

A DESCRIPTION-TYPE PROGRAMMING LEARNING SYSTEM WHICH ADVISES ABOUT MAIN WRONG ANSWERS

SHUN SASAKI AND HIROSHI KAMADA*

Graduate Program in System Design Engineering
Graduate School of Engineering
Kanazawa Institute of Technology

3-1 Yatsukaho, Hakusan, Ishikawa 924-0838, Japan

b6901796@planet.kanazawa-it.ac.jp; *Corresponding author: kamada@neptune.kanazawa-it.ac.jp

Received March 2021; accepted May 2021

ABSTRACT. *In recent years, information technology has made progress and contributes to improving the convenience of society. While the scale of the IT market is expected to expand in the future, the shortage of human resources who can utilize IT technology is becoming more serious worldwide. By incorporating programming as a compulsory education from such problems, efforts to develop human resources at an early stage are being implemented worldwide. In order to spread programming to the younger generation, it is important to devise ways to increase learning motivation and encourage continuous learning. In this research, we devised a learning system that outputs the answer results of the learner as image and video data based on the learning system devised in the conventional research, and makes correct/incorrect judgment and advice based on the output data, and learns the same. The effect was evaluated. From the results, it was found that the use of images and videos makes it easier to visually recognize their own achievements and improves their motivation for learning. It was also found that by utilizing the advice function, the learner's understanding was deepened and the learning effect was brought to the user.*

Keywords: Programming, Learning system, Image comparison, Advisory function

1. Introduction. In recent years, information technology has made rapid progress worldwide, contributing to the improvement of convenience in social activities and daily life. In particular, the development of AI technology is remarkable, and it is expected to be used in various fields. With the development of information technology, the scale of the IT market will continue to expand in the future, while the shortage of human resources who can utilize IT technology is becoming more serious worldwide. In Australia, the digitization of the economy will create 40,000 new jobs between 2016 and 2017, and demand for IT talent is expected to grow 2.7% annually through 2023 [1]. As a result, Australia will need 100,000 people by 2023 to become a global leader in IT. Due to these problems, efforts are being made worldwide to develop IT human resources at an early stage by incorporating programming as a compulsory education [2]. In the UK, the 2013 National Curriculum stipulates that all students aged 5 to 16 will be required to take a computing lesson that incorporates understanding algorithms and learning programming languages. In Hungary, the class named ‘Informatika’ that utilizes programming and ICT, has been introduced as a compulsory subject in primary and secondary education since 2003. In primary education, classes centered on IT literacy are conducted, and in secondary education, classes centered on programming are conducted. In India, ICT or computer science has been incorporated into education since 2005, and computer classes, including programming from primary education, have been conducted. With the global promotion of programming education, it is expected that the public interest in programming will

increase and the number of programming learners will increase rapidly. On the other hand, there is concern that the number of learners who are not good at programming and give up learning will increase. In particular, the description method using a programming language has many complicated points, so it is considered that the percentage of learners who give up learning is high. Therefore, in conducting programming learning, a system that motivates learners and promotes continuous learning is required.

The purpose of this study is to realize a learning support system for beginners of programming learning that aims to improve learners' motivation for learning and promote continuous learning. In addition, programming learning is generally divided into algorithm creation ability and programming language description ability, but in this research, we focus on learning programming language description ability.

Conventional system and its problems are described in Section 2, our methods to solve conventional problems are described in Section 3, the system is described in Section 4, evaluation experiment is described in Section 5, and we conclude in Section 6.

2. Conventional System and Its Problems. As a conventional study on programming learning, there is an example of using the visual programming language Scratch in primary education [3-5]. Developed by MIT Media Lab, Scratch [6] is an easy-to-use programming learning software that is ideal for children because it allows you to assemble instructions with a simple drag-and-drop mouse operation and visually grasp the results. Previous studies have shown significant improvements in the practical learning of programming concepts, logic, and computation by incorporating Scratch into primary education. Therefore, it emphasizes the usefulness and importance of implementing a visual programming language in the primary education curriculum. On the other hand, more serious programming such as creating websites and applications is often difficult to express visually, so visual programming languages are not suitable for complicated processing. Therefore, since what can be developed is limited, learning by a text programming language is required to realize more complicated processing.

