THE IMPACT OF ALTERNATIVE METHODS OF TEACHING MATHEMATICS ON THE COMPUTATIONAL THINKING OF PRIMARY SCHOOL STUDENTS

[a]LUCIE BRYNDOVA, [b]MILAN KLEMENT, [c]EVA BARTKOVA, [d]KVETOSLAV BARTEK

Palacký University Olomouc Faculty of Education, Žižkovo nám. 5 779 00 Olomouc, Czech Republic.
[a]lucie.bryndova@upol.cz, [b]milan.klement@upol.cz, [c]eva.bartkova@upol.cz, [d]kvetoslav.bartek@upol.cz

Published within the frame of the University Palacky Faculty of Education grant *Investigating the impact of alternative methods of mathematics instruction on the development of levels of computational thinking and related computational concepts in elementary school students*, GFD_PdF_2023_01.

Abstract: The article presents the findings of a research focused on the comparison of the development of computational thinking in pupils taught by an alternative method of teaching mathematics and the classical method of teaching mathematics. The research builds on the findings of previous research, which suggests a link between the development of some dimensions of students' computational thinking and the development of mathematical thinking. We built on previous published research assumptions, which suggested that students taught in a constructivist way of teaching mathematics perform better on a test of computational thinking and thus potentially have better developed computational thinking. The study presented in this article tested 741 pupils aged 10 to 13 in the Czech Republic taught using an alternative constructivist method of teaching mathematics (the so-called "Hejny method") and pupils taught using the "classical" predominantly non-constructivist method of teaching mathematics. Statistical results show that the chosen teaching method does not affect the development of pupils' computational thinking.

Keywords: CT, Computational Thinking, mathematical thinking, computer science in education, development of key skills.

1 Introduction

The targeted development of computational thinking (CT) in primary education is currently one of the great challenges of modern pedagogy (CSTA&ISTE, 2011; European Commission, 2020). In pedagogical discourse, computational thinking represents a set of cognitive abilities, skills and approaches to analyse and solve a complex problem, that is to say abilities which are important for an individual's personal development, active participation in society and future employment in life (Perlis, 1962; Wing, 2006; Zapata et al., 2021). Yet, CT is still a relatively new concept, the understanding of which is evolving almost in real time. What is certain, however, is that it is a key capability with applicability beyond the confines of computer science and digital technologies, and its development therefore cuts across many disciplines, particularly as it relates to STEM related subjects.

Nowadays, a growing number of authors (Rambally, 2016; Pérez, 2018; Wu & Yang, 2022) are focusing on the possibilities of connecting CT and mathematical thinking (MT). The targeted outcomes of the development of MT pupils and the outcomes of the development of CT pupils are in many methodologies complementary or even identical. The full use of CT in solving practical problems in many cases presupposes the application of mathematics (Kynigos & Grizioti, 2018). Thus, it is possible to speak of an underlying relationship between MT and CT, the targeted development of which in the primary education environment is most likely to be influenced by each other.

Previous testing of pupils in the Czech Republic aimed at determining the level of development of CT in fifth grade pupils suggested a possible correlation between the method of mathematics teaching and the level of development of individual CT concepts (Bryndová & Bártek & Klement, 2023). This qualitative research has shown that pupils who were taught mathematics according to a non-traditional method of teaching mathematics, performed better on the domestic CT test than their classmates who were taught using the traditional method of teaching mathematics. This alternative teaching method was a constructivist approach aimed at building a network of mental mathematical schemas known as the Hejny method.

The verification of such a tendency is particularly important for the investigation of the development of the dependence of CT on the development of related mathematical concepts. This paper therefore focuses on examining this tendency in a broad sample of the population of fifth grade students in the Czech Republic.

2 Alternative Mathematics Teaching – Hejny method

The Hejny method of teaching mathematics focuses purely on the development of mathematical thinking and competence. On the other hand, this method cannot be classified as a method which can be applied in the educational process in isolation according to individual, specific differences in the teaching style of the teacher. It is a comprehensive system of teaching mathematics, probably the most widespread of the alternative systems at the first level of primary schools at present.

