DOI: https://doi.org/10.15625/2615-8957/22210108

Math and Coding in STEM education

Daryna Vasylieva[1✉], **Tetiana Hodovaniuk**[2]

[1] vasilyevadarina@gmail.com
Institute of Pedagogy of NAES
(UKRAINE)

[2] tgodovanyuk@ukr.net
Pavlo Tychyna Uman State Pedagogical University
(UKRAINE)

✉ Corresponding author

ABSTRACT: *Overall, this article focuses on the implementation of STEM education in Ukraine and the description of different models of STEM education. Specifically, it defines the concept of STEM literacy, and concentrates on the interdisciplinary links between mathematics and coding in the context of STEM education by analysing various curricula of mathematics and coding. Firstly, this paper gives examples of the Ukrainian mathematics curriculum for the 5th and 6th graders, including the necessary themes that form a mathematical base for the further study of coding and other STEM subjects. This paper also provides an overview of the students' prerequisite mathematical knowledge to take part in the 7th grade's coding class. Secondly, it provides examples of specific mathematics tasks (e.g., compiling algorithms, working with flowcharts, reading parts of codes) that help strengthen the math-coding links. A survey was sent out to mathematics and ICT teachers for data collection.*

KEYWORDS: **Mathematics, coding, interdisciplinary links, Mathematics curriculum, models of STEM education, STEM literacy.**

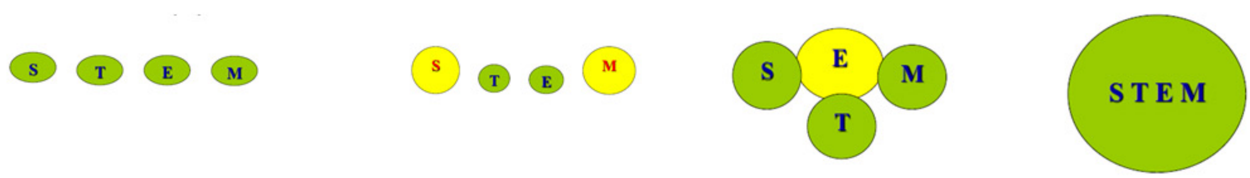
➔ Received 10/10/2021 ➔ Revised manuscript received 12/02/2022 ➔ Published 30/03/2022.

1. Introduction

The Ukraine Ministry of Education and Science, Business and IT proposed an initiative to implement STEM education in Ukraine and considers it as one of the promising innovative approaches in the process of modernizing the modern education system. In 2020, the Cabinet of Ministers of Ukraine adopted the Concept for the Development of Science and Mathematical Education (STEM-education) (The concept, 2020), and the implementation of which is scheduled for 2027. According to the Concept, training methods and curricula of STEM-education are aimed to form learning competencies that are relevant to the labor market. For instance, some of the competencies might include critical thinking, engineering and algorithmic thinking, information processing, and data analysis skills, digital literacy, creative qualities and innovation, and communication skills. Morrison (2006) also believes that STEM education is an approach that provides students with better problem-solving skills, innovation, self-confidence, logical thinking and helps them become STEM literate.

According to William Dugger (2010), there are different models of STEM education: (1) to teach each of the four STEM disciplines individually in schools (figure1a); (2) to teach each of the four STEM disciplines with more emphasis going to one or two of the four (figure1b); (3) to integrate one of the STEM disciplines into the other three being taught (figure1c); and (4) to infuse all four disciplines into each other and teach them as an integrated subject matter (figure1d).

It is agreed that in order to implement STEM education, it is not necessary to combine 4 subjects into one integrated course, and students can study subjects separately. However, interdisciplinary



(a) (b) (c) (d)

Figure 1. Models of STEM education

links should be simultaneously strengthened. Moreover, regardless of the types of model of STEM education, the main goal is to form students' STEM literacy (literacy in technology, science, engineering and mathematics).

