Olympiads in Informatics, 2024, Vol. 18, 89–100

DOI: 10.15388/ioi.2024.07

From Concept to Code: A Two-Day Workshop for Secondary Students on Computational Thinking and Programming

Felix STEINERT, Julia KUMMER,
Martina LANDMAN, Lukas LEHNER
TU Wien, Institute of Information Systems Engineering
Favoritenstraße 9-11, 1040, Vienna, Austria
e-mail: felix.steinert@tuwien.ac.at, julia.kummer@tuwien.ac.at,
martina.landman@tuwien.ac.at, lukas.lehner@tuwien.ac.at

Abstract. Introducing programming and informatics concepts to the next generation of computer scientists is essential. This experience report presents a detailed overview of a two-day informatics workshop for 134 Austrian school kids, ages eleven to thirteen. The workshop program consists of several unplugged activities about algorithms, AI, robotics and coding and block-based programming using Scratch and Sphero BOLT. We evaluated feedback from 110 participants regarding their experience of these two days. Many children reported a significant gain in knowledge. We will also explain the pupils' favourite activities and educational concepts in detail. This report also covers the experience of the ten workshop leaders of the past two days. The workshop was widely positive received and the participants reported a high interest in computer science.

Keywords: computer science education, school outreach, AI, Scratch, algorithms, workshops.

1. Introduction

Incorporating informatics education into secondary schools is essential for preparing students for success in today's technology-driven world.

There is a persistent demand for IT professionals, and it is crucial to broaden participation, particularly among underrepresented groups like girls. Efforts to engage students creatively are vital for fostering interest and participation in computer science, as reported by many institutions, for example Giannakos *et al.* (2013), describing a project with a two-day workshop using scratch and recycled materials or Rottenhofer *et al.* (2022), reporting on a computer science (CS) Workshop in the setting of Circus performance. As Sabitzer *et al.*, (2014) highlight, initiatives like CS Unplugged demonstrate success in engaging students' creativity and fostering interest in technology. They also underscore the value of computational thinking (CT) and early exposure to informatics concepts.

CT is a set of problem-solving skills everyone needs, not only computer scientists, but everyone in their everyday lives (Wing, 2006).

As part of the TU Wien Informatics Didactics group, the eduLAB team provides workshops (Prinzinger, 2021) aimed at enhancing CT and programming skills (Landman *et al.*, 2022; Unkovic & Landman, 2023) among students, starting from the 2nd grade onwards (Landman *et al.*, 2023). Our workshops are designed to prioritize hands-on, age-appropriate activities that empower students to develop and implement their problem-solving strategies, inspired by CS-unplugged activities (Bell *et al.*, 2015). Through engaging tasks, students explore fundamental concepts and methods of computer science, fostering a deeper understanding of computational thinking (CT). CT, understood as a set of problem-solving skills in CS, is crucial for learning informatics, especially including programming education (Grover & Pea, 2013).

Passing on these core skills to the future generation is one of our goals. Through suitable learning settings, we try to increase young people's interest in programming and the broad field of computer science. As part of a secondary school's "informatics days" in February 2024, the TU Wien Informatics eduLAB conducted a two-day workshop program. The workshops were held by CS university students in the bachelor's, master's, or PhD program. In this article we report on the feedback from the eleven-to thirteen-year-old pupils on this two-day workshop intervention.

2. Program Overview

The program spanned two days, with 134 secondary school pupils from the 6th grade divided into eight groups, each comprising 16 to 17 participants. Following an introductory session during which pupils completed an initial questionnaire, they proceeded to follow a predefined workshop schedule. Each pupil engaged in every session offered as part of the program. Further details regarding the workshop program are provided below.

2.1. *Day 1: Concepts of Computer Science*

The first day of the workshop was dedicated to unplugged concepts of computer science through a series of four 55-minute interactive workshop sessions. In each session the pupils worked in groups of 4–5 persons, engaging with materials that are designed to be exploratory, playful, and hands-on. With the "unplugged" concept we try to enable interactive learning processes for informatics exercises that do not require computers.

Codes

In this session the topic "Codes" was introduced utilizing everyday life analogies, like crosswalks and traffic lights to effectively convey the concept of symbols having associated meaning.