The Hejny method - as it is currently perceived, is very close to Montessori pedagogy. It is based on pedagogical constructivism (or the constructivist conception of learning, see Průcha p. 77), for which the learner's own activity is crucial in order to construct his or her own knowledge on the basis of their own (in our context mathematical) activity. In this way, both of the educational systems mentioned above seek to avoid or eliminate the formalism which is present to a greater or lesser extent in traditional forms of teaching. On the other hand, Průcha (ibid.) also mentions critical voices against the above-mentioned concept in terms of overestimating the importance of the mechanisms of knowledge construction by the learner and underestimating the role of the teacher in the educational process.

3 Hejny method of teaching mathematics

In contrast to traditional mathematics teaching focused on practicing standard problems, the Hejny method focuses on building a network of mental mathematical schemas created by each student by solving appropriate problems and discussing his or her solutions with classmates. The goal of the Hejny method of mathematics education is to make the child discover mathematics on his/her own and with pleasure. It does this by respecting the 12 key principles, which are put together in a coherent concept. These key principles will now be described in more detail (H-mat, 2024a):

- Building Schemes

Hejny method can also be referred to as Schema-oriented teaching. A schema is a collection of interconnected knowledge related to a known environment. As the author of the teaching method himself states, "Mathematical schemas are also strongly interconnected. For example, the schema of the concept of a rational number is formed by connecting the schemas of the concepts of natural number, fraction, decimal number, and negative number."

- Working in environments

The environment contains a series of interlocking problems with the same topic. These tasks are designed to encourage pupils to experiment, discover and interlink different mathematical phenomena.

- Interlinking topics

By interlinking topics through different learning environments or different tasks, concepts, processes and phenomena are well understood and better consolidated.

- Personality development

The Hejny method encourages children to think independently, teaches pupils to discuss, argue and evaluate. It encourages diversity of opinion.

- Genuine motivation

This principle is based on the so-called intrinsic motivation of a person. A child with an intrinsic need to know knows more intensely, deeply and comprehensively.

- Real experience

Mathematics teaching oriented towards the construction of diagrams is based primarily on children's own experience, from which the child can subsequently make general judgements.

- The joy of mathematics

Mathematical environments in textbooks are designed to allow pupils to discover. Even the difficulty of the problems is set so that each pupil can achieve the joy of success.

- Own observation

This principle is based on the belief that knowledge gained by one's own reasoning is of higher quality than knowledge which is taken in. The pupil solves problems and collects a range of experiences to discuss with classmates and the teacher. Afterwards, they test their theories on other tasks.

- The role of the teacher

The teacher becomes a mentor rather than an authority in the mathematics classroom. He is the one who organises the lesson, encourages the pupils to work, sets appropriate problems and guides their discussion.

- Working with error

Error handling plays a very important role in the Hejny method. It is used as a means of learning. Children are encouraged to identify the error themselves and to try to explain why they made it.

- Appropriate challenges

Textbooks contain tasks of all difficulties. The teacher distributes the tasks within the class according to pupils' needs in order to support their continuous motivation.

- Encouraging collaboration

Pupils are given plenty of space to work together and discuss directly in mathematics lessons.

3.1 Linking mathematics and computer science teaching through the Hejny method

The Hejny Method uses a variety of learning environments to help students understand mathematical concepts and the relationships between them in a playful and enjoyable way. One of the learning environments linking mathematics and computer science is the Flowchart environment.

"This environment is used to graphically record an algorithm or general process. From a computer science point of view, working with a condition appears in a flowchart: "If something is true, then do something, if it is not true, do something else." From a mathematical point of view, students are introduced to statements, deciding on their truth. An important part of this process is collecting the data produced by a flow chart and evaluating it" (H-mat, 2024b).

One of the first tasks through which students are introduced to the new learning environment of Flowcharts is seen in Figure 1.