As another study on programming learning using visual expressions, a learning support system was devised that utilizes the text programming language Processing, displays the learner's answer results in images and videos, and judges whether they are correct or not [7,8]. In these systems, the correctness judgment is made by comparing the pixel values of the answer image and the correct answer image prepared in advance. In the conventional learning support system, the difference in the output result can be visually understood when an incorrect answer is given by displaying the images side by side after the correctness judgment. However, since it is unclear how to modify the program, there is a problem that it is difficult to prompt the learner to answer again.

3. Our Methods to Solve Conventional Problems. Since visual programming languages such as Scratch are not suitable for practical programming learning, we will develop a learning system using a text programming language in this research. In addition, in the learning system [7,8] that outputs the answer result of the learner as an image and determines the correctness, the correctness of the answer can be confirmed by comparing the pixel values of the two images of the learner's answer and the correct answer. It was possible, and by displaying the images side by side, it was possible to visually understand the difference from the correct answer image. However, by simply displaying the images side by side, if the learner makes an incorrect answer, he/she can understand that his/her own answer is incorrect, but it is difficult to understand how to specifically modify the program. We could not urge the person to answer again. Therefore, it is considered necessary to have a mechanism that can present the correction points of the program to the learner as advice so that the learner can be guided to the correct answer based on the image data output by the learner's wrong answer. Therefore, in this research, we

devised a learning system that implements an advisory function using a group of incorrect answer data. Figure 1 shows the solution for this system. For the task, prepare an answer image of the predicted wrong answer in addition to the correct answer image. Then, the advice text for correcting the wrong answer program is linked to the wrong answer image. Multiple patterns of this data set are prepared and used as a data group. The image output by the learner's answer is compared with the incorrect answer image registered in the data group, and if they match, the linked advice text is presented to the learner.

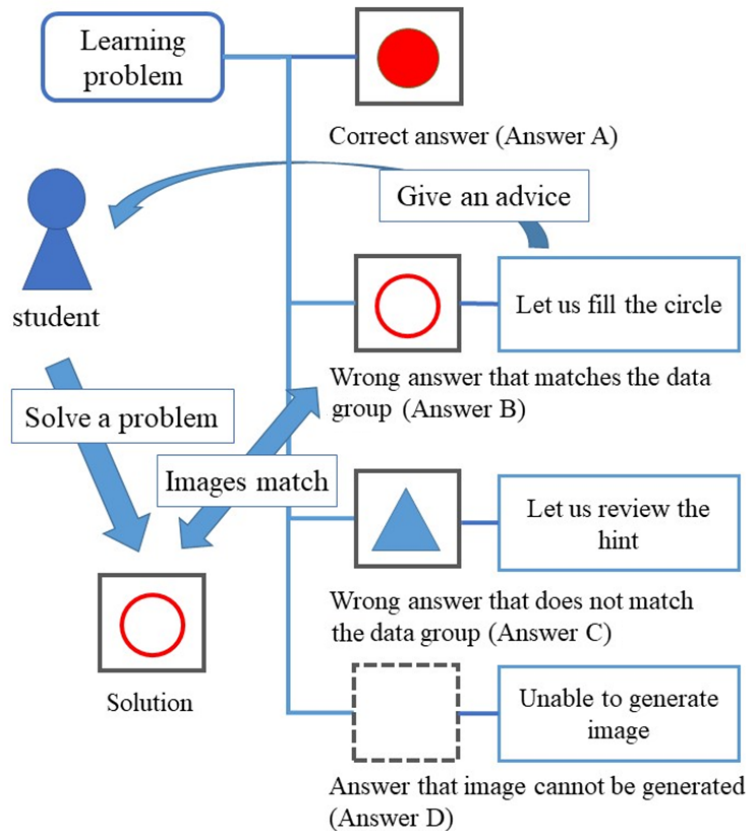


FIGURE 1. Advice function using incorrect answer data group

4. **The System.** In this system, two types of answer formats, selection type and descriptive type, are implemented. In the selection formula, three programs including the correct answer program are displayed, and the answer is given by moving the cursor to the option and pressing the mouse. Regarding the choices, two wrong answers were randomly extracted from the correct answer choices and the wrong answers in the data group, and the two wrong answers were sorted and displayed. This prevents the learner from giving duplicate answers. In the descriptive formula, a text box is displayed, and after inputting the answer, the answer button is pressed to give the answer. The selection formula is less difficult than the descriptive formula, and the answer is given by operating the mouse, so even a learner who is not good at operating the keyboard can easily tackle the problem. Since the answer candidates are not displayed in the descriptive formula and the answer is given using the keyboard, it is more difficult than the options and it is considered that practical skills can be acquired. On the question answer screen, the question sentence is displayed in the upper left of the screen, and the program sentence including the answer content is displayed in the lower left. In the area in the center of the screen, an image of a sample that is the correct answer and an image of the execution result of the program that the learner answered are displayed. In addition, a hint button has been installed in the upper right corner of the screen so that you can refer to hint sentences that help

you learn while answering. The answer is interrupted while referring to the hint, and the answer function can be used by closing the hint sentence. In this way, it is considered that the learner can answer the hint by ruminating it with his/her own head.

