Promoting STEAM education using Codey Rocky in Japan

Ono, E., Susono, H., & Koyama, K.

Abstract: This paper describes a project for programming education in Japan. Since 2022, we have conducted more than 40 visiting lessons to teach programming to elementary and middle school students in remote areas in Japan. Based on the previous visiting lesson, we re-designed these lessons. First, we placed the students in groups to work collaboratively. They had a more positive reception to the basic programming concepts in their groups, than learning individually in a teacher-centered environment. We also had teachers participate in our visiting lessons as students. This allowed them to master the basic concepts as well as to experience learning in a student-centered environment. Third, we focused on basic codes. After experimenting with codes in groups, the students became familiar with the basic concepts and naturally integrated them into subject areas. We believe that our approach might be effective in promoting STEAM education in Japan and other countries in Asia.

Introduction

In Japan, programming education was added to the Courses of Study, which functions as the national curriculum, for elementary schools (MEXT, 2018). Incorporating programming into existing subject areas, such as Math and Science, became compulsory in 2020 and onwards. Based on the survey of 1036 elementary school teachers conducted in 2023 by "Minnano Code", one of the leading educational NGOs in Japan supported by multiple organizations such as Google, 69 % of them had the opportunity to participate in workshops to learn how to teach programming in the past year. Only 29% of those who participated in the workshops reported that the workshops were sufficient to implement programming education (Minnano Code, 2023). Furthermore, 42% of them reported that they managed to implement programming education in 2022. Technically, 58% of teachers failed to integrate programming into their teaching, which meant that their lessons did not meet the national standards.

We believe that there are mainly two reasons why the majority of teachers failed to implement programming education. First, the apparent lack of professional development opportunities to learn programming seemed to have contributed to this issue. In addition to the report by Minnano Code, only 28% of teachers implemented programming education. The Ministry of Education, which made programming education compulsory and established the standards, reported that 75% of teachers took professional development sessions for technology integration into classrooms in 2022 (MEXT, 2023), but they did not reveal the number of teachers who had received training for programming education. This may be attributed to the Ministry of Education making programming education compulsory without providing professional development opportunities for the teachers. Additionally, there are 15 sample lesson plans available on the Ministry of Education website (MEXT, 2020). These sample lesson plans: 1) lacked a detailed description of programming education, which made it difficult for teachers to learn and/or use it, and 2) focused on theoretical information (e.g., what is "Gen-AI") rather than hands-on activities.

The second reason might be the strong influence of the traditional teacher-centered approach. In Japan, the maximum number of students in elementary classrooms is currently 40 and will be reduced to 35 by 2025. To teach such a big class, Japanese teachers need to conduct their lessons with a teacher-centered approach. They also need to be knowledgeable in all subjects since homeroom teachers usually teach all subjects, including programming. There are no educational technology specialists in public schools in Japan. There are no assistants when implementing programming education. The homeroom teachers are autonomous and therefore may have challenges in effectively teaching programming to their students. In order to develop and maintain their authority in the programming subject area, they need sufficient time and training to master programming. Unfortunately, these two crucial resources are currently unavailable to them.

To solve these problems and promote programming education, the group of authors applied for the research grant given by the Ministry of Education in Japan, called "Kaken." Funded by the Kaken, we gave free visiting lessons at over 40 elementary and middle schools in the rural areas in Japan from 2017 to 2021. During this period, most schools did not have computers with the required specs for programming. So, we needed to bring 35 laptops for the visiting lessons. Due to the limited access to the Internet at schools in the target area, we downloaded the Scratch office editor to our computers, which was a free programming language and made available in more than 70 different languages, including Japanese. We taught the students "hands-on" programming with a step-by-step approach. After the quick instruction, we encouraged individual students to practice with the aim of having them experience: 1) breaking down complex problems into smaller pieces to foster problem-solving skills, and 2) analyzing data and identifying patterns and trends to promote analytical thinking. At the end of the visiting lessons, we had individual students create their own projects.