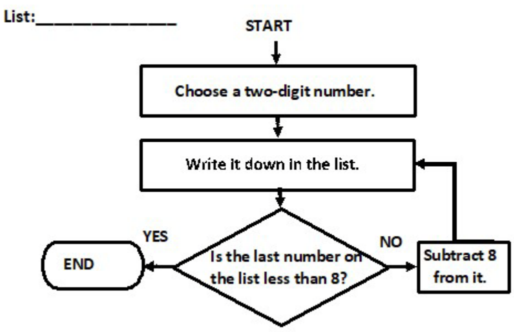

Figure 1.: Quotient with remainder (based on Hejný et al.,2021)

Students follow a series of instructions, continuously making decisions about the validity of the phenomenon, and collecting data to be evaluated later. These are recorded in a "List" item.

Let us now choose number 35. In this particular case, the "List" will contain the following numbers: 35, 27, 19, 11, 3, that means numbers obtained by repeatedly subtracting 8 from 35. The operation of division is therefore represented by this flow chart as repeated subtraction. The last number in the List is the remainder after dividing by 8. Thus, 35: 8 = 4 (col. 3). The number 4 is called the Quotient with remainder and is given by the number of numbers after the input number in the "List" (we subtracted the number 8 four times).

Another way in which we can use the Flowcharts environment is when introducing the concept of the greatest common divisor of two natural numbers (see Figure 2). When determining the greatest common divisor of two numbers, we look for the largest number, which divides the two given numbers without remainder.

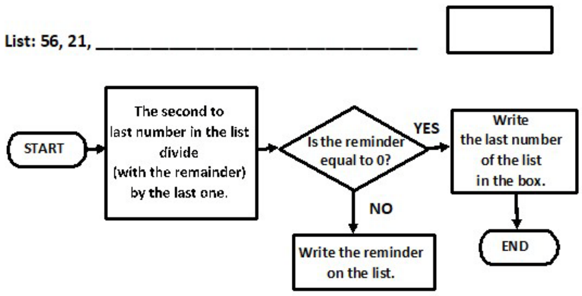

Figure 2.: Greatest common divisor (based on Hejný et al., 2021)

In this particular case, the list will contain the following numbers: 56, 21, 14, and 7, which is the largest number, which divides the first two numbers in the List.

This method of finding the greatest common divisor of two natural numbers is known as the Euclidean algorithm. Valid:

$56 = 21 . 2 + 14$

$21 = 14 . 1 + 7$

$14 = 7 . 1 + 0$

3.2 The Hejny method of teaching mathematics in the context of pupils' skill development

The Institute for Research and Development of Education of the Faculty of Education of Charles University published the final report of Hejny's Method of Teaching Mathematics in International Research TIMSS in March 2022.

"This report presents the results of secondary analyses of data from the 2015 and 2019 TIMSS (Trends in International Mathematics and Science Study) international surveys in relation to the Hejny method of teaching mathematics. In addition to the commonly available TIMSS data, information on the teaching method used, collected by the Czech School Inspectorate, was used as a national supplement to the data collection for each class involved in the testing" (Greger et al., 2022, p.4). The study, among other things, addressed the question of whether there are differences in the average results of pupils using the Hejny method and pupils who do not use the Hejny method. It was found that *"students taught with the Hejny method scored statistically significantly higher in mathematics in both 2015 and 2019"* (Greger et al., 2022, p. 30).

Recent research suggests that the development of mathematical skills and computational thinking are closely related (Rambally, 2016; Pérez, 2018; Wu & Yang, 2022). There are links between the dimensions in computational thinking and metacognitive knowledge, experience, monitoring and mathematical modelling skills (Zhang, 2024). Our research on the development of the dimensions of computational thinking in pupils in Czech schools carried out by 2023 suggested the possibility that this development might depend on the method of teaching mathematics (Bryndová & Bártek & Klement, 2023).

The data suggested a greater development of computational thinking in pupils taught with Hejny method compared to the global sample tested (about 8%). The tested sample of pupils showed better skills in algorithmizing (6% better than the parallel sample of pupils taught with classical mathematics), abstraction (5.8% better), and syntax (15% better) (Bryndová & Bártek & Klement, 2023).