Figure 2 shows the answer screen of this system.

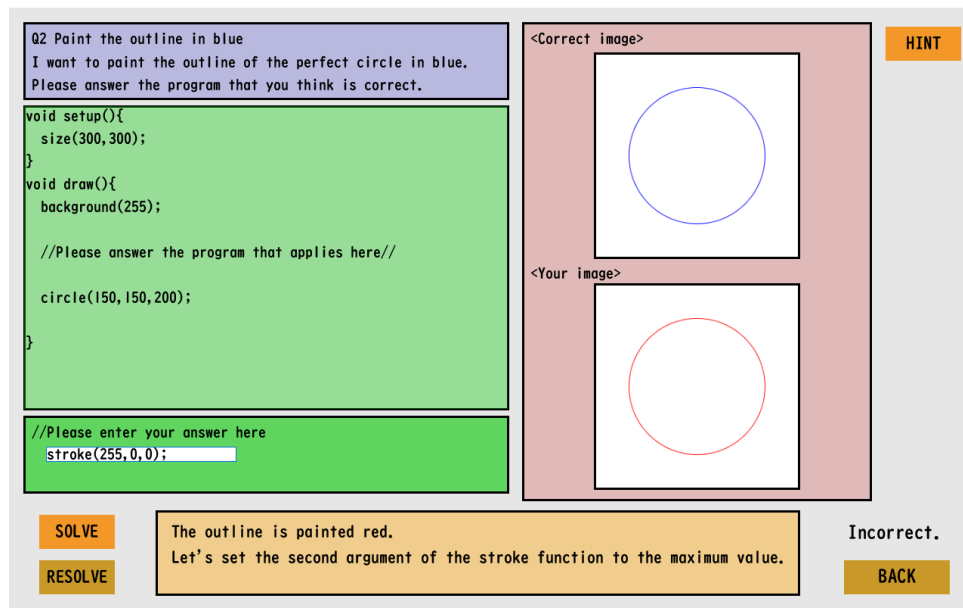


FIGURE 2. Answer screen of this system

Correctness judgment is performed by comparing the pixel values of the output image and the image of the data group. Pixels with a width of 270px \times a height of 270px = 72900px are compared, and if all the values are equal, it is determined that the images of the data group match. Also, in the case of the problem of outputting a moving image, the moving image is expressed by generating the data of 150 still images or preparing them in advance and displaying them continuously. Therefore, when determining the moving image, 150 patterns of comparison of the pixel values of 72900px are performed, and it is determined that the moving images match when the values are the same in all the patterns. By making a correct/incorrect judgment, the learner's answer results are roughly classified into four patterns as shown in Figure 1. Generate an answer that matches the correct image (Answer A), an answer that matches the incorrect image in the data group (Answer B), an answer that does not match the incorrect image in the data group (Answer C), and an image/video. It is an answer that cannot be done (answer D). When the output answer is judged to be answer B in the correctness judgment, the learner is presented with the correction points of the answer program as advice. Advice is given by displaying the character string of the text file associated with the image/video stored in the data group after answering. In the case of answer A, it is not necessary to modify the program, but the correct answer program will be explained using the same mechanism to deepen the learner's understanding of the answer. In the case of answer C or D, since there is no image that matches the data group, a certain text prompting the answer is displayed as advice. Therefore, when the answer C or D is output, it is not possible to give appropriate advice to the learner's answer. When you select a descriptive formula in the answer format and answer, you can output an image or video, but the answer does not exist in the data group, that is, in the case of answer C, the input answer program and the incorrect answer images are automatically integrated into the wrong data group. By adding text data as advice to the accumulated data and reflecting it in the system, it is possible to increase the number of incorrect answer patterns. However, the developer must manually add the advisory text. Since the options other than the correct answer displayed

when answering the selection formula are drawn from the incorrect answer pattern of the data group, the range of options can be expanded by accumulating the data. Figure 3 shows the mechanism of the data collection function.

