

The participants comprised second-year students who were in the Life technology course in one senior high school in Taiwan. These students had not yet been taught computational thinking. Hence, they were beginners in computational thinking. Two second-year classes were selected. The experimental group included 20 students and the control group included 19 students. There were a total of 39 students. Each student was given one computer and one Webduino instructional tool.

3.2 Procedure

The experiment lasted 10 weeks. There were 50 minutes in one session. Week 1 of the experiment included the international computational thinking competence pretest; Week 2 and 3 covered basic operation instruction of Webduino instructional tool and the purpose was to allow students to be familiar with operation of the Webduino instructional tool to successfully proceed with experimentation. Week 4 to 9 covered the Webduino instructional tool. The experimental group learned using a situated learning strategy, while the control group followed a traditional course. The experiment lasted for 6 weeks. During week 10 the international computational thinking competence posttest was conducted. After the course, the data was analyzed and documented (see Figure 1).

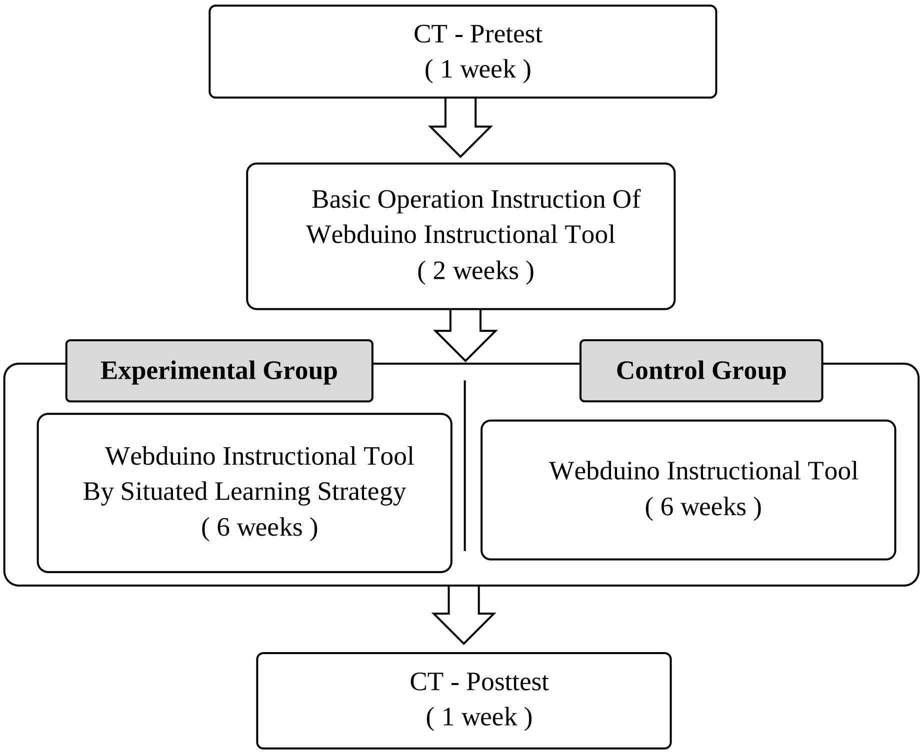

Fig. 1. Experimental Procedure

3.3 Research Tool

3.3.1 Examination Questions of International on Informatics and Computational Thinking

To recognize students' computational thinking competence before the instructional experiment, this study conducted computational thinking competence tests to avoid effects of learners' prior knowledge. Pretest items were those of Senior (grade 11 and 12) group of the 2015 international computational thinking competence test; posttest items were those of Senior (grade 11 and 12) group of the 2016 international computational thinking competence test. The purpose of the test was to trigger students' learning interest in information science, reinforce students' competence to solve problems through information measures and avoid students' fear of information science. The test was scored by difficulty degree of items. Scores of correct answers were added and wrong answers were deducted. The omitted answers were not scored. To avoid negative scores, the starting point of the test was the total of deduction of items. There were totally 15 items. Five items were equally distributed at different degrees of difficulty. The starting point was 60, the lowest was 0 and the highest was 300, as shown in Table 1.

