Graduate School of Fundamental Science and Engineering
Waseda University

博 士 論 文 概 要

論 文 題 目

Study on the Learning Effects Based on Characteristics in Programming Learning Environments for Novice Learners
初学者向けプログラミング学習環境の特性に基づく学習効果に関する研究

申 請 者

Daisuke	SAITO
齋藤	大輔

情報理工・情報通信専攻　高信頼ソフトウェア工学研究

2017 年 10 月

Novice learners often use programming learning environments when learning to programme. Novice learners are children aged 6 to 12 who are inexperienced in programming learning. Examples of programming learning purposes include computer science learning and mathematical learning. It is also used to develop problem solving abilities and abstraction abilities.

Thee programming learning environments have various characteristics. In this research, I identify characteristics of programming learning environments and investigate the learning effect based on the characteristics.

Numerous programming learning environments are utilized for novice learners. First, I investigated the kinds of programming learning environments. I employed a Google Custom Search API with specific keywords. As a result, I got 800 search results. Then I extracted the programming learning environments by morphological analysis and visual observations, yielding over 70 environments for programming learning as examples, Scratch, Alice, and Greenfoot are used in a visual programming language, while CodeCombat and Minecraft Education Edition exist in game software.

Previously, Kellaher et al. classified multiple programming environments, demonstrating that these environments have unique characteristics. However, environments including learning environments continue to be developed. Learning effects in programming learning environments have been shown in several studies. Several studies have shown that the learning environment called Scratch is suitable to improve learners' interest and passion for programming. Other studies have shown that using a game called Minecraft tends to improve programming skills. These studies suggest that the learning effect may depend on the method of programming learning and the learning environment. However, the issue is that these environments are used at the discretion of educators and learners. Moreover, it is unclear what kinds of learning effect are derived from the characteristics of each learning environment. Investigating the characteristics of learning environments should reveal the learning effect. Hence, I consider the programming learning environment for novice learners and investigate the learning effects based on characteristics. This can be used as information to maximize the learning effect of the novice learners.

Furthermore, the research question is “How can novice learners maximize learning effects in programming learning? “. Additionally, the goal of this research is to clarify the learning effect by grasping the characteristics of the programming learning environment. A novice learner can improve learning by achieving this goal.

Chapter 1 highlights that there are over 70 kinds of programming learning environments, leading to issues with programming learning environments. Additionally, I explain the my research outline and research goals.

Chapter 2 explains the taxonomy to evaluate multiple programming environments and shows the classification results based on the taxonomy. I create a taxonomy by defining items to classify programming learning environments by referencing Kelleher et al. Specifically, I optimized Kelleher's table for learning environments and added the category. The taxonomy table divides the 56 items into 11 categories.

Then I applied the taxonomy to classify several programming learning environments. Based on the results, I evaluated the characteristics in each environment, including the attributes of visual programming language environment and game software. I surveyed 43 kinds of environments with an emphasis on visual language and software that works alone on PCs or other devices to create a taxonomy table for programming learning environments. The proposed table can evaluate and compare such environments. The experiment confirms that the classification and evaluation results are independent of the evaluator. Therefore, this classification table helps users (learners and educators) to know the characteristics of the programming learning environment.

Chapter 3 investigates the learning effect as a function of characteristics in the same environment. Herein the differences between visual and text input methods (Representation of code and Construction of programs) are investigated in the same Lua programming environment to determine if the input method influences the learning effect. There are many visual and text comparative studies. However, investigations including characteristics such as linguistic representation are scanty. This characteristics difference should affect learning effect. Specifically, I compare a combination of text (Representation of code) and typing code(Construction of programs) with a combination of image (Representation of code) and drag-and-drop(Construction of programs).

The results indicate that a visual input method is better suited for a novice to programming learning. However, the comparison results suggest that actions change the learning effect. Hence, the text input method can be used for programming learning of novice learners from the viewpoints of the representation of code and construction of programs in the programming environment.

Chapter 4 investigates the characteristics of multiple environments as well as the learning effect of each environment. In this chapter, I consider the learning effect based on the characteristics of programming constructs and game elements, in addition to the characteristics discussed in Chapter 3.

I conducted a quantitative evaluation by a workshop on six programming learning environments. The characteristics of the classification influence the learning effect. However, if the software involves "physical object" and "assembling physical objects," the learner may become bored as the workload increases. The three groups (visual programming language, game software, and physical environment) show a difference in attitude toward programming. The visual programming language tends to soften programming difficulty. Although environments with game elements tend to make programming more fun, they also increase perceived difficulty of programming.

Chapter 5 summarizes this thesis. In addition, I explain the future research,

In the future research, the three main areas are to “propose and create a programming learning environment,” “optimize the characteristics and functions of the taxonomy table” and “create guidelines to select the appropriate programming learning environment.” As one future work, I am proposing an environment to predict learning effect from characteristics. I tried developing an environment that considers the learning effect. The proposed environment is an extension based on an existing environment. In this work, only partial environments or prototypes are implemented. Currently, I am working on expanding the function of this environment.