Pantomime: As introductory task, pupils played a charades-like game, allowing them to uncover characteristics of good codes, like uniqueness and universal understanding. During this task, the pupils guessing the words were not allowed to speak either but could use gestures to communicate their questions before writing down their guesses.

Variable-Length Codes: Following this, the pupils were introduced to the concept of variable-length codes through the exercise “juice bar”, where they were asked to find erroneous codes and determine the binary representation for various fruits based on given combinations.

Uniform-Length Codes: As bonus task for fast pupils, this task let them explore the binary coding system further by playing an adapted version of the memory game “I packed my bag”: instead of verbally adding items to a figurative bag or suitcase, they physically pin different combinations of 3D printed elements onto a board to encode and “store” their items.

Algorithms

Similar to the Codes session, the concept of algorithms was introduced using everyday analogies, such as the step-by-step process of preparing a frozen pizza or making breakfast. Conditionals were explained using the analogy of being vegetarian (“if you are vegetarian use cheese, else use ham”).

Sorting Cards: With the use of the cards of a game called “Ligretto”, the pupils could playfully develop their own sorting strategies for a shuffled stack of cards in a small group (Fig. 1). As described by Landman *et al.* (2023), this task not only represents a hands-on implementation of sorting algorithms but also provides a practical understanding of key concepts in computer science, such as “Divide and Conquer”, distributed computing and resource optimization.

Scheduling: During this task, pupils learned about scheduling algorithms. While planning activities for an entertainment park and a zoo, they discovered the necessity for algorithmic solutions to optimize scheduling processes. After having tried to find and describe their own scheduling algorithm, they were introduced to a “line-sweep” algorithm, which they applied hands-on to solve the scheduling challenges.

Fig. 1. Sorting activity using Ligretto cards.

Fig. 2. An Ozobot robot following a black line.

Artificial Intelligence

Decision Tree: The artificial intelligence (AI) session focused on decision trees with pupils asked to build a decision tree from 3D printed materials that could classify several types of fruit. Pupils were supplied with a data set represented by playing cards. Once they built a model that could correctly classify the given data, the pupils received additional test data that included some intentional stumbling blocks. The outcomes of their models' predictions were discussed in the end, including problems like missing classes or attributes and potential strategies for improving their models were elaborated.

Computational Thinking supported with Ozobots

Ozobot-Maze: The Ozobot-Maze session provided pupils with a hands-on experience with algorithms. The robot follows black lines (Fig. 2) and performs actions based on colour codes. The pupils first needed to decipher the meaning of these codes for the Ozobot. In subsequent steps, the pupils were asked to apply these commands using stickers to navigate the robot through different maps.

2.2. Day 2: Programming Basics

On the second day, the focus shifted to hands-on programming experiences as pupils were introduced to programming using Scratch and Sphero BOLT.

Programming basics with Scratch

Scratch*:* During this session, the pupils learned programming basics using Scratch, a beginner-friendly, block-based visual programming language. In the first part, the pupils were introduced to the development environment and worked together to complete three exercises. In the second part, they brainstormed and programmed their own unique games, with workshop leaders providing assistance as needed.

Programming basics with Sphero

Sphero: The Sphero BOLT is an educational robot sphere that can be programmed using a block- or text-based programming language. Therefore, it is suitable for beginners in programming. In this session the pupils could work through tasks at their own pace, beginning with very easy tasks (only using commands in a linear sequence) up to intermediate tasks (using nested loops). The robots provide motivation and offer space for creativity, e.g. by letting pupils program colourful pictures onto the integrated LED matrix.

3. Feedback Evaluation

3.1. *Method*

The pupils were provided one initial questionnaire in the beginning of the two-day workshop program and a second one concluding the workshops. Both questionnaires were conducted anonymously using Microsoft Forms.

The initial questionnaire focused on the pupils' interest in informatics, which they were asked to rate on a scale from 1 to 5.

The final questionnaire, with a total of 12 questions, was divided into three sections: Anonymous personal information, general workshop feedback and questions about informatics.

In the first section participants were asked to provide demographic details such as gender and age.

The section on workshop information allowed participants to offer general feedback on the two-day workshop and indicate whether they would recommend the workshop to other pupils. Additionally, they could select their favourite workshop session of each day and rate the difficulty of all workshop tasks on a Likert scale ranging from "very difficult" to "very easy".