However, this sample was not primarily selected with regard to the method of teaching mathematics and was therefore too small to yield conclusive conclusions. Therefore, we aimed to create a diagnostic tool aimed at determining the level of development of computational thinking in students, which would adequately assess the development of individual pupils' abilities depending on the method of teaching mathematics.

4 Research tool

Several types of assessment tools are currently described which can be used to determine the level of a student's computational thinking. Typically, depending on the definition of CT, these are tests, which incorporate the testing of individual computational concepts or subdomains. This division allows testing the development of the learners separately in each domain (Román-González, 2015; Chen et al., 2017; Zapata et al., 2021).

Thus, for the purpose of testing the level of CT development, we used the definition based on CT sub-dimensions, which are Syntax and Coding, Algorithmic thinking and Abstraction and Decomposition. This division has been used in the context of CT testing in previous research (Bryndová & Bártek & Klement, 2023), the results of which are being followed up. The sub-dimensions were defined as follows:

- Syntax and coding

In the context of primary school education in the Czech Republic, we include this dimension in the complex of abilities and skills which should be possessed by an IT-minded student. We define it as the ability to write solutions using an adequate programming language or code, at a level appropriate to the age of the pupil. It is further defined as the understanding of the principle of this notation, the ability to copy and emulate the code as specified, the orientation to the code and the solution procedure, the respect of the laws of notation, the ability to rewrite the solution so that it can be understood by a computer or an adequate machine, and other related computational procedures and perspectives related to the notation of the solution.

- Abstraction and Decomposition

In this category we include simplification of the problem into its basic form or into parts so that essential information is not lost, the ability to select and solve important parts of the problem and ignore irrelevant parts, rough path planning, abstract solution design, the ability to work with diagrams and schematic forms of the problem, etc.

- Algorithmic thinking

Algorithmic thinking is defined for this testing as concrete planning of a solution to a problem with steps, the ability to design a solution to a problem before it is rewritten into formal code, creating and reading flowcharts, and informal coding.

The test tasks were subsequently designed to be independent of a particular programming language or environment, which may vary depending on the individual school. At the same time, we required that the test tool could be deployed in the classroom without the need for specialized software (graphical tasks with printing and manual completion options).

Each test task consisted of a complex task, the solution of which required the use of a certain dimension of computational thinking. Specifically, we focused on the dimensions of Algorithmic Thinking, Abstraction and Decomposition, and Syntax and Coding. The constructed set of tasks was subjected to an expert review, in which 22 experts from among computer science educators and methodologists participated. The prerequisites for the selection of the evaluators were a qualification in computer science teaching, experience in the development of computational thinking and at least five years of experience. These experts evaluated the difficulty of the test question for a fifth-grade student, validity and the specific dimension predominantly represented in the solution of the question. The following figure illustrates the design principle of the tasks created.

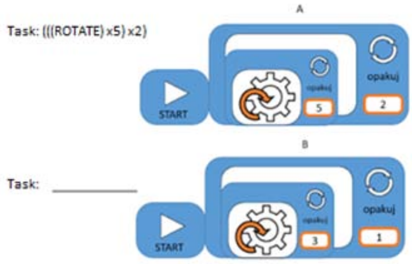

Figure 3.: Example of a test task focusing on the syntactic dimension of computational thinking

4.1 Statistical verification of the measuring instrument

Following statistical validation of the properties of the developed didactic test based on data from the universal testing of primary school pupils confirmed that the developed didactic tool meets the necessary prerequisites for a standardized test. The standard requirements for didactic test sensitivity, i.e., for Q = 20 to 30 and Q = 70 to 80, d≥ 0.15 and for Q = 30 to 70, d ≥ 0.25, were required as satisfactory sensitivity values. Since none of the questions with extreme difficulty ranks scored d < 20 and none of the moderately difficult questions scored d < 25, the minimum test sensitivity requirements were met.

