



HAL
open science

Introduction to the papers and posters of TWG11: Algorithmics

Christof Weber, Janka Medova, Maryna Rafalska, Ulrich Kortenkamp, Simon Modeste

► **To cite this version:**

Christof Weber, Janka Medova, Maryna Rafalska, Ulrich Kortenkamp, Simon Modeste. Introduction to the papers and posters of TWG11: Algorithmics. Twelfth Congress of the European Society for Research in Mathematics Education (CERME12), Feb 2022, Bozen-Bolzano, Italy. hal-03808530

HAL Id: hal-03808530

<https://hal.science/hal-03808530>

Submitted on 10 Oct 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Introduction to the papers and posters of TWG11: Algorithmics

Christof Weber¹, Janka Medova², Maryna Rafalska³, Ulrich Kortenkamp⁴, and Simon Modeste⁵

¹University of Teacher Education Lucerne, Switzerland; christof.weber@phlu.ch

²Constantine the Philosopher University in Nitra, Slovakia; jmedova@ukf.sk

³University of Côte d'Azur, France; maryna.rafalska@univ-cotedazur.fr

⁴University of Potsdam, Germany; ulrich.kortenkamp@uni-potsdam.de

⁵University of Montpellier, France; simon.modeste@umontpellier.fr

Keywords: Algorithms, algorithmic thinking, teaching and learning of algorithms.

In CERME12, our working group “Algorithmics” started its work as a newly established TWG. Since algorithms have always been at the heart of mathematics and their importance has been steadily increasing since the beginnings of theoretical computer science, the design and analysis of algorithms – called *algorithmics* (Traub 1964, Knuth 1985) – lies at the intersection of mathematics and computer science. For this reason, on the one hand, various algorithms and algorithmic activities have their traditional place in mathematics curricula at all levels. At the school level, mathematics and computer science have interacted since the 1980s, when many schools set up labs with computers equipped with programming software. On the other hand, many questions arise in the context of teaching and learning algorithms: a first, more applied group of questions aims at algorithms in mathematics education and curricula, a second, more theoretical group of questions seeks to clarify the concepts of algorithm and algorithmic thinking.

Conference presentations

Due to the Corona pandemic, the conference was held as a virtual event. Nevertheless, a total of 11 papers and 7 posters were presented remotely by their authors at the conference, with a total of 24 group participants from 11 countries. The contributions were considered in four themes, as follows.

Theme 1: Beliefs and domains in which algorithmic thinking occurs

A first group of papers focuses on the place and importance of algorithms in mathematics in general and arithmetic in particular. They assess the beliefs of experts about the role of algorithms in mathematics and mathematics education or their role in mathematics courses.

- Lockwood, DeJarnette, Thomas and Mørken offer three perspectives on algorithms, particularly in computational settings: an algorithmic approach in a mathematical example, the view of a mathematician, and the view of an undergraduate students taking a course in mathematics.
- Geraniou and Hodgen interviewed two mathematics educators who had experience using technology to solve mathematical problems, and they, too, shared very different views on algorithms in mathematics education, one even not seeing the use of algorithms as a mathematical activity.
- Kortenkamp analyzes an arithmetic course for pre-service primary teachers. He identifies several algorithmic activities in the topics covered in the course, such as designing algorithms, specifying algorithms, performing algorithms, proving their correctness, and comparing algorithms.

- Leifeld and Rezat's poster provides a thorough analysis of the possibilities of certain arithmetic algorithms for addition and subtraction to deepen students' understanding of inverse operations.

Theme 2: Teaching and learning of algorithmic thinking at primary level

Another group of papers focuses on teaching and learning algorithmic thinking in primary school. They use different tasks with different goals: Some use algorithmic thinking as a means to an end (in the sense of learning a new mathematical concept), some use algorithmic thinking as a goal (in the sense of understanding a given algorithm, or developing an algorithm to solve a problem):

- Crisci, dello Iacono and Ferrara Dentice explore how primary school children can be stimulated to learn new mathematical concepts by Scratch. In their report, they present a specially designed task that required visual programming to complete a given figure so that it becomes axially symmetric. The children developed two different strategies to solve this task. For example, they found out that points that are axially symmetrical to each other must be equidistant from the axis.
- Funghi and Ramploud are interested in how to teach the standard long-division algorithm so that children understand why it "works." To this end, they had fourth graders compare the optimized, digit-by-digit long-division procedure with the procedure in which the divisor is repeatedly subtracted from the dividend. Their analysis of class discussions suggests that this approach could actually result in less rote learning, but in more conceptual learning.
- Zindel's study wants children to acquire algorithmic thinking without using computers. In her papers, children are instructed to decrypt and encrypt certain words. They had to articulate the necessary steps themselves and record them in writing. Although these texts show great differences, the author succeeds in reconstructing some constituents of algorithmic thinking.
- In Gaio's study, too, children are asked to develop algorithms in the sense of systematic procedures, without the help of computers. Here, the tasks given to children of different school levels (3rd to 8th grade) are in the context of sorting problems. As the author reports, he can see traces of classical algorithms in the procedures that the children have worked out cooperatively.

Theme 3: Teaching and learning of algorithmic thinking at university level

Concerning the development of algorithmic thinking at university, the discussions showed two big issues: the development of algorithmic thinking and algorithmics in mathematics, for students, independently of their projects, and more specifically, the development of algorithmic thinking in mathematics for future teachers, and in particular future primary teachers.