STEM literacy is the ability to apply concepts in STEM to solve problems that cannot be solved using a single discipline or would benefit from a creative solution involving multiple disciplines (Dönmez, 2020). It is possible to promote the formation of students' STEM literacy both via individual subjects and via integrated courses. Ukraine has a regulatory framework for the implementation of any STEM education models, but the model of teaching individual STEM subjects is more popular in Ukrainian schools compared to other models. Students can gain STEM-literacy by learning four STEM fields separately, but provided that there are strong interdisciplinary links between fields and enhanced applied approach for each of them. In this case, it is important that the curricula of the four fields are agreed with each other. In conclusion, in order to lay the foundation for the implementation of STEM education, the curricula of the mathematical, scientific, engineering and technological fields must be agreed upon.

Hasanah and Tsutaoka (2019) aimed to identify and classify the barriers in light of the STEM education worldwide. In STEM education, the intrinsic barrier arises from the personality of the teacher and the student, while the extrinsic barrier mainly arises from the inadequate and/or improper arrangement of the infrastructure. Institutional barriers are concluded to be the curriculum, policy, technology and organizational feeding in the field of education. It is agreed that in order to implement STEM education, it is necessary to make changes to the policy, curricula, technology and organizational feeding in the field of education, and it is also mandatory to improve the pedagogical skills of teachers of all STEM subjects.

In order to acquire knowledge in science, engineering or technology, students must already have some knowledge in mathematics, since it is the underlying language that belongs to all fields (Wang, 2005). Yakman (2008) and Ben-Jacob (2019) further add that the basis of science communication is the language of mathematics and therefore obtaining mathematical knowledge and skills is essential in achieving integrative education. The authors of this paper, similar to Wang and Ben-Jacob, are also convinced that technology, engineering, and science are based on a mathematical basis; therefore, a certain mathematical base is a necessary condition for studying other STEM fields. To conclude, for the quality implementation of STEM education, it is extremely important that every mathematics teacher is aware of the role of Math in STEM learning. It is not only one of the four components, but it also builds a basis for acquiring knowledge from the other three fields.

In addition, mathematics has a huge potential in the formation of students' analytical, algorithmic, critical and creative thinking, which is necessary for students to understand the basics of coding easily and simply. By learning to code and thinking from the perspective of a computational scientist, students would become better problem solvers, designers, and innovators. Coding skills would also benefit students in terms of employability (Sengupta, Dickes & Farris, 2021). In the context of Ukraine, students begin to get acquainted with the elements of coding in primary school levels, but a systematic coding course only begins in the 7th grade. It is possible to introduce students to different elements of coding without a sufficient base in mathematics, but to prepare students for a systematic course, it is compulsory to determine the basic set of mathematical knowledge. To do so, it is necessary to build a productive mathematics curriculum for the 5th and 6th grades, which would not only equip students with necessary knowledge but also provide close cross-curricular links to mathematics and ICT, so that students are fully equipped to start the coding course in the 7th grade.

As stated above, this article considers the construction of mathematics curriculum (5th and 6th grades) which expectedly prepares students for the coding course in the 7th grade. Examples of exercises and teaching methods for these curricula are also provided, with an aim to strengthen the interdisciplinary links between mathematics and coding.

2. Methodology

This study analyzed (1) different mathematics and ICT curricula in Ukraine, (2) the software in which students learn at ICT lessons and (3) the teachers' survey to obtain the perspectives of teachers in STEM teaching and learning.

A survey was developed and conducted on 122 Ukrainian ICT teachers about the coding software, necessary knowledge of mathematics to successfully learn the systematic coding course at school (7th grade) and about different mathematic exercises which expectedly strengthens the interdisciplinary links between mathematics and coding. Google Form was used as a tool to design this survey, with 6 questions included. Among those, 2 questions were about the coding software, 1 question was about the math topics that students are recommended to learn before the 7th grade, and the rest concerned about methodological approaches that strengthen the interdisciplinary links between mathematics and coding in 5th and 6th grades.

The survey involved Ukrainian ICT and Mathematics teachers teaching from the 5th to the 9th grades from different educational institutions, at different ages and with a wide range of teaching experiences, who agreed to take part in this survey. Based on the analysis of the mathematics and ICT curricula and the results of the survey, an alternative mathematics curriculum was suggested for 5th and 6th grades. This curriculum was approved by the Ministry of Education and Science in 2021 and tested in 12 experimental classes for the 2021-2022 year.