We gave more than 100 visiting lessons in 40 schools in four years. The students had sufficient time to master the basic programming concepts; however, not many students became capable of programming with Scratch. We learned that exploring the different blocks and commands did not engage Japanese students in programming. We analyzed our visiting lessons and came up with some possible problems. Firstly, we should have created opportunities for the students to work together, solve problems collectively to facilitate reciprocal learning, instead of experimenting individually when mastering the basic concepts of programming. Secondly, we should have had the teachers participate in the visiting lessons instead of merely observing. By participating in the visiting lessons, we could have given professional development opportunities for teachers to familiarize them with programming thinking concepts and teaching methodologies. Thirdly, we should have focused on the basic

programming concepts before introducing cross-curriculum integration. It might be too challenging for the students to integrate the basic concepts, which they have just been introduced to, into subject matters.

Visiting Lessons

In 2022, we again applied for the Kaken grant and started to teach programming with Codey Rocky, which was a programmable educational robot designed for children, to familiarize the students with the logic of computer programming. The Codey Rocky robot assumes a feline-like form capable of making forward/backward and circular movements as programmed. It supports block-based programming, such as mBlock by Makeblock's coding platform for beginners based on Scratch 3.0. We conducted more than 40 visiting lessons to teach programming with Codey Rocky between 2022 and 2024.

To avoid any repeated mistakes, we carefully redesigned our visiting lessons. First, we had the students work in pairs or groups. Working collaboratively might allow them to devise multiple solutions to a problem and evaluate their effectiveness to foster creativity and innovation. Since there was no single answer/solution to the problem, the teachers were not expected to know their answers and were less concerned about their level of authority on that subject matter. Second, we encouraged the homeroom teachers, who usually took care of all subjects, including programming, to participate in the visiting lessons as students. In this way, we could provide professional development opportunities with teachers to master the programming thinking concepts and teaching methodologies. Third, we spent sufficient time for the students and teachers to become familiar with the basic programming concepts. Codey Rocky is designed for younger children; therefore, it requires simple code. Thus, the students and teachers could focus on the basic codes instead of numerous codes in Scratch.

We started the visiting lessons in 2022. The students and teachers worked in pairs or groups to practice with Codey Rocky after the short introduction. Even though it was a quick introduction, both the students and teachers could figure out how to program Codey Rocky by themselves. The simple codes with Codey Rocky resulted in engaging the Japanese students and teachers. Surprisingly, they gradually incorporated the idea of programming into the existing subject areas. For example, some students were eager to move the robot in an equilateral triangle shape with simple codes they had just mastered. Other students programmed the robot to play a simple song they had learned in music classes. It was surprising to witness that once they mastered the basic programming concepts, they naturally integrated the concepts into subject areas.

After our visiting lessons, some teachers seemed to become confident in teaching programming. It seemed as if they had mastered the basic programming concepts and methodologies to teach programming lessons. Some borrowed the robots from us and taught programming by themselves.

Figure 1

Teaching Programming in Relation to Mathematics by the First Author in the Visiting Lesson

Implications

We taught programming lessons in the rural areas in the central part of Japan. Although our experiences were limited and would not be applicable to other areas in Japan and other countries, we would like to draw some implications. We hope that our implications will be helpful for other schools that are struggling with promoting programming education.

First, the basic programming concepts might be better mastered when students work together. The teachers should foster opportunities for student collaboration and reciprocal learning. Most students might not get used to working collaboratively

since they are used to the teacher-centered environments throughout their school journey. Programming would be a good start in getting familiar with learning in groups. They need to be prepared for the future workforce, where both collaboration and basic programming concepts are essential. The students need a "paradigm shift" from teacher-centered to student-centered learning environments.

Second, Japanese teachers need professional development opportunities to get accustomed to programming thinking concepts and how to integrate them into other subject areas. In addition, Japanese teachers need to learn what a "paradigm shift" is and why it is essential for programming education. Based on the report by Minnnano Code (2023) and our visiting lessons between 2019 and 2022, most learners, who worked individually, failed to master the basic programming concepts. Programming skills flourish in a collaborative learning environment.