Table 1: Calculation of the difficulty value and ULI for the final version of the test

Question number	Value Difficulty (Q)	ULI (*d*)
1	20	0,28
2	24	0,24
3	37	0,3
4	32	0,32
5	30	0,38
6	44	0,26
7	53	0,35
8	55	0,25
9	60	0,35
10	57	0,4
11	72	0,33
12	64	0,33

Determining the reliability of the final set of questions according to the Spearman-Brown formula: for the final, medium version of the test, the Pearson correlation coefficient was calculated as *r* = 0.432071794. Thus, the reliability coefficient according to the Kuder-Richardson formula takes values r_{sb} = 0,603.

Although the reliability coefficient is close to the cut-off value typical for a test with a low number of questions (0.60), it is satisfactory for the given set of items (*n* = 12). However, to be on the safe side, given the low reliability value determined by halving the test set, we proceeded with a combined reliability check and performed a determination of the internal consistency measure of the items using Cronbach's alpha.

Thus, Cronbach's alpha for the final set of questions was determined as α = 0.602625667. This figure is virtually identical to the reliability coefficient calculated by the Kuder-Richardson halving method, which was determined to be 0.603. Thus, the resulting coefficient is consistent for the two methods used. Consequently, it can be concluded that the set of questions, despite the relatively low value of the reliability coefficients, fulfils the minimum requirements for a reliable test with a low number of questions (n = 12).

Validation of the difficulty *Q*, sensitivity *d* and reliability coefficients of the set of questions therefore showed that the test meets the requirements for a standardized test. At the same time, based on expert assessment of validity, it demonstrably measured the development of computational thinking.

4.2 Test sample

The sample of pupils tested consisted of 741 fifth grade pupils in primary schools in the Czech Republic. The majority, 60%, were aged 11 years, 31% were aged 10 years, 6% were aged 12 years and 3% reported "other age". In terms of gender representation, 53% (n = 394) of pupils were male and 47% (n = 347) were female. The final test was distributed to respondents between January 2023 and May 2023.

Table 2: Composition of the research sample of tested pupils

Criterion	Number of pupils by criterion	
Gender	Female	347
	Male	394
Age	10	253
	11	419
	12	39
	13	12
	Different age group	18

The collected data were first subjected to an analysis aimed at examining the dependencies between gender and age of the respondents. At this stage of the analysis, we set 2 research questions:

- Q1: Does the development of computational thinking differ between boys and girls?
- Q2: Does the age of students influence the development of computational thinking?

The following 2 hypotheses emerged:

$H_1 0$: There are no statistically significant differences between the results of boys and girls.

$H_2 0$: There are no statistically significant differences between the results of pupils of different ages in grade 5.

To evaluate the correlations between the data, we used Spearman's correlation coefficient processed using Statistica 12 software. Correlations were considered significant at the level of $p < 0.05$.

Table 3: Spearman's correlation coefficient processed using Statistica 12 software

Variable	Spearman correlation variable, significant at p < 0.05 level.		
Result	1,000000	0,12250	0,054863
Age	0,12250	1,000000	0,12250
Gender	0,054863	0,12250	1,000000

As it is clear from the table above, the differences in the success rate of the didactic test in the context of their gender and the age of the tested are statistically insignificant. Therefore, we accept both hypotheses H10 and H20. Given that these were pupils from the same classes, i.e. with the same level of education, we had expected this condition. Nevertheless, this is a crucial finding. If pupils of different ages in the same educational level were shown to have different results in the overall development of computational thinking, it would mean that there are other psychological aspects of cognitive development which have a significant influence on the development of computational thinking. This would be necessary to consider in any pedagogical intervention. At the same time, from our point of view, the results confirm that the constructed measurement tool of computational thinking works well.

5 Analysis of the survey results

First, we analysed the summary test results using primarily the non-parametric Mann-Whitney test, which does not test the agreement of the means of two independent samples but determines the extent to which their distribution functions agree. This is a non-parametric analogue of the Student's t-test.

