# TEACHING ENVIRONMENTAL STUDIES WITH KUBO ROBOTICS

**Sara NIKL**
Primary School Sladki Vrh, Slovenija
sara.nikl97@gmail.com

UDK: 502:]37.091.3:004]=111

**Polona JANČIČ HEGEDIŠ**
Faculty of Education, University of Maribor, Slovenija
polona.jancic1@um.si

**Vlasta HUS**
Faculty of Education, University of Maribor; Slovenija
vlasta.hus@um.si

**Abstract**

Computational thinking is defined as a key competence of modern times, as it combines diverse cognitive abilities and skills that help individuals become a successful part of modern society. It is therefore important to start developing it among students from primary school onwards. In this article, we present the activities for developing computational thinking through the content of environmental studies, encompassing objectives from the thematic heading *Space* in third grade. Through tasks, which were supported with KUBO robotics, students learned how to solve problems, process information systematically, think logically and algorithmically, use abstraction, generalise, cooperate, and above all, it gave them the opportunity to be mentally active through all the learning process. In the empirical portion of the article, we show that despite the limited number of lessons, students have advanced and better understood the basic programming concepts after the activities took place. The activities proved to be interesting as students were drawn to KUBO, which proved to be a useful and applicable tool to teach beginner programming.

**KEYWORDS**: *computational thinking, environmental studies, KUBO robotics teaching*

## Introduction

The following article is written based on the concept of implementing computational thinking through the early introduction of programming education. It seeks to emphasise the importance of developing this cognitive skill from an early age and highlights the potential strategy for achieving this goal through the integration of KUBO robotics into educational studies curricula.

## Computational thinking

Many authors (Krajnc et al., 2017; Haseski, Ilic & Tugkin, 2018; Heljakka et al., 2019; Bers, 2019) consider computational thinking as a key competence of 21st century, as it encapsulates an essential set of skills required for individuals to succeed in modern society. The world-renowned non-profit organisation, the International Society for Technology in Education (ISTE), lists computational thinking as one of its seven learning standards.

The concept of computational thinking originated in the past century. On one hand, it empowers learners to understand the basics of technology and its usability; on the other, the professionals use it to solve complex scientific dilemmas and analyse large quantities of data (Wing, 2011). Researchers agree on the need to develop this concept further but seem unable to outline its borders since computational thinking offers a great variety of viewing angles. Wing (2006) defines it as the fundamental ability to solve problems, design systems, analyse problems, and think abstractly or apply heuristic thinking. Wolfram (2016) says that the process—which must be clear, systematic, and effective—of telling the machine what needs to be done, is crucial. Aho (2012) was inspired by Wing's definition (2011), although he emphasised that end results should be presented in steps and algorithms. Computational thinking could also be perceived as analytical thinking with many parallels to mathematical, engineering, and scientific thinking (Bers, González-González & Armas–Torres, 2019). We should not equate it solely with programming (Balanskat & Engelhardt, 2015) or algorithmic thinking (Sterling, 2015) since many different thought processes participate in this complex mental activity. Therefore, it represents a complex mental activity in which different thought processes participate, thus making it erroneous to equate it with programming (Balanskat & Engelhardt, 2015) or algorithmic thinking (Sterling, 2015). These definitions help us identify characteristics of computational thinking related to the ability to abstract and generalise patterns, systematic information processing, algorithmic thinking, structured decomposition of the problem, iterative, recursive, and parallel thinking, logical thinking, efficiency and effectiveness, and finally, systematic detection and correction of errors (Grover & Pea, 2013). In addition to thought processes, Csizmadia et al. (2015) emphasise computational thinking techniques relating to evaluation, programming, creation, analysis and practical use. Computational thinking can be divided into concepts, practice, and perspective (Brennan & Resnick, 2012).

## Activities

In the third grade, we started developing computational thinking using KUBO robotics, which combines the two most common forms of early programming teaching: physical computing and computer-free activities. KUBO is a robot programmed by students using Tag-Tiles and a map prepared by the teacher according to the curriculum, allowing students to understand complex and abstract processes on a concrete level. The activities were designed for students working in a couple or small groups, meaning that they were trained in communication and collaborative skills. As part of the teaching, we carried out seven learning sessions lasting from 45 to 90 minutes. The students received a KUBO kit (robot and Tag-Tiles) and a special map, designed for the lesson, onto which they placed Tag-Tiles, using them to navigate paths to the destination, which was the main objective. The lessons were planned in a cross-curricular fashion, and the objectives and concepts of computational thinking were defined in detail. The table presents the objectives, content (Slovenian National Curriculum) and images of computational thinking throughout.