The final section focused on the broader topic of informatics. Here, the pupils could again rate their interest in informatics and provide insights into the extent of new knowledge they gained. Using a Likert scale, participants were asked to express whether they could envision themselves working in a computer science-related field in the future. Various job titles like "Student in Informatics", "Programmer", "Computer Science teacher" and "IT technician" were provided for rating. Lastly, a free text form was included to provide the possibility for individual feedback.

3.2. *Participants*

The workshop participants were eighth-grade pupils from a secondary school, aged between 11 and 13 years old. All attendees were from the same school. A total number of

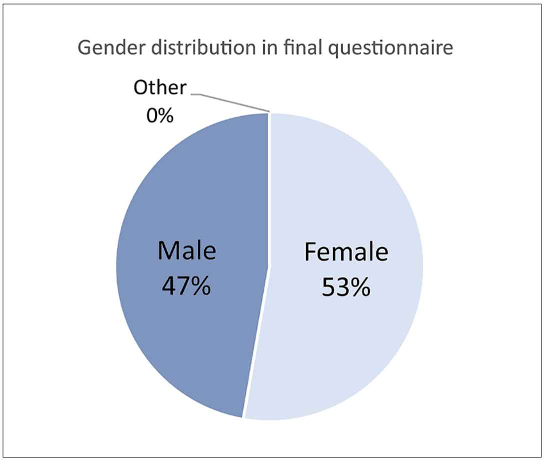


Fig. 3. Distribution of pupils' gender in final questionnaire.

134 pupils participated in the workshop, 125 of them answered the initial questionnaire and 116 the final feedback questionnaire.

From the final questionnaire's responses, six were excluded from the analysis due to suspicion of non-serious answers. Among the remaining 110 responses, the gender distribution was balanced, with 53% identifying as female and 47% as male (Fig. 3. Distribution of pupils' gender in final questionnaire).

4. Evaluation Results

Feedback

Overall, the workshop was well received with 43% of pupils giving the workshop 5 out of 5 stars and the average being 4,07.

By far the most popular task on day one was Ozobot-Maze with 53 pupils voting it as their favourite. In second place was Sorting Cards with 25 votes (Fig. 4). On day two Scratch and Sphero BOLT received roughly the same numbers of votes.

When asked about the difficulty of the tasks over 50% of participants reported "easy" or "very easy" on every exercise (Fig. 5).

Acquired Knowledge

33% of the 110 participants that finished the second questionnaire reported that they learned a lot about informatics.

Attitude towards computer science

We asked the pupils before and after the workshop about their interest in CS on a scale from 1 to 5 stars. Before the workshop 50% of pupils rated their interest a 4 or 5 after the workshop this increased to 63%. On the other side the percent of pupils who rated

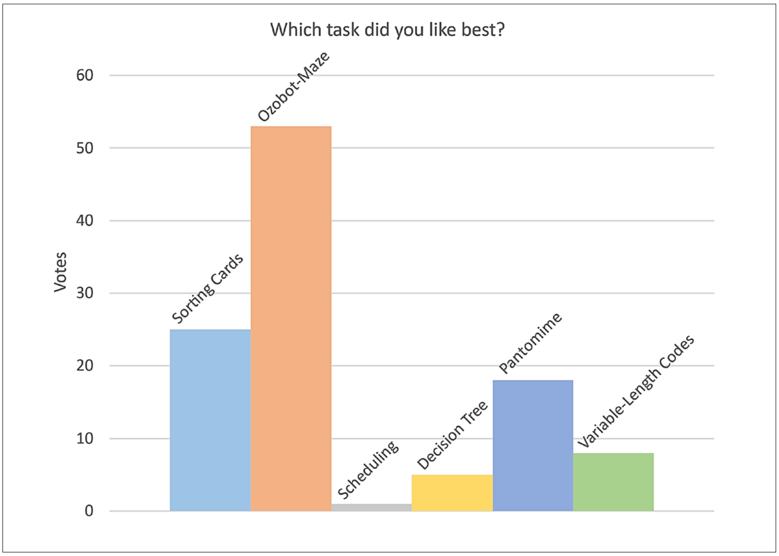


Fig. 4. Favourite workshop tasks as voted by the pupils.