Third, integrating programming into subject areas should be a gradual process. Once the students master the basic concepts, they naturally integrate them into the subject areas, as observed in our visiting lessons. Most students and teachers need sufficient time to master the basic programming concepts. We suggest that you should not introduce more complex concepts until they have mastered the basic concepts.

Conclusions

It has been four years since programming education became compulsory in Japan. Based on our experiences of visiting lessons between 2017 and 2021, we needed to redesign the programming of visiting lessons for Japanese schools. First, we decided to teach programming with a student-centered approach. We put students in groups and have them collaboratively work together to master basic programming concepts. Second, we had homeroom teachers in elementary schools who taught all subjects, and they participated in our visiting lessons as students. We made sure that they had sufficient time to master the basic concepts and learning in a student-centered environment. Third, we focused on the basic concepts. Our students had sufficient time to fully master the basic concepts; then, they naturally integrated the programming concepts into subject areas.

Implementing programming education in constructivist environments would be a breakthrough to promote STEAM education, which has been introduced but not successfully implemented yet in Japan. In Japan and other Asian countries, most subject areas are taught with the teacher-centered approach, probably for two reasons. First, the teachers need to remain the authority figure due to the class sizes to effectively manage their classrooms. Second, students who are taught with the teacher-centered approach tend to perform better in the entrance exams, which assess only rote knowledge instead of actual problem-solving skills. It is very competitive for students in Asia to go to universities. They are able to learn in teacher-centered environments to effectively memorize the information. Unfortunately, the memorized information is not sufficient for the students to survive the challenge of globalization. Thus, we need to enhance students' creativity and innovation. Programming education encourages students to think creatively and foster innovation. It allows students to devise multiple solutions to a problem and evaluate their effectiveness.

Moreover, we need to foster students' digital literacy. Programming education serves as a foundation for understanding how technology works and equips students with the skills necessary to navigate and thrive in a technology-driven world. Furthermore, programming education fosters future workforce readiness. Many industries require professionals who can utilize technology and data effectively. We can prepare our students for the future workforce by integrating programming into the curriculum and enhancing their career prospects.

One of our next goals is to explore affordable tools other than Codey Rocky, which costs about 100 USD per robot. Tools like Codey Rocky make it easy for Japanese teachers to teach programming with a student-centered approach. They are also effective for students to master the basic concepts with a more playful and relaxed approach. In addition, we plan to provide opportunities for students to attempt real-world applications. We believe that connecting programming education to real-

world applications is more meaningful for students. We would like to show them how programming is used in various fields, such as healthcare, transportation, and entertainment. Guest speakers (e.g., drone pilots) and field trips to factories (e.g., Toyota) can help students see the practical applications.

References

References

Ministry of Education, Culture, Sports, Science and Technology (2023). Reiwa 4nendo gakkouniokeru kyoikuno johoka nikansuru chosakekka [A report on technology integration into schools in 2020]. https://www.mext.go.jp/content/20231031-mxt_jogai01-000030617_1.pdf

Ministry of Education, Culture, Sports, Science and Technology (2018). Shogakko programming kyoiku hisshuka ni mukete [A guide for compulsory programming education in elementary schools]. https://www.mext.go.jp/b_menu/shingi/chukyo/chukyo3/004/siryo/__icsFiles/afieldfile/2018/10/05/1409851_6.pdf

Ministry of Education, Culture, Sports, Science and Technology (2018). Shogakko programming kyoiku shidoan shu[Sample lesson plans for programming education in elementary schools]. https://www.mext.go.jp/content/20200408-mxt_jogai01-000006441_001.pdf

Minnano Code (2023). Programming kyoiku jittaichosa hokokkusho 2022 [A report on programming education in elementary schools]. https://code.or.jp/news/20230809/