3. Research results

3.1. What coding software were used in ICT lessons?

In most Europe countries, elements of coding have been introduced from primary school (up to 6th grade) levels and systematically taught at secondary schools. *Scratch* is often used to introduce coding at primary school (especially in 5th and 6th grades) levels, and *Python* is used for the 7th and higher grades. This circumstance is similar to that in Ukraine, shown via the survey results (see figure 2 and figure 3).

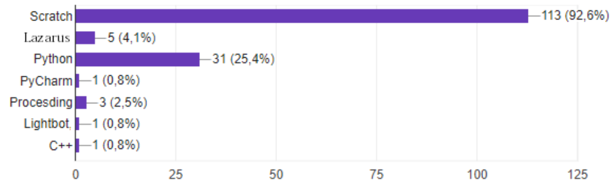


Figure 2. Coding software used for 5th and 6th grades

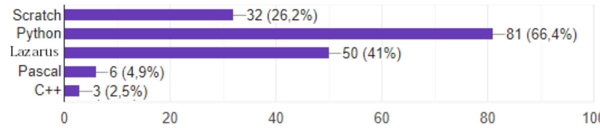


Figure 3. Coding software used for 7th to 9th grades

The survey shows that ICT teachers start introducing students to coding using different software, but the majority (92,6%) of teachers prefer to use *Scratch* (visual coding) for students in 5th and 6th grades. This suggests that in 5th and 6th grades, mathematics teachers can introduce students to the parts of code created in *Scratch,* because the interface is likely to be familiar to students. From 7th to 9th grades, the number of teachers who use *Python* (66,4%) and *Lazarus* (41%) for coding is much higher than the those using *Scratch* (26,2%).

3.2. What should be included in the mathematics curriculum?

The analysis of all existing mathematics curricula suggests the mathematical knowledge needed to learn coding, surveyed on ICT and computer science teachers.

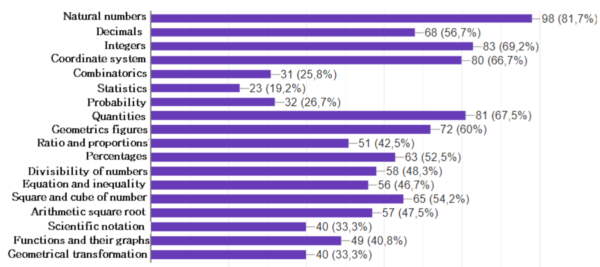


Figure 4. Mathematics topics for 5th and 6th grades

According to the survey results, learning coding requires mathematical knowledge on the following topics: (1) numbers and operation with them (including negative and positive numbers) (81,7%); (2) quantities (67,5%), (3) coordinate system (66,7%); (4) the simplest geometric shapes and their properties (60%); (5) percentages (52,5%); (6) property of divisibility (48,3%); (7) degree with an integer exponent (54,2%); (8) equations and inequalities (46,7%). It is also shown that developing formed ideas on scientific notation and geometric transformations is also desirable to prepare for learning coding.

Taking the linearity of the mathematics course into consideration, before learning a systematic course of coding, ideas on the following mathematics concepts should be formed in 5th and 6th grades: (1) simplest geometric figures (point, ray, line, segment, angle) – polygons and their properties – circle – geometric transformation of figures; (2) natural numbers – multiplication of natural numbers – degree of a number with a natural exponent – common fractions – decimals – scientific notation; (3) natural numbers – divisibility of numbers – GCF and LCM of numbers – common fractions with different denominators and (4) decimals – percentages; and (5) coordinate ray – perpendicular lines – integers – coordinate line – coordinate system. These knowledge should be properly reflected in the mathematics curriculum

Ukraine is now in the process of reforming the structure, content, and organization of education, in an attempt to transition from a knowledge-centred paradigm to a more competent paradigm. This reform was called New Ukrainian School (NUS). This new Ukrainian School model supports that students must be equipped with STEM-related skills in preparation for STEM-related professions (including the profession of a coder). This new standard on educational curriculum has been approved by the state and various model curricula for each subject are being created across Ukrainian regions. Based on the analysis of mathematics and ICT curricula and the survey results, a model curriculum of mathematics for the 5th and 6th grades of NUS (Burda & Vasylieva, 2021) was created and approved. The topics of this curriculum are provided below in the following order:

For the 5th grade:

1. Natural numbers and actions with them (up to 12 digital numbers)
2. The simplest geometric shapes (point, line, plane, angles).
3. Segments and their lengths. Polygon chain
4. Coordinate ray. Device scale
5. Parallel and perpendicular lines
6. Combinatorics
7. Properties of divisibility
8. Quantities
9. Numerical and letter expressions. Formulas
10. The degree of a number with a natural exponent.
11. Equations and inequalities.
12. Polygons. Congruent shapes. Symmetry. The axis of symmetry and the center of symmetry of the figure.
13. Triangles. Types of triangles. Inequality of triangles
14. Quadrilaterals. Inequality of a quadrilateral.
15. The sum of the angles of a triangle and a quadrilateral
16. Common fractions. Actions with fractions with the same denominators
17. Decimals and actions with them
18. Map Scale
19. Arithmetic mean. The average value.
20. Bar and line charts.

For the 6th grade:

1. Integers
2. Coordinate line
3. Coordinate system
4. Properties of divisibility. GCF and LCM of numbers.
5. Common fractions. Actions with fractions with different denominators
6. Infinite decimal fractions. Rounding
7. Ratio and proportions
8. Probability
9. Circle
10. Bar and line charts. Pie charts
11. Rational numbers
12. Dependent quantities
13. Graphs
14. Scientific notation

15. Equations. Solve problem with the equations

The authors of this paper agree that firstly, it is inexpedient to introduce the concept of arithmetic square root in 5th or 6th grades. Instead, students should learn this content in the 7th to 9th grades and this coding topic should be left for 7th to 9th graders. Secondly, such mathematics curriculum gives the possibility of learning elements of coding in the 5th grade on the basis of the properties of geometric figures, and in the 6th grade on the basis of knowledge about the coordinate system.

3.3. What should be the method of teaching mathematics?

As mentioned above, changing the mathematics curriculum is not enough to transform the current system. It is important to create necessary changes to the current approaches to teaching mathematics. Specifically, teaching methods, forms of teaching, teaching tools that strengthen the interdisciplinary links between mathematics and coding must be introduced.

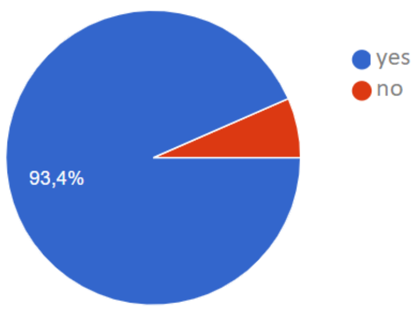


Figure 5. The expediency of compiling algorithms in Mathematics lessons (answers from ICT teachers)

From the pie chart, 93,4% of ICT teachers claimed that it would be appropriate to teach students to compose algorithms in mathematics lessons (figure 5). Mathematics teachers need to provide students with opportunities to develop their algorithmic thinking. The teacher can present some theoretical materials in the form of an algorithm, ask students to create an algorithm for solving a certain problem, or ask students to formulate the problem according to a given algorithm.

After receiving such answers from ICT teachers, 82 mathematics teachers were also asked to compare the results. Teachers were asked if they think it is appropriate to teach students how to compose algorithms, and whether they offered certain activities to help students improve their perceptions of coding.

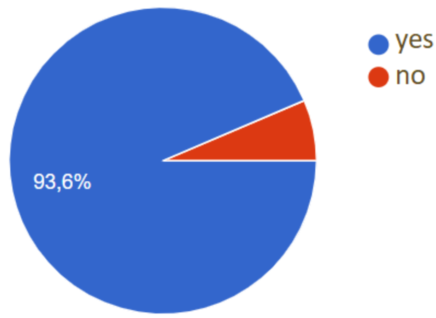


Figure 6. The expediency of compiling algorithms in mathematics lessons (answers from Mathematics teachers)

The results demonstrate that most mathematics teachers (93,6%) noted that they sometimes offered this type of activity (figure 6).