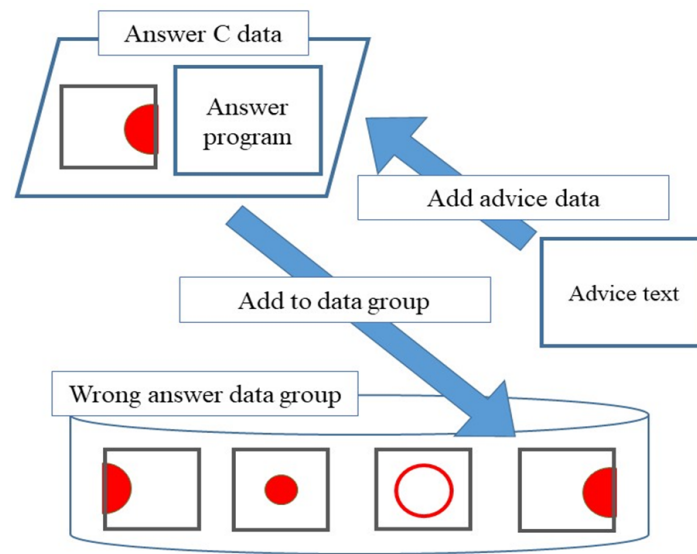

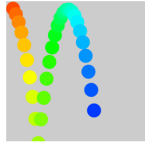
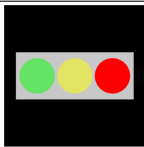


FIGURE 3. Data accumulation function

This system used Processing as a learning language. Processing is a programming language for electronic art and visual design that simplifies the Java language and specializes in graphic functions [9]. Since visual feedback can be obtained immediately, it is considered that it can be used for correctness judgment and advice function by images/videos. In addition, it is considered to be a language suitable for beginners to learn, and it is considered to be suitable for learning another language as the next step.

In order to encourage learners to continue using it, we set a task to create a simple video by referring to the learning items [10] of the Vector of Nature of Code. Although it is simple, since multiple steps are required to create a video, it is considered difficult to answer the question of the video by itself. Therefore, by dividing the task into multiple processes and setting the tasks in order, the load on the learner’s answer was reduced. Since each of the three issues was divided into four processes, a total of 12 questions were implemented in this system. Table 1 shows a list of implemented issues and processes.

TABLE 1. Video problems and creation process

| Problems | Creation process | Correct answer image |
|---------------------------|---|---|
| Ripples drawing | Drawing of shapes Color settings Use of variables Iterative processing |  |
| Of the ball jump | Drawing of shapes Color settings Use of operators Conditional branch |  |
| Lighting of traffic light | Drawing of shapes Color settings Use of variables Conditional branch |  |

5. Evaluation Experiment. In order to measure the learning effect of this system, an evaluation experiment was conducted on 27 university and graduate students. In the evaluation experiment, four steps of pre-test, learning, post-test, and questionnaire were carried out in order. In the pre- and post-test, 10 questions of 10 points are given per question, and the score is given up to 100 points. In order to verify the usefulness of the learning effect of this system, statistical hypothesis testing is performed using the results of pre- and post-tests as paired sample data. Set the null hypothesis H_0 to “the system has no learning effect” and the significance level α to 0.05, and perform a two-sided test to see if the null hypothesis can be rejected. In addition, the average score of each item is calculated for the results of the questionnaire, and the impression of the system is evaluated. Table 2 shows the test mean, standard deviation, and t -value and p -value after the test, and Table 3 shows the mean of each item in the questionnaire.

TABLE 2. Pre- and post-test results

| | Pre-test | Post-test |
|--------------------|----------|-----------|
| Average | 53.7 | 67.8 |
| Standard deviation | 26.8 | 21.5 |
| t -value | 2.65 | |
| p -value | 0.013 | |

TABLE 3. Questionnaire results

| Content | Average |
|-------------------------------------|---------|
| Sense of accomplishment of the task | 4.2 |
| Hint function | 4.0 |
| Advice content | 3.9 |
| Difficulty of the problem | 3.8 |
| Contents of choice | 3.6 |
| Improvement of weakness | 3.5 |
| Fun of the learning item | 3.5 |
| Number of learning items | 3.4 |
| The continuous use | 3.3 |