Four papers dealt with algorithmic thinking in advanced mathematics, three at university, and one concerning an education program for gifted students. Above them, three were interested in links with discrete mathematics, combinatorics, graph theory, which illustrates the specificity of those mathematical fields, at the interface with computer science:

- De Chene and Lockwood explore the use of programming and computer science in solving basic counting/combinatorics tasks in college, and how the knowledge of student in computer science can influence their solving strategies and support their learning.
- Medová, Milicic and Ludwig study the competencies involved in algorithmic thinking for university, and in particular abstraction, modelling, and visualization skills which are difficult to master for students, and questions the development of computational thinking in mathematics.

- Bóra and Gosztonyi analyze the place given to algorithms and what could be seen as algorithmic problem solving, in Hungary's advanced mathematics programs, questioning what can be considered traditionally as algorithmic in mathematics and its place according to mathematical culture of the country.
- The paper of Calor, Palha and Kubbe concerns at the same time advanced mathematics and pre-service secondary teachers' education. It deals with analysis, and in particular developing instructional material concerning differential equations for algorithmic thinking and programming. First results show that students did indeed develop algorithmic thinking in their work.

The two other contributions dealt with algorithmic thinking in and for primary teacher training:

- Weber's paper examines primary teachers' use of loops to solve a geometrical problem and their conceptions of the loop construct. It elaborates some challenges in their conceptions and some misconceptions that require deepening their understanding from a teacher training perspective.
- Dobgenski and da Fontoura's poster presents and reflects on an experience of making pre-service primary teachers deal with computational thinking using Scratch.

Theme 4: Concepts related to algorithmic thinking: computational thinking, algebraic thinking, problem solving, and mathematical literacy

The last group of contributions deals with different, no less relevant aspects of algorithmic thinking:

- Rafalska's paper illustrates how tasks could be constructed in order to lead children in mathematics lessons (without the use of computers) to algorithmic thinking in the sense of developing a solution strategy and which individual learning processes can be triggered by these tasks.
- Pohlkamp and Lengnink's paper takes a different look at algorithms: It discusses algorithms that make decisions and are thus socially relevant. Addressing and studying them in the classroom would mean taking more seriously the educational mandate to teach social skills as well.

Finally, two poster proposals deal with two concepts related to algorithmic thinking:

- Rekstad and Rasmussen investigate the question to what extent aspects of computational thinking mentioned in the literature are also reflected in teachers' beliefs when asked about the role of computational thinking in mathematics education.
- The relationship between algorithmic and algebraic thinking is the subject of Müller-Spáth, who plans to investigate how algorithmic thinking (realized by an app) affects the development of the ability to generalize and thus of algebraic thinking.

Conference discussions

As mentioned earlier, our working group has just begun its work, and a common understanding of the concepts has yet to be developed: What does algorithmics mean in the context of teaching and learning mathematics? What is algorithmic thinking? To this end, after the presentations in which quite different views were expressed, we worked on the following three questions:

Question 1: Which mathematical algorithms could stimulate algorithmic thinking?

The discussion of this question revealed relatively unanimously five mathematical types of algorithms: i. Algorithms based on the place value system (standard algorithms for addition etc., algorithm for calculating logarithms), ii. graph-theoretic algorithms (shortest path problem, Königsberg

problem), iii. approximation algorithms (Heron's algorithm, Newton's method, etc.), iv. sorting algorithms (heap sort, bubble sort, etc.), and v. miscellaneous (Gauss's Easter algorithm, etc.). We were not in agreement of whether each procedure is also an algorithm. For example, everyday procedures (tying shoes, making jam sandwiches, etc.) were not considered by all participants to be suitable for addressing and promote algorithmic thinking in its "proper sense" because they show only one characteristic feature of algorithms: the order of steps.

Question 2: Which mathematical topics could promote algorithmic thinking?

The discussion of this somewhat broader question also yielded five topics from which tasks could come to stimulate algorithmic thinking: i. number theory (arithmetic, prime number tests, factorization), ii. discrete mathematics (graph theory, combinatorics, counting problems, etc.), iii. geometry (transformations, algebraic geometry), iv. computer science (cryptography, etc.), and v. games and puzzles (Rubik's cube, tower of Hanoi, etc.). One participant's question about what properties these fields would have in common was discussed intensively and controversially.

Question 3: What (human) activities with algorithms can we think of?

The activities discussed suggest a wide range of possible activities to deal with algorithms: i. creating (developing algorithms, improving algorithms, debugging algorithms, optimizing algorithms, transferring algorithms to an analogous situation, etc.), ii. analyzing (effectiveness and proof, efficiency, complexity, stability, similarity etc.), and iii. comparing (comparing different algorithms for the same problem, comparing analogous algorithms for different problems, classifying algorithms, etc.). Although executing an algorithm without any reflection would be a possible activity with algorithms, most participants do not want this to be understood as algorithmic thinking.

Surely the reader can think of further examples or answers to these questions. In other words, the three questions need to be discussed further and their answers are still quite open.

Outlook

As the overview of the contributions as well as the first answers to central questions show, there is a great variety of approaches (theories, methods) and views (topics, perspectives) in our working group. Given that we are entering a young (or at least long-neglected) area of research in mathematics education and that we have just begun work in our TWG, it was to be expected that the results would be disparate and sometimes controversial. However, in terms of a first step towards a robust and sustainable understanding of concepts, this diversity makes us confident that there are many more questions around the challenging topic of algorithms and algorithmic thinking that are worth working on. With this in mind, we look forward to CERME13 and hope for a fruitful continuation of the work we have begun – and that it can then be carried out again as a physical conference.

References

- Knuth, D. E. (1985). Algorithmic thinking and mathematical thinking. *The American Mathematical Monthly*, 92(3), 170–181. <https://doi.org/10.1080/00029890.1985.11971572>.
- Traub, J. F. (1964). *Iterative methods for the solution of equations*. Prentice-Hall.