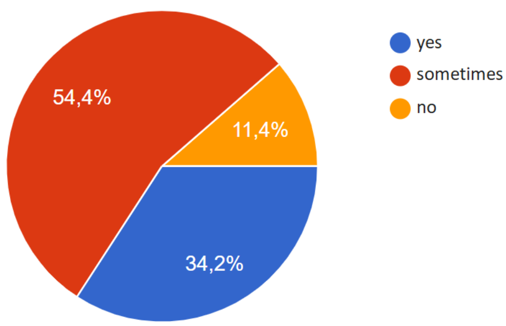


Figure 7. Proposal to compose algorithms in mathematics lessons

The pie chart suggests that not all teachers taught their students how to compose algorithms (figure 7). In addition, the percentage of teachers who did not offer such task (≈11%) is greater than the percentage of teachers who found such tasks in mathematics lessons inappropriate (≈6%). That is, there are teachers who are aware of the importance of such exercises, but at the same time do not offer them to students. It is also indicated that the majority of teachers (54,4%) offered such exercises sporadically.

The following questions of the survey

concerned the use of special exercises in mathematics lessons that contain flowcharts and parts of the code.

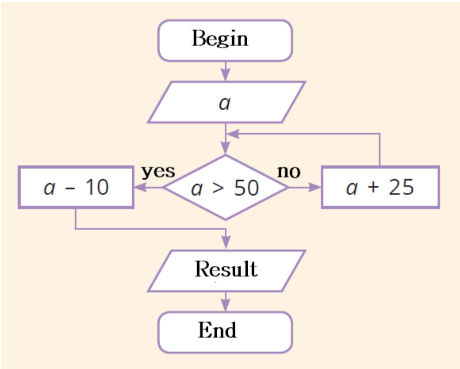


Figure 8. Example of a flowchart

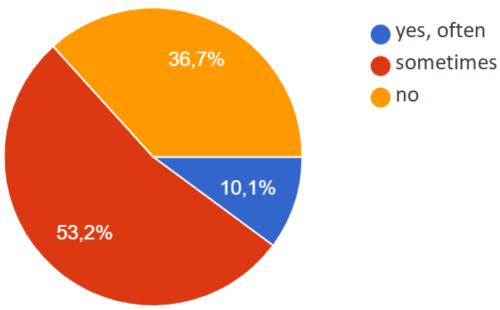


Figure 9. Proposal to work with flowcharts

The statistics suggest that 53,2% of mathematics teachers claimed that they sometimes offered their students exercises to read or build *flowcharts*, but there was still 36,7% of teachers said that they did not provide any exercises relating to flowcharts (figure 9). Only 10,1% of math teachers offered such exercises systematically and regularly.

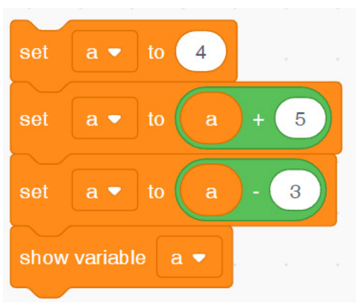


Figure 10. Example of codes part

It is shown that 35,4% of teachers did not provide students with exercises with parts of the code in mathematics lessons, and 54,4%

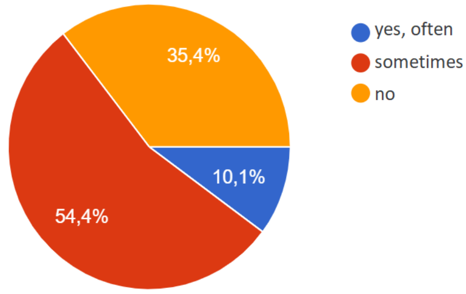


Figure 11. Proposal to work with codes part

offered such types of tasks sporadically (figure 11). Considering the results of this survey on mathematics teachers, it is clear that only about 35% of teachers systematically offered students to work with algorithms, block diagrams and parts of code in their mathematics lessons, and 10% did not offer such exercises at all.