We set the null hypothesis:

$H_1 0$= There is no statistically significant difference between the test scores of students who are educated using an alternative method of teaching mathematics (Hejny method) and students who are not educated using this method.

Against the aforementioned, we set the alternative hypothesis:

$H_1 A$= There is a statistically significant difference between the test results of pupils who are educated by an alternative method of teaching mathematics (Hejny method) and pupils who are not educated by this method.

N_1...pupils educated using the Hejny method (N_1=231)

N_2...pupils educated using the classical method (N_2=510)

Table 4: Mann-Whitney U Test showing no statistically significant difference between the test scores of students

variable	Mann-Whitney U Test (w/continuity correction) By variable Hej=1 class=0 Marked tests are significant at p<.05000				
variable	Rank Sum Group 1	Rank Sum Group 2	U	Z	p-value
Suma	84411,5	190500	57615,5	-0,482	0,6330
variable	Z adjusted	p-value	Valid N Group 1	Valid N Group 2	
Suma	-0,4816	0,63	231	510	

It is clear from the table that the null hypothesis cannot be rejected, i.e. there are indeed no statistically significant differences between the overall results of the students between the two groups. A control test (using the Kolmogorov-Smirnov test) led to the same results.

We also performed a test using the parametric Student's t-test, which confirmed our findings, too.

Table 5: Student's t-test showing no statistically significant difference between the test scores of students

variable	T-tests > Grouping Hej=1; class=0 Group 1: 1 Group 2: 2				
variable	Mean 1	Mean 0	t-value	df	p
Suma	6,7793	6,9118	-0,73558	739	0,4622
variable	Std. Dev. 1	Std. Dev. 0	Valid N Group 1	Valid N Group 0	

Suma	2.3385	2,2413	231	510	

In further analyses, we focused on a more detailed examination of the individual subcategories-dimensions of computational thinking, which corresponded to the division of four consecutive questions into three dimensions

Q1 to Q4 to determine the level of "Algorithmic Thinking"

Q5 to Q8 to determine the level of "Abstract thinking and decomposing"

Q9 to Q12 to determine the level of "Syntax and Coding"

We set null and alternative hypotheses for the summative test results in the sub-dimensions:

H_20= There is no statistically significant difference between the test results in the dimension "Algorithmic thinking" of pupils who are educated by an alternative method of teaching mathematics (Hejny method) and pupils who are not educated by this method.

Against this, we have constructed an alternative hypothesis:

H_2A= There is a statistically significant difference between the test results of the "Algorithmic thinking" dimension of pupils who are educated by an alternative method of teaching mathematics (Hejny's method) and pupils who are not educated by this method.

Similarly, we set hypotheses H_30, H_3A for the dimension "Abstract thinking and decomposing" and H_40, H_4A for the dimension "Syntax and coding".

Table 6: Mann-Whitney U Test for the dimension "Syntax and coding"

variable	Mann-Whitney U Test (w/continuity correction) By variable Hej=1 class=0 Marked tests are significant at p<.05000				
	Rank Sum Group 1	Rank Sum Group 2	U	Z	p-value
Sum Q1-Q4	85198,5	189713	58402,5	-0,186	0,85245
Sum Q5-Q8	83653	191258	56857	-0,7586	0,44808
Sum Q9-Q12	85737	189175	588670	0,01297	0,9897
	Z adjusted	p-value	ValidN Group 1	ValidN Group 2	
Sum Q1-Q4	-0,196	0,8450	231	510	
Sum Q5-Q8	-0,787	0,4313	231	510	
Sum Q9-Q12	0,0134	0,9893	231	510	

Since the p-value is greater than the significance level, we do not have enough evidence to reject the null hypothesis. This suggests that the differences between the groups are not statistically significant, and none of the hypotheses H20, H30 and H40 could be rejected. The results of the two treatment groups in the areas of algorithmic thinking, abstract thinking, and syntax and coding do not show significant differences, and thus the method of mathematics instruction does not affect the test results of the treatment groups.