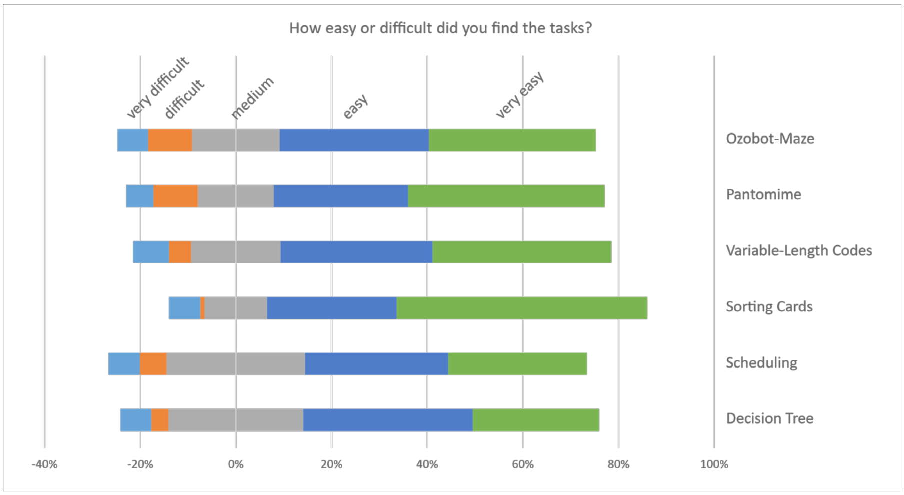


Fig. 5. Difficulty of the workshop tasks as voted by the pupils.

their interest in CS as a 1 or 2 decrees from 32% to 17% (Fig. 6). This shows we were able to increase the interest in CS in the pupils through our workshop directly after the two days.

When pupils where asked if they could imagine to work various IT jobs in the future only 15% answered with “fairly imaginable” and “very easy to imagine” (Fig. 7). A problem with this question could be, that they did not know what those jobs do and require.

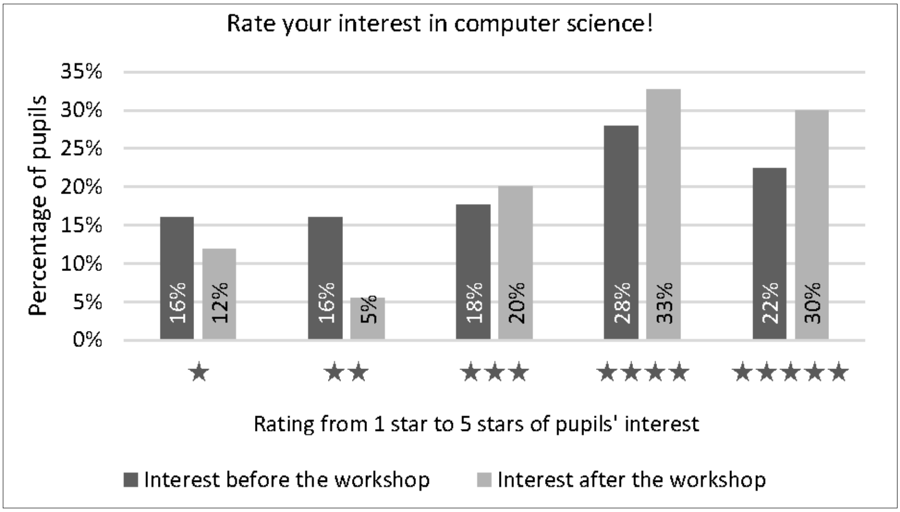


Fig. 6. Pupils' interest in computer science before and after the workshop.