4. Discussion

As a result of the research described in the article, the mathematics curriculum for 5th and 6th grades should be built in such a way that by the time of learning systematical coding course in the 7th grade, students have basic knowledge of: natural numbers, common fraction and decimals; quantities; numerical and literal expression, equations, inequality; geometric shapes and their properties, transformation of geometric figures; integer numbers, and coordinate system.

A new mathematics curriculum (Burda & Vasylieva, 2021) was created, approved and tested at 12 experimental classes for the 2021-2022 year. So far, this mathematics curriculum has received positive responses from mathematics and computer science teachers. The results of the probation confirmed that students should first get acquainted with the mathematical concepts, which would be useful for coding lessons.

The strong interdisciplinary links between mathematics and coding help establish not only the content but also the methods of teaching mathematics. Methodology of teaching mathematics should include the exercises of writing algorithms, reading codes and working with flowcharts. It is advised to offer such exercises in ICT lessons, but the article better concentrated on how mathematics teaching should be organized to strengthen the interdisciplinary connections between mathematics and coding.

In fact, near 30% of teachers of each survey were teachers of Mathematics and ICT, and it is likely that this group of teachers understood the need of interdisciplinary links between mathematics and coding. Thus, there is a higher chance that they offered such exercises in both Mathematics and ICT lessons. This hypothesis is confirmed by the results of the mathematics teachers' survey, which indicated that about 35% of mathematics teachers systematically provided students with these types of exercises. Most likely, these teachers are teaching both mathematics and ICT. For the vast majority of other mathematics teachers, these types of tasks are not typical.

In the field of mathematics, there is a potential to equip students with the ability to read and understand parts of the code. To do this, a mathematics teacher does not need to be well versed in software, or be able to work with students in a specific software in class. It is enough to simply offer fragments of code that can be created in different (or one) software and give students the opportunity to try to understand and decipher them (without even having the appropriate knowledge of ICT). It is also worth mentioning that in some European countries (for example, Norway, Sweden), there are attempts to introduce a block of coding into the curriculum of mathematics teaching. However, European teachers' knowledge of coding and its didactics is still limited.

5. Conclusions

The purpose of STEM education is to acquire students' STEM literacy. The student can acquire this literacy both via individual subjects and by learning an integrated course. Regardless of the education STEM models, it is important to synchronize the content of four fields: mathematics, engineering, technology and science. The formation of student STEM literacy via the learning of individual subjects contributes to the coordination of curricula of four STEM disciplines.

Mathematics is often mentioned as underpinning other disciplines of STEM because it serves as a language for science, engineering, and technology (Schmidt & Houang, 2007). That explains why more attention is paid towards the construction of the mathematics curriculum in such a way that it provides the necessary mathematical apparatus for learning other disciplines. For successful coding learning in 7th grade, the mathematics curriculum of the 5th and 6th grades should be built in such a way that students have knowledge of numbers, geometric figures, coordinate systems, quantities, equations, and inequality. An example of such a curriculum is given in this article. From a methodological point of view, mathematics teachers should keep in mind that the creation of algorithms for solving maths problems would contribute to better perceptions of coding. The authors of this paper agree with Cápay & Magdin (2013) that the development of algorithmic thinking is very important not only for STEM education, but also for a number of activities in real life. Contextually, mathematics and ICT intersect mostly in terms of algorithm, which is defined as a kind of instruction designed to solve a particular problem.

In addition, it is advisable to teach students how to read parts of code (created using one or different software) and flowcharts during mathematics lessons. Such exercises will not only strengthen interdisciplinary connections, but also positively affect students' attitudes towards mathematics (Ke, 2014) and develop mathematical thinking (Calao, Moreno-León, Correa, & Robles, 2015).

For the quality implementation of STEM education, it is necessary to train qualified teachers who would be literate in all four areas. At present time, the number of such teachers is insignificant, which affects the pace of implementation of STEM education. But even with limited resources for studying all four areas, it is still important to discuss the implementation of STEM education. This is because the curricula of these four areas are coordinated, the work of teachers in these fields is thus synchronized, and there are evidently strong interdisciplinary links between areas. It is also suggested that future research should explore the links between mathematics and web-design, as well as the construction of mathematics curriculum for 7th to 9th grades, which would be consistent with the curricula of teaching technology, engineering, and science.