早稲田大学　博士（工学）　学位申請　研究業績書

齋藤大輔　　印

（2018 年 2 月　2 日現在）

種 類 別	題名、　発表・発行掲載誌名、　発表・発行年月、　連名者（申請者含む）
論文 ○	Daisuke Saito; Hironori Washizaki and Yoshiaki Fukazawa, Comparison of Text-Based and Visual-Based Programming Input Methods for First-Time Learners, Journal of Information Technology Education: Research, Vol. 16, Informing Science Institute, Jun. 2017, pp. 209-226. Daisuke Saito and Tsuneo Yamaura, A New Approach to Programming Language Education for Beginners with Top-Down Learning, International Journal of Engineering Pedagogy, Vol. 3(S4), International Society of Engineering Education, Dec. 2013, pp. 16-21.
講演 (国際会議) ○	Daisuke Saito; Ayana Sasaki; Hironori Washizaki; Yoshiaki Fukazawa and Yusuke Muto, Quantitative Learning Effect Evaluation of Programming Learning Tools, Teaching, Assessment, and Learning for Engineering (TALE), 2017 IEEE International Conference on. IEEE, Dec. 2017, pp. 209-216, Hongkong, Chania.
○	Daisuke Saito; Ayana Sasaki; Hironori Washizaki; Yoshiaki Fukazawa and Yusuke Muto, Program Learning for Beginners: Survey and Taxonomy of Programming Learning Tools, Engineering Education (ICEED), 2017 IEEE 9th International Conference on. IEEE, Nov. 2017, pp. 137-142, Ishikawa, Japan.
○	Daisuke Saito; Hironori Washizaki and Yoshiaki Fukazawa, Analysis of the Learning Effects Between Text-based and Visual-based Beginner Programming Environments, Engineering Education (ICEED), 2016 IEEE 8th International Conference on, IEEE, Dec. 2016, pp. 208-213, Kuala Lumpur, Malaysia.
○	Daisuke Saito; Hironori Washizaki and Yoshiaki Fukazawa, Influence of the Programming Environment on Programming Education, Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education, ACM, Jul. 2016, pp. 354-354, Arequipa, Peru.
○	Daisuke Saito; Hironori Washizaki and Yoshiaki Fukazawa, Work in progress: A Comparison of Programming Way: Illustration-based Programming and Text-based Programming, Teaching, Assessment, and Learning for Engineering (TALE), 2015 IEEE International Conference on, IEEE, Dec. 2015, pp. 220-223, Zhuhai, China. Daisuke Saito; Akira Takebayashi; Tsuneo Yamaura: Hironori Washizaki and Yoshiaki Fukazawa, An Evaluation and Result or a Workshop Using Minecraft for ICT Education. Replaying Japan 2015: 3rd International Japan Game Studies Conference, May. 2015. Kyoto, Japan.

早稲田大学　博士（工学）　学位申請　研究業績書

種 類 別	題名、　発表・発行掲載誌名、　発表・発行年月、　連名者（申請者含む）
講演 （国内会議）	Daisuke Saito and Tsuneo Yamaura, Applying the Top-down Approach to Beginners in Programming Language Education, Interactive Collaborative Learning (ICL), 2014 International Conference on, IEEE, Dec. 2014, pp. 311-318. Dubai, UAE. Daisuke Saito; Akira Takebayashi and Tsuneo Yamaura, Minecraft-based Preparatory Training for Software Development Project, Professional Communication Conference (IPCC), 2014 IEEE International, IEEE, Oct. 2014, pp. 1-9, Pittsburgh, USA. Daisuke Saito; Akira Takebayashi; Taiki Nizuma; Renato Nojiri and Tsunao Yamaura, Minecraft-based Communication Learning to Elementary School Students and Junior High School Students, Replaying Japan 2014: 2nd International Conference on Japanese Game Studies, Aug. 2014, pp. 30, Edmonton, Canada. Daisuke Saito and Tsuneo Yamaura, A New Approach to Programming Language Education for Beginners with Top-down Learning, Teaching, Assessment and Learning for Engineering (TALE), 2013 IEEE International Conference on, IEEE, Aug. 2013, pp. 752-755, Bali, Indonesia. 齋藤 大輔; 鷲崎 弘宜, Python を含む複数のプログラミング言語の初学者向け学習環境の特性・特徴の分析, PyCon JP 2017, 2017 年 9 月, 東京都. 齋藤 大輔, 第四次産業革命としごとの在り方」- こども向けプログラミング学習の観点から-, 2017 年度 第 1 回しごと能力研究学会部会・研究会, しごと能力研究学会, 2017 年 8 月, 宮城県. 佐々木 綾菜; 鷲崎 弘宜; 齋藤 大輔; 深澤 良彰; 武藤 優介; 西澤 利治, 小学校におけるプログラミング教育において活用可能なルーブリックの提案, 日本デジタル教科書学会第６回年次大会, 日本デジタル教科書学会, 2017 年 8 月, pp. 33-34, 東京都.
○	齋藤 大輔; 佐々木 綾菜; 鷲崎 弘宜; 深澤 良彰; 武藤 優介. 初学者向けプログラミング学習ツールにおけるゲームソフトウェアの調査と分類, 日本デジタルゲーム学会 2016 年度 年次大会, 日本デジタルゲーム学会, 2017 年 3 月, pp. 51-54, 愛知県. Daisuke Saito, Learn to Program with Minecraft: A Comparison of the Effects of Learning with Programming Methods, PCS-J 2nd Technical Meeting and General Assembly 2015, IEEE Professional Communication Society Japan Chapter, 2015 年 12 月, 長野県.

早稲田大学　博士（工学）　学位申請　研究業績書

種 類 別	題名、　発表・発行掲載誌名、　発表・発行年月、　連名者（申請者含む）
著書	齋藤大輔. Minecraft で楽しく学べる Python プログラミング. ソーテック社. 2017 年 6 月 10 日. ISBN: 978-4800711656. 松尾 高明; 齋藤 大輔; ナポアン; nishi. みんな大好き！マインクラフト るんるんプログラミング！コマンドブロック編. ソシム. 2017 年 3 月 21 日. ISBN: 978-4802610780.