Table 1. The Scoring rules of Examination Questions

Group	Difficulty						Questions	Starting score	Highest score
	C		B		A				
	Correct	InCorrect	Correct	InCorrect	Correct	InCorrect			
Senior	12	-3	16	-4	20	-5	15	60	300

3.3.2 Webduino Fly Microcontroller

This study adopted a Webduino Fly microcontroller unit developed by KING JIH of Taiwan in 2015. It was cloud extension board of Arduino UNO. By assistance of Webduino Fly, users could possess complete Webduino development function through Wifi. Without complicated hardware techniques, they could control Arduino compatible sensor modules [25]. In addition, this study also adopted the Webduino electronic component package with a learning sheet. The Webduino electronic component package includes 13 electronic component modules and 7 sensors. The components and sensors makes the course more diverse and interesting.

3.3.3 Worksheet

This study expected that students can recognize important relationship between subjects and lives thorough computational thinking learning in the course. It constructs knowledge in real activities and combines activity and culture to allow learners to develop knowledge by interacting with scenarios. It emphasizes that learners make progress in situated context of life and use knowledge as a tool in the scenarios. Likewise, the Ministry of Education repetitively emphasizes that life related practice learning activities and course instruction should be combined with life situations with significance. The learning sheet of the experimental group included three parts. Part 1 scenario, Part 2 practice of examples and Part 3 extended questions; The control group had the same apart from Part 1 scenario.

4 Results

A one-way ANOVA were performed with IBM SPSS. According to the homogeneity test results shown in Table 2, two groups' p value of computational thinking competence is 0.638 and higher than the significance level of .05. The null hypothesis is thus accepted. It shows that the slope of two groups' regression lines are the same. In other words, the relationship between covariance (pretest score of computational thinking competence) and dependent variable (posttest score of computational thinking competence) are not different because of different processing levels of independent variable. It meets the hypothesis of homogeneity of in-group regression coefficient in analysis of covariance. Analysis of covariance can be further conducted.

Table 2. Two Groups of Regression Coefficient Homogeneity Test Summary

Item	F	df	*P*
Computational Thinking Skill	.225	1	.638

Table 3 shows the analysis of the covariance of two groups' computational thinking competence. According to the results, after eliminating the effect of covariance (pretest score of computational thinking competence) on the dependent variable (posttest score of computational thinking competence), it shows that experimental effect of two groups' learning of independent variable (experimental group and control group) on total score of computational thinking competence is significant. ($F(1, \ldots) = 84.705$, $p < .001$). It reveals that after receiving situated learning strategy, experimental group's overall computational thinking competence is superior to that of control group.

Table 3. Two Groups one-way ANCOVA Summary

Item	SS	df	MS	F	*p*
Computational Thinking Skill	51717.093	1	51717.093	84.705	.000***

5 Conclusion

The results confirm the importance of situated learning strategy on the cultivation of students' computational thinking competences. The experimental group's posttest score of computational thinking is significantly larger than that of the control group..

There are several possible factors that have resulted in the experimental group's higher posttest score of computational thinking. First, learning sheet items are simulation of realistic situations. Teachers simulate real environments in class to allow students to be involved in the instructional scenario which lead to more efficient learning. The classroom resembles a learning community and teachers and students collectively create and discuss real daily situations to solve problems. Situated learning can effectively result in knowledge construction. Brown et al. stated the importance of situated learning. Situated learning should be established in real situations and students develop knowledge by interacting with situations. Situated learning emphasizes that students are in real situational context and use knowledge as tool in realistic scenarios [4]. Likewise, computational thinking competences implies effective problem solving using information technology. Thus, the thinking process is needed to analyze problems that result in answers. Computer science education should break through the past boundary of information technology use, emphasize content of computer science, reinforce students' comprehension of computer science content, cultivate their logic thinking and problem-solving competence, introduce learning in daily lives to trigger students' learning interest and satisfy their basic life needs in digital era.

This study is limited to Second-year senior high school students. Thus, it is necessary to expand the scope of the study to include other student groups. Future research will validate various advantages to apply situated learning strategy.

Acknowledgments.

This research is partially supported by the Ministry of Science and Technology (MOST), Taiwan, R.O.C. under Grant MOST 104-2511-S-224-003-MY3, MOST 105-2628-S-224-001-MY3.