Acknowledgments

We highly appreciate the cooperation of the Mathematics and ICT teachers, who took part in the survey. We would also like to send our sincere thanks to the experts and our colleagues who provided us with constructive comments.

References

Ben-Jacob, M. (2019). The importance of mathematics in STEM education. *EDULEARN19 Proceedings*, 2496–2500. https://library.iated.org/view/BENJACOB2019IMP

Burda, M., & Vasylieva, D. (2021) Model curriculum of Mathematics (5th and 6th grades). https://mon.gov.ua/storage/app/media/zagalna%20serednya/Navchalni.prohramy/2021/14.07/Model.navch.prohr.5-9.klas.NUSH-poetap.z.2022/Matem.osv.galuz-5-6-kl/Matem.5-6-kl.Burda.Vasileva.14.07.pdf

Calao, L., Moreno-León, J., Correa, H., & Robles, G. (2015). Developing Mathematical Thinking with Scratch. In: *Design for Teaching and Learning in a Networked World. EC-TEL 2015. Lecture Notes in Computer Science*, 9307(3). https://doi.org/10.1007/978-3-319-24258-3_2

Cápay, M., & Magdin, M. (2013). Alternative Methods of Teaching Algorithms. *Procedia - Social and Behavioral Sciences*, 83(1), 431–436.

Dönmez, I. (2020). STEM Education Dimensions: from STEM Literacy to STEM Assessment. *Research Highlights in Education and Science*, 154–170. https://www.researchgate.net/publication/347976048_STEM_Education_Dimensions_from_STEM_Literacy_to_STEM_Assessment

Dugger, W. (2010). Evolution of STEM in the United States. In: *The 6th Biennial International Conference on Technology Education Research*. Australia. http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=A6A14AE81341A397E4AC798DF5445181?doi=10.1.1.476.5804&rep=rep1&type=pdf

Hasanah, U., & Tsutaoka, T. (2019). An Outline of Worldwide Barriers in Science, Technology, Engineering and Mathematics (STEM) Education. *Jurnal Pendidikan IPA Indonesia, 8*(3), 193–200.

Ke, F. (2014). An implementation of design-based learning through creating educational computer games: A case study on mathematics learning during design and computing. *Computers & Education, 73*(2), 26–39.

Model curricula for 5th-9th grades of the New Ukrainian School (to be implemented in stages from 2022). https://mon.gov.ua/ua/osvita/zagalna-sere dn ya-osvita/navchalni-pro g rami/modelni-navchalni-prog rami-dlya-5-9-klasiv-nov oyi-ukrayi nskoyi-shkoli-z aprovadzhuyutsya-po etapno-z-2022-roku

Morrison, J. (2006). TIES STEM Education Monograph Series: Attributes of STEM Education. http://www.leadingpbl.org/f/Jans%20pdf%20Attributes_of_STEM_Education-1.pdf

Sengupta, P., Dickes, A., & Farris, A. (2021). *Voicing Code in STEM: A Dialogical Imagination.* The MIT Press. https://doi.org/10.7551/mitpress/1166 8.001.0001

Schmidt, W., & Houang, R. (2007). Lack of focus in the mathematics curriculum: A symptom or a cause*? In T. Loveless (Eds.), Lessons learned: What international assessments tell us about math achievement,* 65–84. https://www.semanticscholar.org/paper/Lack-of-focus-in-the-mathematics-curriculum%3A-or-Schmidt-Houang/4f0812556f98a8f5769e2d98700b7d4b18838ec5

The concept of development of natural and mathematical education (STEM-education). (2020). https://mon.gov.ua/ua/news/uryad-uhvaliv-koncepciyu-rozvitku-stem-osviti-do-2027-roku

Wang, J. (2005). Relationship Between Mathematics and Science Achievement at the 8th grade. *International Journal of Science & Math Education, 5*(1), 1–17. https://eric.ed.gov/?id=ED490042

Yakman, G. (2008). STEAM Education: an overview of creating a model of integrative education. *PATT-17 and PATT-19 Proceedings*, 335–358. https://www.iteea.org/File.aspx?id=86752&v=75ab076a