**TABLE 1** Activities

| Lesson plan | National Curriculum Goals | National Curriculum Content | Programming Concepts |
|---|---|---|---|
| KUBO introduces himself | Students learn cardinal directions (north, south, east, west). | Cardinal directions | Sequences |
| Kubo's *Hidden Treasure Hunt* | Students know how to use different types of sketches and maps. | | Sequences |
| Kugo goes around the world. | Students expand their knowledge about different landscapes. | World, oceans, continents | Sequences / introducing functions |
| KUBO in Slovenia | | Native landscape, Slovenia, Europe | Functions |
| KUBO in Sladki Vrh | Students get to know the types of settlements | Settlements (city/village) | Functions / introducing loops |
| KUBO on a farm | Students learn about living and working on a farm during different seasons. | | Loops |
| KUBO at the market | Students get to know the market and its offers. | | Loops |

In the introductory part of the lessons, activities were delivered to motivate students and content was presented. In the central part, students actively explored the range with the help of KUBO, and in doing so, they cooperated, helped one another, and explored the robots and their functions. At the end, we summarised the content together through a conversation; a review of what had been done or that of a game. Each lesson was analysed as part of the self-evaluation and opinion of the teacher, with an emphasis on the appropriateness and effectiveness of teaching content with KUBO robotics. According to the three-dimensional model (Brennan & Resnik, 2012)—which divides computer-type of thinking into concepts, practice, and perspective—we developed two out of seven images, four out of five courses, and all perspectives. Furthermore, all the mental operations necessary for successful computational thinking were addressed. Students responded positively to the activities, as these encouraged both independent and collaborative work. The only limitation identified was time; students required additional opportunities for repetition to consolidate their newly acquired knowledge.

## Research

The empirical part aimed to determine the potential differences between the student's prior knowledge and the knowledge acquired in understanding basic programming concepts. Those differences were related to reading a sequence of commands, finding the error in a series of commands, determining the missing step in the sequence, understanding a loop with one order, understanding a circle with several authorities, and finding the shortest possible route. The assumption was that children would be more successful in understanding basic programming concepts after performing the activities. The research was conducted on a sample of twenty-two third-grade students. Data was obtained via a computer-based reasoning test, which the students solved twice—in the first lesson, before we caried out planned activities, and in last lesson, after the activities were successfully completed. The test consisted of six multiple-choice closed-ended type of tasks. Further, the collected data were statistically processed using a paired-samples *t*-test and Cohen's *d* coefficient. The results obtained through SPSS are presented in Table 2 below.

**Table 2** Results of SPSS

| Hypothesis | Mean | | Standard deviation | | T-test | P | Cohen coefficient |
|---|---|---|---|---|---|---|---|
| | Test 1 | Test 2 | Test 1 | Test 2 | | | |
| H1 | 0.68 | 0.82 | 0.477 | 0.395 | 1.821 | 0.083 | 0.388 |
| H2 | 0.55 | 0.73 | 0.510 | 0.456 | 2.160 | 0.042 | 0.461 |
| H3 | 0.05 | 0.55 | 0.213 | 0.510 | 4.583 | 0.001 | 0.977 |
| H4 | 0.45 | 0.32 | 0.510 | 0.477 | 1.821 | 0.83 | 0.388 |
| H5 | 0.32 | 0.77 | 0.477 | 0.429 | 4.183 | 0.001 | 0.892 |
| H6 | 0.32 | 0.68 | 0.477 | 0.477 | 3.464 | 0.002 | 0.725 |

**Legend:**
H1: It is assumed that students will read a sequence of commands more accurately after carrying out the planned activities.
H2: It is assumed that students will complete the sequence of commands more effectively after carrying out the planned activities.
H3: It is assumed that students will identify the error in a series of commands more successfully after carrying out the planned activities.
H4: It is assumed that students will better understand the significance of a loop with one command after carrying out the planned activities.
H5: It is assumed that students will better understand the significance of a loop with several commands after carrying out the planned activities.
H6: It is assumed that students will find the shortest possible route effectively after carrying out the planned activities.

