

Article

The Level of Programming Among Pupils at Primary School in the Context of Motivation and Professional Focus

Pavel Moc ^{*}, Jarmila Honzík^{ová}  and Tetjana Tomášková ^{*}

Faculty of Pedagogy, University of West Bohemia in Pilsen, 30000 Pilsen, Czech Republic; jhonziko@kmt.zcu.cz

^{*} Correspondence: pavelmoc@kmt.zcu.cz (P.M.); tomaskot@kmt.zcu.cz (T.T.)

Abstract

Currently, teaching programming in primary and secondary schools is already standard practice in many countries. Although teaching methods and tools vary, the goal remains the same: to teach students how to program, i.e., to create appropriate algorithms for solving various tasks. In our research, we focused on the influence of personal interest and career orientation as motivation for better performance in programming and algorithm design. The main objective of the research was to determine the influence of student motivation, personal preferences, and career orientation tests on programming results. The secondary objective of the research was to verify in the practical part whether elementary school students (eighth and ninth grade) are able to program an industrial machine that they will encounter at secondary vocational schools. A structured questionnaire and an unconventional device, the PLC Logo from Siemens, were used as testing tools. Research has shown that students who have the prerequisites for studying at a technical secondary school achieve better results in programming than students who do not have these prerequisites.

Keywords: primary education; algorithm design; career orientationAcademic Editor: Vasilis
Grammatikopoulos

Received: 21 July 2025

Revised: 21 August 2025

Accepted: 21 August 2025

Published: 26 August 2025

Citation: Moc, P., Honzík^{ová}, J., & Tomášková, T. (2025). The Level of Programming Among Pupils at Primary School in the Context of Motivation and Professional Focus. *Education Sciences*, 15(9), 1111. <https://doi.org/10.3390/educsci15091111>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Programming is the process of creating and writing instructions (code) that tell a computer what to do. It allows you to create software, websites, and mobile applications or control smart devices. Programming uses various programming languages such as Python, Java, C++ or JavaScript (Nouri et al., 2020) and newer ones like Scratch, Blockly, Code.org, and LEGO Education Spike (Dostál, 2018).

At primary school, it is also possible to use the LEGO Mindstorms kit (Zaharija et al., 2013) for programming. Although the production of LEGO Mindstorms was officially discontinued in 2022, there are many building kits of this type in schools. Some education systems, such as Sweden, already have coding included in their educational curricula (Vinnervik, 2023). In other educational programs, as shown by a study by the authors Century et al. (2020), this addresses the problem of using problem-oriented transdisciplinary modules (i.e., “Time4CS” modules) that combine lessons in English Language Art (ELA), Natural Sciences, and Social Studies with the CS Code.org “Fundamentals” curriculum. The issue of the ambiguity of incorporating programming into the curriculum in Chinese primary and secondary schools is addressed by Ou (2023), who link the necessity of incorporating programming into primary and secondary schools with the rapid development of the artificial intelligence era. Curriculum designers in many countries are recognizing the importance of integrating science, technology, engineering, and mathematics education, known as STEM (Science, Technology, Engineering, Mathematics), which is becoming an

important pillar of school curricula around the world. It is an integrated curriculum that aims to make students understand how science and engineering concepts are related and how they can be applied to solve problems. This approach promotes critical thinking, creativity and collaboration, which are key skills for the future. Given the STEM way of teaching, there is a great deal of support for projects where programming is a natural tool not only for creating outputs—websites, games, presentations, and automated processes—but also specifically for algorithm design, which enables the following:

- Develops logical, analytical, and systematic thinking;
- Prepares children for future work with technology;
- Strengthens the ability to plan, predict, and test different solutions with the help of technology.

Using technology, programming and working with data are essential skills for today's digital world. However, for pupils to be interested in working meaningfully in integrated learning, it is necessary to use motivation that does not feel complex or abstract. In the primary school this includes (processed according to [Dostál, 2009, 2018](#); [Krotký, 2009](#); [Moc, 2023](#)) the following:

- Using newer block programming tools such as Scratch, Blockly, Code.org, and LEGO Education Spike.
- Working with robots and real objects, e.g., Ozobots, Bee-Bots, and LEGO robots, pupils see the result of their algorithm in motion; industrial solutions and home automation systems such as LOGO Siemens, Loxone, and others can also be used.
- Gamification and competitions—using games as motivation (e.g., “program a game”, “create an escape game”) and organizing small challenges or hackathons (teamwork with a predetermined goal).
- Connection to the real world—one can show how programming is used in everyday life, such as smart homes, apps, transport, and entertainment. Through this, children better understand the meaning—and when they see “why”, they want to know “how”.
- Supporting each individual and giving positive feedback—schools should appreciate even small successes and create a safe environment where a mistake is not a failure but a step forward, because not everyone will be a programmer, but everyone can be successful and achieve the desired result at their own pace.
- Letting kids present their projects to classmates or parents—this gives meaning and the motivation to finish things.

STEM education is important because it develops students' ability to think logically, solve problems and be creative. It connects science, technology, engineering and math into meaningful units that reflect the real world. It helps children understand connections and prepares them for life and work in a modern, technology-driven society. It promotes collaboration, communication and digital literacy, which are key to the future. Through STEM education, pupils learn not only theoretical knowledge but also how to apply it in practice, as it is clear that the Industry 5.0 generation will not only operate machines but also program and interact with them ([Berg, 2022](#)). Industry 5.0 intends to capture the value of innovative digital technologies and their human–machine interactions. In education for the Industry 5.0 era, it is necessary to focus on creative thinking ([Sosna et al., 2025](#)), which is precisely what programming and algorithm design enable, and also focus on incorporating collaboration between technologies, students, and teachers. This may not just be in terms of production equipment but also in ordinary households, where we are already partly surrounded by appliances that need to be communicated with, commanded, and programmed in some way. And this programming will depend on precise human inputs. Therefore, in our research we focused on the ability of primary school students

to successfully program and work with algorithmic design. The basic question for the research presented here was whether a student's interest in a technical field influences their results in programming the Siemens PLC LOGO.