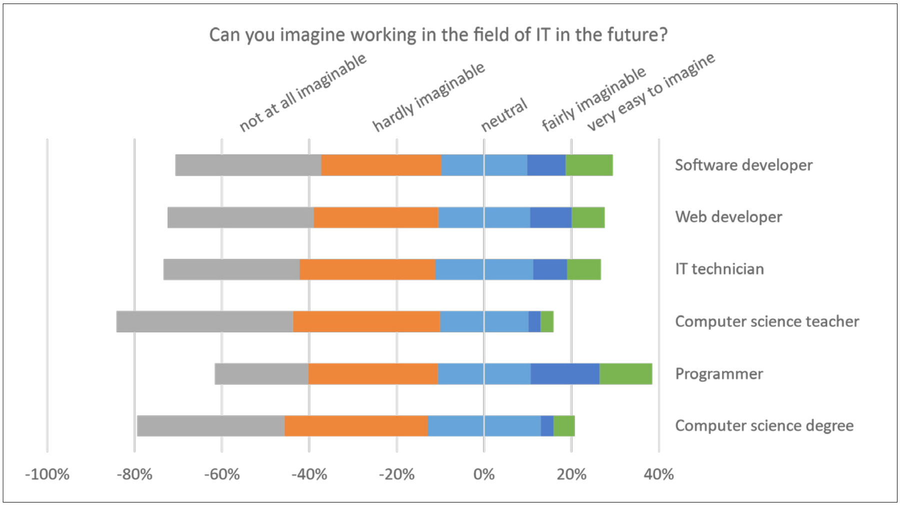


Fig. 7. Pupils' attitude towards their possible futures in IT.

5. Workshop Reflections

Concluding the workshop, we gathered informal feedback from the workshop leaders. The feedback was collected through oral discussions, where observations, insights and opinions on the interaction with the pupils were described. This feedback was then compared with the results from the questionnaires completed by the participants.

Workshop leaders could observe that the children were more excited about the practical tasks on the second day, where they needed to use their own laptop, compared to the more theoretical tasks on the first day. However, as some workshop leaders noticed, a few pupils faced challenges to stay focused, especially given the many distractions on their laptops. Additionally, it was observed that the later it was the more the children struggled with concentration.

Another observation was that depending on the order in which the sessions were completed, pupils could reproduce concepts they learned in the previous session, potentially relevant for the current one. For example, those who had already completed the algorithm task, involving sorting cards, were able to reproduce algorithm principles in the Ozobot-maze session.

Furthermore, a few workshop leaders noticed that the experience and interest with Scratch varied among the pupils, which made it difficult to engage all pupils in the task. One workshop leader reported that some pupils immediately started to create their own games instead of following the course.

Based on the findings of the questionnaires shown in Fig. 4 many pupils reported that they perceived the AI task easy or very easy. However, a workshop leader shared that when asking the pupils to employ the decision tree model many of them used their human intelligence to solve the problem. This shows up a potential gap in understanding the AI concept.

In comparison to similar initiatives, one aspect that sets us apart is that our workshop offered a broad range of topics, spanning from concepts of CS, like algorithms, coding and AI, to hands-on programming activities. Our workshop provides pupils with a combination of practical programming exercises and “hands-on” learning of theoretical basics. This comprehensive approach within the two-day setup ensures that pupils have the opportunity to explore different areas within the field of CS and discover their strengths and interests.

Moreover, our extensive experience in conducting workshops, based on our regular program at the eduLAB, where over 2500 school children of all ages participate in annually , sets us apart. Based on this experience, our workshop is designed to be accessible to pupils of all skill levels. Whether they are complete beginners or have some prior experience with computer science, our workshop offers multiple difficulty levels in most tasks, which allowed them to proceed at their own pace or in a group setting.

Additionally, our workshop incorporates quantitative data analysis from questionnaires filled out by pupils. This approach enables us to gather valuable insights into the effectiveness of the workshop and adapt future exercises to better meet the needs of participants.

6. Conclusion

In this experience report we showed how we introduced computational thinking through unplugged activities and programming to pupils over a two-day workshop. On the first day we covered informatic concepts like algorithms, codes, and AI with unplugged exercises. The second day was focused on pupils programming themselves with Scratch and Sphero BOLT. We also collected feedback before and after the workshop from the pupils, using two questionnaires. The results showed that there is a high interest in CS but that despite of that not many students can imagine working in the field of IT in the future. Also, the students reported that they learned a lot of new information about CS. The questionnaire responses indicated that many pupils found most tasks to be easy or very easy. This suggests an opportunity to enhance the exercises to make them more challenging. Overall, the workshop was a success and showed that there is a lot of interest in CS.