Test 1 indicates the pre-test, carried out before planned activities, while Test 2 represents the post-test, respectively. The students solved the task of reading a sequence of commands better on Test 2 than on Test 1, but the difference was not statistically significant ($t(21) = 1.821$, $p = .083$). In the pre-test, the students demonstrated a satisfactory level of understanding ($\bar{x} = 0.68$), and their performance improved slightly ($\bar{x} = 0.82$). Thus, the progress in reading the series of commands was of medium strength (Cohen's $d = 0.388$). Therefore, H1 can be neither confirmed nor rejected with certainty.

A statistically significant difference was observed in the task concerning the sequences of orders completed, with a statistically significant difference between Test 1 and Test 2 ($p = .042$), meaning that because of the performed activities, students learned to better identify the missing command in the sequence of commands. The corresponding effect size was of medium strength (Cohen's $d = 0.461$). Hence, the H2 is confirmed.

There was also a statistically significant difference in their success in identifying an error within the series of commands ($t(21) = 4.583$, $p = .001$), which indicates that their conceptual understanding in this area improved substantially. The observed advancement was large, as reflected in Cohen's $d = 0.977$. Based on these findings, H3 is confirmed.

A comparison of students' performance in understanding the loop that encapsulates a single command showed no statistically significant difference between the results ($t(21) = 1.821$, $p = .083$). It turned out that, on average, the students scored slightly higher on the pre-test ($\bar{x} = 0.45$) than on the post-test ($\bar{x} = 0.32$). In both cases, the mean values were low, indicating that most students did not understand the loop with one command, which means the H4 cannot be confirmed.

However, a statistically significant difference was found in students' understanding of the loop consisting of several orders ($t(21) = 4.183$, $p = .001$). The arithmetic mean in the pre-test ($\bar{x} = 0.32$) was lower than in the post-test ($\bar{x} = 0.77$), indicating that the students more often scored a point and solved the task correctly after the KUBO classes. Also, the progress in knowledge was high as evident in Cohen's $d = 0.892$, which further indicated that the pupils developed conceptual understanding regarding loops with multiple commands, facilitated by help KUBO learning environment. This confirms H5.

There was a statistically significant difference between Test 1 and Test 2 ($t(21) = 3.454$, $p = .002$) when the task of finding the shortest possible route is considered. According to Cohen's scale, the effect size was of medium strength (Cohen's $d = 0.725$), indicating that they have acquired this knowledge and that the H6 is confirmed.

## Conclusion

In conclusion, this article serves as a comprehensive exploration of the intersection between early programming education, computational thinking, and the practical application of these concepts through KUBO robotics. It underscores the theoretical foundations while emphasising the tangible benefits derived from their implementation in educational contexts, thereby promoting a holistic understanding of this pivotal subject matter.

Results show that despite the few lessons, students advanced in understanding basic programming concepts. The most significant differences and the medium-sized influence, according to Cohen's $d$, were observed in tasks involving the reading of command sequences, where students had to identify the missing order and correct the path. Substantial gains were also evident in students' understanding of loops with multiple commands, and in a task where they had to find the shortest path in a similar-to-real-life situation. Overall, it can be said that all the tasks were better understood and solved on post-test.

Based on the results, we conclude that with planned activities, computational thinking was advanced in the context of learning about the environment since pupils, apart from one task, achieved a better results on average after the KUBO activities performed. Developing computational thinking skills in environmental education is essential because it facilitates data analysis for informed decision-making, enables modelling and simulation of environmental systems, promotes problem-solving for complex environmental challenges, enhances the utilisation of technology in monitoring and conservation efforts, encourages a multidisciplinary approach to understand interconnected environmental issues, etc. Overall, computational thinking equips learners with the tools needed to address environmental issues effectively and make informed choices for a sustainable future.

Since the survey involved 22 pupils only from 3rd grade, the results cannot be generalised or transferred to other steps. However, it can be argued that KUBO robotics is a suitable and valuable tool for developing computational thinking, especially in the early educational period. It can be an excellent introduction to the world of programming. By fostering computational thinking in environmental education, we can equip the next generation with the tools they need to make informed decisions in life.