References

1. Resnick, M: Learn to code, code to learn. *EdSurge, May* (2013)
2. Chen, G., Shen, J., Barth-Cohen, L., Jiang, S., Huang, X., Eltoukhy, M.: Assessing elementary students' computational thinking in everyday reasoning and robotics programming. *Comput. Educ.* **109**, 162–175 (2017)
3. Opmanis, M., Dagiene, V., Truu, A.: Task types at 'Beaver' contests, Inf. Technol. Sch., pp. 509–519 (2006)
4. Brown, J. S., Collins, A., Duguid, P.: Situated cognition and the culture of learning, Educ. Res. **18**, 32–42 (1989)
5. Wing, J.: Research notebook: Computational thinking—What and why? The Link Magazine, Spring. Carnegie Mellon University, Pittsburgh (2011)
6. Chen, G., Shen, J., Barth-Cohen, L., Jiang, S., Huang, X., Eltoukhy, M.: Assessing elementary students' computational thinking in everyday reasoning and robotics programming. Comput. Educ. **109**, 162–175 (2017)
7. Leonard, J., Buss, A., Gamboa, R., Mitchell, M., Fashola, O. S., Hubert, T., Almughyirah, S.: Using Robotics and Game Design to Enhance Children's Self-Efficacy, STEM Attitudes, and Computational Thinking Skills. J. Sci. Educ. Technol. **25**, 860–876 (2016).
8. Chaudhary, V., Agrawal, V., Sureka, P., Sureka, A.: An Experience Report on Teaching Programming and Computational Thinking to Elementary Level Children using Lego Robotics Education Kit. in 2016 IEEE 8th International Conference On Technology For Education (T4E 2016), pp. 38–41 (2016)
9. Yindi, D.: Visual Basic Program Designing Based on Computational Thinking Capabilities Training, in Proceedings Of The 2nd Information Technology And Mechatronics Engineering Conference (ITOEC 2016), vol. 24, pp. 175–178 (2016)
10. Parmar, D., Isaac, J., Babu, S.V., D'Souza, N., Leonard, A. E., Jorg, S., Gundersen, K., Daily, S. B.: Programming Moves: Design and Evaluation of Applying Embodied Interaction in Virtual Environments to Enhance Computational Thinking in Middle School Students. in 2016 IEEE Virtual Reality Conference (VR), pp. 131–140 (2016)
11. Hemmendinger, D.: A plea for modesty, Acm Inroads **1**, 4–7 (2010)
12. Gardiner, L. R., Corbitt, G., Adams, S. J.: Program assessment: Getting to a practical how-to model, J. Educ. Bus. **85**, 139–144 (2009)
13. Meyers, S., Lester, D.: The Effects of Situated Learning Through a Community Partnership in a Teacher Preparation Program, SAGE Open, vol. 3, no. 3, p. 2158244013497025 (2013)
14. Lave, J., Wenger, E.: Situated learning: Legitimate peripheral participation. Cambridge university press (1991)
15. Stein, D.: Situated Learning in Adult Education. ERIC Digest No. 195. (1998)
16. Chen, H.-R., Lin, Y.-S.: An examination of digital game-based situated learning applied to Chinese language poetry education. Technol. Pedagog. Educ. **25**, 171–186 (2016)

17. Angeles Bravo-Alvarez, M., Escolano-Perez, E., Navas-Macho, P., Luisa Herrero-Nivela, M., Eguinoa-Zaborras, F., Acero-Ferrero, M.: A Situated Learning Experience: Student Satisfaction And Improvement Of Their Assessment Skills In Joint Reference Disorder. in ICERI2015: 8th International Conference Of Education, Research And Innovation, pp. 7819–7825 (2015)
18. Lin, W.-C., Huang, D.-Y., Huang, C.-W., Liu, Y.-C., Chen, G.-D.: A Video Comic For Group Situated Learning In The Classroom, in Edulearn15: 7th International Conference On Education And New Learning Technologies, pp. 4264–4271 (2015)
19. Dagienė, V. Futschek, G.: Bebras international contest on informatics and computer literacy: Criteria for good tasks, Informatics Educ. Comput. Think., 19–30 (2008)
20. Verhoeff, T., Horváth, G., Diks, K., Cormack, G.:2"A proposal for an IOI syllabus. Teach. Math. Comput. Sci. **4**, 193–216 (2006)
21. Dagiene, V., Skupiene, J.: Learning by competitions: olympiads in informatics as a tool for training high-grade skills in programming. in ITRE 2004. 2nd International Conference on Information Technology: Research and Education 2004, pp. 79–83 (2004)
22. Dagiene, V.: Information technology contests-introduction to computer science in an attractive way. Informatics Educ. **5**, 37 (2006)