As limitations to the results, we must consider that we could only collect data from a small age group (eleven- to thirteen-year-old) in one school. We aim to repeat this workshop to get better and more accurate feedback, as well as to increase the sample size. Additionally, the questionnaire we used was not validated. It is based on our curiosity in the pupils' perception of our work. In this experience report, we focused on the feedback of the students. Repeating the workshops but including pre-and post-tests with an existing validated questionnaire instead of our questionnaire, can be considered for future work. Nevertheless, this experience report provides meaningful insights into how to engage pupils into CS and CT. Looking further ahead, it would be desirable to deepen the insights gained and incorporate them into the various CS curricula.

In conclusion, supporting young people through workshop interventions, teaching key computer science concepts, is a big step forward to increase the number of young talents competing in programming challenges and competitions. Increasing the interest in CS is especially important to get young minds participating and contributing to the CS community with their skills and ideas.

References

Bell, T., Witten, I., Fellows, M. (2015). *CS Unplugged: An Enrichment and Extension Programme for Primary-aged Students*. https://www.csunplugged.org/

Giannakos, M., Jaccheri, L., Proto, R. (2013). Teaching Computer Science to Young Children through Creativity: Lessons Learned from the Case of Norway. In.

Grover, S. & Pea, R. (2013). Computational Thinking in K–12. *Educational Researcher*, 42(1), 38–43. https://doi.org/10.3102/0013189X12463051

Landman, M., Futschek, G., Unkovic, S. & Voboril, F. (2022). Initial Learning of Textual Programming at School: Evolution of Outreach Activities. *Olympiads in Informatics*, 43–53. https://doi.org/10.15388/ioi.2022.05

Landman, M., Rain, S., Kovács, L. & Futschek, G. (2023). Reshaping Unplugged Computer Science Workshops for Primary School Education. In: J.-P. Pellet & G. Parriaux (Ed.), *Lecture Notes in Computer Science. Informatics in Schools. Situation, Evolution, and Perspectives: 16th* (Vol. 14296, p. 139–151). Springer International PU. https://doi.org/10.1007/978-3-031-44900-0_11

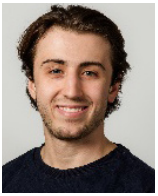
Prinzinger, P. (2021). Informatik für alle! Aktivitäten zu Computational Thinking, Programmieren und „Zaubertricks". *OCG Journal*, 46(1–2), 13–15. https://repositum.tuwien.at/handle/20.500.12708/137759 (Erstveröffentlichung 2021)

Rottenhofer, M., Kuka, L. & Sabitzer, B. (2022). Clear the Ring for Computer Science: A Creative Introduction for Primary Schools. In (p. 103–112). Springer, Cham. https://doi.org/10.1007/978-3-031-15851-3_9

Sabitzer, B., Antonitsch, P.K. & Pasterk, S. (2014). Informatics concepts for primary education. In: C. Schulte (Ed.), *ACM Digital Library, Proceedings of the 9th Workshop in Primary and Secondary Computing Education* (p. 108–111). ACM. https://doi.org/10.1145/2670757.2670778

Unkovic, S. & Landman, M. (2023). Supporting Non-CS Teachers with Programming Lessons. In: J.-P. Pellet & G. Parriaux (Ed.), *16th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2023, Local Proceedings* (p. 61–74). Zenodo.

Wing, J.M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35. https://doi.org/10.1145/1118178.1118215



F. Steinert studies Software & Information Engineering at TU Wien in the bachelor program. Since the winter semester of 2023 he works at eduLAB as a student assistant conducting workshops for school classes and helping with research.

J. Kummer is a student of the master program Media- and Human Centered Computing at TU Wien. Alongside her studies she works as student assistant at the TU Wien Informatics eduLAB, conducting workshops for school classes and helping with research.

M. Landman Researcher at TU Wien and member of the Informatics eduLAB group in the research unit of Information & Software Engineering since 2021. She has experience in teaching computer science from 5th to 12th grade. She organizes the computer science faculty's school outreach activities, where she develops, organizes, and conducts weekly workshops for school classes. Her research focuses on algorithmic problem solving for kids.

L. Lehner is a researcher and PhD candidate in informatics didactics at TU Wien. His research focuses on how to best promote young people's AI literacy. He develops unplugged learning materials that let learners discover the technical functioning of various machine learning methods without the use of computers.