## References

BALANSKAT, A., ENGELHARDT, K. (2015). *Computing our future: Computer programming and coding—Priorities, school curricula and initiatives across Europe*. European Schoolnet. https://www.researchgate.net/publication/284139559_Computing_our_future_Computer_programming_and_coding-_Priorities_school_curricula_and_initiatives_across_Europe

BERS, M. U. (2019). Coding as another language: A pedagogical approach for teaching computer science in early childhood. *Journal of Computers in Education, 6*(4), 499–528. https://doi.org/10.1007/s40692-019-00147-3

BERS, M. U., GONZÁLEZ-GONZÁLEZ, C., ARMAS-TORRES, M. B. (2019). Coding as a playground: Promoting positive learning experiences in childhood classrooms. *Computers & Education, 138*, 130–145. https://doi.org/10.1016/j.compedu.2019.04.013

BRENNAN, K., RESNICK, M. (2012). I am using artefact-based interviews to study the development of computational thinking in interactive media design. In *New frameworks for studying and assessing the development of computational thinking*. American Educational Research Association Meeting, Vancouver. http://scratched.gse.harvard.edu/ct/files/AERA2012.pdf

CSIZMADIA, A., CURZON, P., DORLING, M., HUMPHREYS, S., NG, T., SELBY, C., WOOLLARD, J. (2015). *Computational thinking: A guide for teachers*. Computing At School. http://computingatschool.org.uk/computationalthinking

GROVER, S., PEA, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher, 42*(1), 38–43. https://doi.org/10.3102/0013189X12463051

HASESKI, H. I., ILIC, U., TUGTEKIN, U. (2018). Defining a new 21st century skill—Computational thinking: Concepts and trends. *International Education Studies, 11*(4), 29–37. https://doi.org/10.5539/ies.v11n4p29

HELJAKKA, K., IHAMAKI, P., TUOMI, P., SAARIKOSKI, P. (2019). Gamified coding: Toy robots and playful learning in early education. In *2019 International Conference on Computational Science and Computational Intelligence (CSCI)* (pp. 800–805). https://doi.org/10.1109/CSCI49370.2019.00152

KOLAR, M., KRNEL, D., VELKAVRH, A. (2011). *Učni načrt, Program osnovna šola, Spoznavanje okolja.* Ministrstvo za šolstvo in šport; Zavod RS za šolstvo. http://www.mss.gov.si/fileadmin/mss.gov.si/pageuploads/podrocje/os/devetletka/predmeti_obvezni/Spoznavanje_okolja_obvezni.pdf

KRAJNC, R., KOŠIR, K., ČOTAR KONRAD, S. (2017). Računalniško mišljenje—Kaj je to in zakaj bi ga sploh potrebovali? *Vzgoja in izobraževanje, 48*(4), 9–19.

STERLING, L. (2015). An education for the 21st century means teaching coding in schools. https://theconversation.com/an-education-for-the-21st-century-means-teaching-coding-in-schools-42046

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

WING, J. M. (2011). Research notebook: Computational thinking—What and why? *The Carnegie Mellon University School of Computer Science Magazine*, 20–23.

WOLFRAM, C. (2016). *Anchoring computational thinking in today's curriculum.* http://www.conradwolfram.com/home/anchoring-computational-thinking-in-to-days curriculum

# *POUČAVANJE PRIRODOSLOVNIH I DRUŠTVENIH SADRŽAJA UZ KORIŠTENJE KUBO ROBOTIKE*


**Sažetak**

Računalno razmišljanje definira se kao jedna od ključnih kompetencija suvremenoga doba jer objedinjuje različite kognitivne sposobnosti i vještine koje pojedincima omogućuju uspješno sudjelovanje u modernome društvu. Stoga je važno započeti njegov razvoj već od osnovnoškolske dobi. U ovome radu predstavljene su aktivnosti usmjerene na razvoj računalnoga razmišljanja kroz sadržaje prirode i društva, obuhvaćajući odgojno-obrazovne ishode iz tematskoga područja *Prostor* u trećem razredu osnovne škole. Kroz zadatke potpomognute KUBO robotikom učenici su učili rješavati probleme, sustavno obrađivati informacije, logički i algoritamski razmišljati, primjenjivati apstrakciju i generalizaciju, surađivati te su, ponajprije, imali priliku biti mentalno aktivni tijekom cijeloga procesa učenja. U empirijskome dijelu rada prikazano je da su, unatoč ograničenome broju nastavnih sati, učenici nakon provedenih aktivnosti ostvarili napredak i bolje razumijevanje osnovnih programerskih pojmova. Aktivnosti su se pokazale zanimljivima jer su učenici bili izrazito motivirani za rad s KUBO robotom, koji se pokazao kao koristan i primjenjiv alat za poučavanje početnoga programiranja.


**KLJUČNE RIJEČI**: *računalno razmišljanje, priroda i društvo, KUBO robotika, poučavanje*