In the last stage of the investigation, we examined the results in even greater detail, going down to the level of the results of the individual questions designed to investigate the level of development of each dimension of computational thinking.

Table 7: Mann-Whitney U Test of each dimension of computational thinking

	Rank Sum Group 1	Rank Sum Group2	U	Z	Rank Sum Group1
Var6	82768,5	192142,5	55972,5	-1,0863	0,277334
Var7	86929,5	187981,5	57676,5	0,45498	0,649121
Var8	85867	189043,5	58738,5	0,0615	0,950958
Var9	82206	192705	55410	-1,2947	0,195411
Var10	85089	189882	58293	-0,2265	0,820762
New var1	89023,5	185877,5	55582,5	1,23083	0.218388
New var2	84496,5	190414,5	57700,5	-0,446	0,655531
New var3	83998,5	190912,5	57202,5	-0,6306	0,5283
Var2	84525	190386	57729	-0,4355	0,663176
Var3	85858	189052,5	58747,5	0,05817	0,953613
Var4	85716	189195	58890,5	0,00537	0,995713
Var5	84126	190785	57330,5	-0,5833	0,559648
	Z adjusted	p-value	Valid N Group1	Valid N Group2	
Var6	-1,6248	0.104194	231	510	
Var7	0,52938	0,596545	231	510	
Var8	0,07119	0,943248	231	510	
Var9	-1,5019	0,133112	231	510	
Var10	-0.2673	0.789179	231	510	
New var1	1,42125	0,155246	231	510	
New var2	-0,5185	0,604093	231	510	
New var3	-0,8151	0,415014	231	510	
Var2	-0,6305	0,52833	231	510	
Var3	0,07817	0,937696	231	510	
Var4	0,00642	0,994881	231	510	
Var5	-0,8078	0,1922	231	510	

Again, there were no statistically significant differences in responses between the two groups.

6 Summary and discussion of the results

As the above analysis shows, in the test of the level of development of computational thinking in fifth-grade pupils conducted by our standardized didactic test, no significant differences were found between the results of the two groups studied, neither in the overall result of the pupils nor in the subdimensions of computational thinking monitored by the test. Thus, when comparing the performance of pupils taught using the Hejny method of teaching mathematics and the global sample of pupils taught using the classical method of teaching mathematics, there were no statistically demonstrable differences. For both groups, the development of the level of computational thinking was almost identical.

This conclusion is based on the results of the statistical tests, which showed no statistically significant differences between the groups of pupils taught by different methods, neither in individual questions nor in the broader categories of ALG (Algorithmic Thinking), ABS (Abstract Thinking and Decomposing) and SYN (Syntax and Coding).

Thus, we were unable to demonstrate the original tendency observed for pupils from schools with parallel mathematics teaching using both the Hejny and 'classical' methods, where pupils taught using the Hejny method performed significantly better, despite having the same computer science teacher. Therefore, according to our research, it is demonstrable that pupils taught using the principles of Hejny mathematics have the same developed computational thinking as pupils taught using the classical method of mathematics teaching. Within the tested global sample, the method of teaching mathematics did not influence the students' computational thinking.

There are several possible explanations why this may be the case. First, the Hejny method focuses primarily on developing mathematical thinking through discovery and group work, which may develop skills other than those directly needed for solving computer science problems. While it may promote a deeper understanding of mathematics, this understanding may not automatically translate into better performance in computer science.

Furthermore, it is possible that the computer science problems on the test did not require skills which are specific to one or the other method of teaching mathematics. The dimensions of computational thinking studied and analyzed are areas which may be more directly influenced by the computer science instruction than by the method of mathematics instruction. Also, in both groups, students could use different strategies but with similar effectiveness, leading to comparable results.

Another factor may be the homogeneity of the groups, where differences between students may be due to other variables such as individual ability or interest in computer science, and not necessarily the method of mathematics instruction. Teacher influence, a particular curriculum or level of preparation for computer science could also play a role in the results.