Since Table 2 shows that the p -value is less than the significance level of 0.05, the null hypothesis H_0 is rejected and the alternative hypothesis H_1 is adopted. Therefore, it can be considered that this system has a learning effect. From Table 3, it can be seen that the sense of accomplishment of the task was highly evaluated. From this, it is considered that by displaying the answer result as an image or a video, the learner can get a sense of accomplishment of the task more easily than when the result is displayed only by text, and the motivation for programming learning can be improved. It can also be seen that the hint and advice functions were also highly evaluated. From this, it can be considered that the hint and advice function is useful for supporting the learner. On the other hand, the continuous use of the system received a low evaluation. This can be attributed to the fact that the implemented problem content did not attract the learner’s attention and the number of problems was insufficient. Therefore, it is necessary to review the content of the questions and the number of questions, and improve the system so that continuous learning can be promoted.

In addition, a study on the comparison between machine learning and human learning [11] describes a method of learning features for identifying a person with a machine, but it is very different from the process of human learning in this paper. We think that it cannot be applied to the theme of programming learning for beginners.

6. Conclusions. The purpose of this study was to realize a learning support system for beginners of programming learning that aims to improve learners' motivation for learning and promote continuous learning. We devised a programming learning system that outputs the answer results as images or videos, and used the wrong answer data group including the wrong answer images and advice prepared in advance to judge the correctness and present the advice, and evaluate the learning effect. From the evaluation results, it was found that this system has a learning effect. From the evaluation results, it was found that using images and videos makes it easier to see one's own achievements and improves motivation for learning. It was also found that by utilizing the advice function for incorrect answers, the understanding of learning programming was deepened and the learning effect was brought to the user.

However, because there were few patterns of incorrect answer data prepared in advance, advice could only be provided for a limited number of answers. Therefore, as a future task, we will ask more learners to try this system, collect incorrect answer data using the data accumulation function, and present the system to learners with more appropriate advice.

Acknowledgment. This work was supported by JSPS KAKENHI Grant No. 18K02836.

REFERENCES

- [1] Deloitte Access Economics, *Driving Australia's International ICT Competitiveness and Digital Growth*, ACS Australia's Digital Pulse, 2018.
- [2] Ministry of Education, Culture, Sports, Science and Technology Ministry, *Research Report on Programming Education in Foreign Countries*, https://www.mext.go.jp/a_menu/shotou/zyouhou/detail/_icsFiles/afiedfile/2018/08/10/programming_syogaikoku_houkokusyo.pdf, Retrieved on March 3, 2021 (in Japanese).
- [3] J. Fagerlund, P. Häkkinen, M. Vesisenaho and J. Viiri, Computational thinking in programming with Scratch in primary schools: A systematic review, *Computer Applications in Engineering Education*, vol.29, no.1, pp.12-28, 2020.
- [4] J. M. Sáez-López, M. Román-González and E. Vázquez-Cano, Visual programming languages integrated across the curriculum in elementary school: A two year case study using "Scratch" in five schools, *Computers & Education*, vol.97, pp.129-141, 2016.
- [5] M. Mladenović, S. Mladenović and Ž. Žanko, Impact of used programming language for K-12 students' understanding of the loop concept, *International Journal of Technology Enhanced Learning*, vol.12, no.1, pp.79-98, 2020.
- [6] MIT Media Lab Lifelong Kindergarten Group, *Scratch*, <https://scratch.mit.edu/>, Retrieved on March 3, 2021.
- [7] H. Kamada, T. Ishikawa and K. Yoshikawa, A proposal of color image processing applications for education, in *Colorimetry and Image Processing*, C. Travieso-Gonzalez (ed.), IntechOpen, 2017.
- [8] Y. Isaka, D. Katsuragawa, T. Aoki, H. Nitami and H. Kamada, Programming learning system, *Selected Papers from the CIEC Academic Meeting*, vol.10, pp.47-52, 2019 (in Japanese).
- [9] *Processing*, <https://processing.org/>, Retrieved on March 3, 2021.
- [10] D. Shiffman, *The Nature of Code*, Lightning Source Inc., 2012.
- [11] T. Bao, B. Wang, S. Karmoshi, C. Liu and M. Zhu, Learning discriminative features through an individual's entire body and the visual attentional parts for person re-identification, *International Journal of Innovative Computing, Information and Control*, vol.15, no.3, pp.1037-1048, 2019.