Overall, we can therefore conclude that the influence of the method of mathematics teaching on achievement in computing areas is not clear-cut and that other factors, such as direct teaching of computing, may be more important in influencing these skills, also depending on the teaching style of the teacher.

Determining the method of teaching mathematics understandably represents a certain limit of research. The inclusion of individual test respondents in the group taught by the Haynes method of mathematics teaching depended purely on the teacher's statement, without dealing with other phenomena such as the teacher's approbation or the application of the method in practice. Similarly, we did not examine or investigate the extent to which teachers who do not teach their students using the Hejny method use elements of constructivism in their teaching.

7 Conclusion

Computational thinking is a new and still developing concept, the nature and implications of which for education are still being researched. Currently, the attention of experts focuses on the possibility of targeted development of computational thinking in students across subject areas. One of the explored ideas is the possible interconnection of computational and mathematical thinking. In our research, which focused on the influence of alternative teaching methods of mathematics that could potentially affect pupils' development.

On the basis of our research, we have shown that the method of teaching mathematics in our sample of Czech students did not affect the development of their computational thinking. Pupils taught by the alternative teaching method achieved statistically similar results in a test focused on computational thinking as pupils taught by the classical method of teaching mathematics.

We consider this conclusion potentially useful for targeted development and evaluation of the development of computational thinking in the population as it implies that computational thinking can be developed to the same quality regardless of the teaching style or the focus of the school. In our view, this trend corresponds to the broad potential of computational thinking in various non-STEM fields. However, the specifics of developing students' computational thinking in cross-curricular areas are still a relatively unexplored area. From our point of view, it is important that different teaching styles do not potentially affect the development of the learner and are of the same quality as those teaching styles that are considered traditional.

Literature:

1. Bryndová, L.; Bártek, K. A Klement, M. *Possibilities of Diagnosing The Level Of Development Of Students' Computational Thinking And The Influence Of Alternative Methods Of Teaching Mathematics On Their Results.* Online. AD ALTA: Journal of Interdisciplinary Research. 2023, 13, 1, 45-51.
2. CSTA & ISTE. (2011). *Operational Definition of Computational Thinking for K–12 Education.*
3. Greger, D., Chvál, M., Martinková, P., Potužníková, E., Soukup, 3. P. & Vondrová, N. (2022) *Hejného metoda výuky matematiky v mezinárodním výzkumu TIMSS – závěrečná zpráva.* Online.
4. Hejny, M., Jirotková, D., Slezáková, J., Kuřík Sukniak, A., Strnad, V., Ročák, Š. (2021) *Matematika 4. Příručka učitele pro 4. ročník.* Praha: H-mat, o.p.s. ISBN 978-80-88247-29-6.
5. H-mat, o.p.s. (2024a) *12 klíčových principů.* https://www.h-mat.cz/principy
6. H-mat, o.p.s. (2024b) *Vývojový diagram.* https://blog.h-mat.cz/didakticka-prostredi/vyvojovy-diagram
7. Kynigos, C., & Grizioti, M. (2018). Programming approaches to computational thinking: Integrating turtle geometry, dynamic manipulation and 3D space. *Informatics in Education*, 17(2), 321–340.
8. Pérez, A. (2018). A framework for computational thinking dispositions in mathematics education. *Journal for Research in Mathematics Education*, 49(4), 424–461.
9. Rambally, G. (2016). The synergism of mathematical thinking and computational thinking. In D. Polly (Ed.), Cases on Technology Integration in *Mathematics Education* (pp. 416–437). Business Science Reference.
10. Wu, W. R., & Yang, K. L. (2022). The relationships between computational and mathematical thinking: A review study on tasks. *Cogent Education*, 9(1).
11. Zhang, J.; Zhou, Y.; Jing, B.; Pi, Z.; Ma, H. *Metacognition and Mathematical Modeling Skills: The Mediating Roles of Computational Thinking in High School Students*. J. Intell. 2024, 12, 55.

Primary Paper Section: A

Secondary Paper Section: AM