2. The Essence of Programming in Education

The key element of programming is algorithm design, i.e., the logical compilation of steps to achieve the desired result. Coding develops logical thinking, creativity and problem-solving skills, which are useful not only in IT but also in other areas of life ([Åkerfeldt et al., 2018](#)).

Therefore, there is no doubt about the need to guide students to understand the principles of programming and algorithmic design of various processes today. Teaching programming and algorithms in primary school is important for several reasons, which are given in previous studies ([Ou, 2023](#); [Zaharija et al., 2013](#); [Moc, 2023](#); [Medeiros et al., 2019](#)). We summarize these reasons in a few points below:

- It creates the foundations of digital literacy. In a world full of technology, it is important for children to understand the principles via which modern technology works.
- Preparing for future careers—IT skills are increasingly in demand in the job market, and programming can be a key advantage for future careers.
- Understanding technological processes—algorithm design teaches students to break down complex problems into smaller parts and look for systematic solutions, which is useful even outside of computer science.
- Interdisciplinary use—programming can be combined with mathematics, science, music, or art, thus strengthening the integration of knowledge and its practical application.
- Automation and efficiency—understanding algorithms helps learners better understand how routine processes can be automated and work made more efficient.
- Developing logical thinking—coding helps children learn to think analytically, solve problems step-by-step, and find effective solutions.
- Encouraging creativity—pupils can create their own projects, games, or animations, which develops their creativity and innovative thinking skills.
- Encouraging teamwork—coding projects often require collaboration, which helps children develop communication and teamwork skills.
- Developing patience and perseverance—writing code and debugging bugs teaches students not to give up at the first failure and to look for ways to solve problems.
- Building confidence—successfully creating your own program or animation gives children a sense of accomplishment and boosts their confidence in their own abilities.

The last five reasons are very important reasons to teach programming and algorithm design in primary schools ([Markovič, 2012](#); [Kaufman & Beghetto, 2009](#); [Kalhous & Obst, 2009](#)).

Teaching programming and algorithm design is not just about code but about the overall development of thinking and skills that are key to life in the digital age of the Industry 5.0 era ([Ivanov, 2022](#)).

Teaching programming and algorithm design plays a key role in preparing students for life in the Industry 5.0 era, which emphasizes the connection between technology and humans, sustainability and creativity. To be prepared for this era, students should be able to work with technology in a way that supports their activities. They should learn to work with artificial intelligence, learn to program and understand the use of machines in line with more sustainable and resilient production systems. The connection between technology and the human factor can take place on various levels, the most important of which are outlined below ([Adolfsson & Alvunger, 2018](#); [Hargreaves, 2005](#); [Moc, 2023](#)).

Better collaboration between humans and technology: Industry 5.0 focuses on humanizing technology, which means that humans will work in symbiosis with artificial intelligence and robots. Programming skills will give pupils a better understanding of how these systems work.

Development of critical and logical thinking: Algorithm design teaches students to analyze problems, look for effective solutions and systematically approach decision making, which is essential for working in automated environments.

Personalization of technology: In Industry 5.0, technology will be tailored to the needs of individuals. Programming and algorithm design give students the opportunity to actively create software and solutions that meet their individual needs.

Innovation and creativity: While Industry 4.0 was more about automation and efficiency, Industry 5.0 supports innovative thinking and the ability to create new technologies—this is where programming plays a big role.

Sustainability and process optimization: Algorithm design that optimizes energy consumption, reduces waste, and contributes to environmentally responsible solutions should be prioritized, which is a key idea of Industry 5.0.

Readying this generation for the job market of the future: The ability to program and understand algorithms will become increasingly important for various professions, not only in IT but also in areas such as biotechnology, healthcare, automation, and sustainable design.

Promoting interdisciplinary thinking: Industry 5.0 combines technology with humanities, ethics, and creativity. Programming and algorithm design teach students how to combine technical and humanities skills to solve real problems.

Safety and ethics of technology: With the increasing influence of artificial intelligence and automation, it is important for learners to understand the principles of cybersecurity and the ethical aspects of using technology (Moc, 2023).

It follows from the above that teaching programming and algorithm design in primary schools is not only preparation for a career in IT but a key element of education for a future where people and technology will work closely together. Thanks to these skills, pupils will be able to actively influence the development of the world in the era of Industry 5.0 (Croft, 2024; Balanskat & Engelhardt, 2015).

3. Research Objectives and Research Assumptions

There is no doubt about the importance of the ability to program and algorithmize in the era of Industry 5.0 (Redakce časopisu, 2023). The factors that affect success in programming can vary, depending on the circumstances and the level of education, but also on the content of the subject taught, in which programming is included. Programming and algorithm design are taught in various subjects at primary schools, most often in Computer Science and Technical Education. We were interested in whether interest in these subjects, and therefore in other technical fields, can affect the level of programming and algorithm design (Moc, 2023).

At the beginning of the research, the basic research question was set:

Does the interest in technical fields affect the level of achievement in programming?

For the purposes of the research, several assumptions were set, the verification of which was also the focus of the research described here.

3.1. Required Requirements

- Premise 1: Pupils in the eighth and ninth grades of primary school will be able to use basic logical functions and solve assigned tasks; for time functions and counters, they will need the help of a teacher to solve them.

- Premise 2: Pupils in the eighth and ninth grades of primary school who have a personal interest in technical studies at secondary school will achieve greater success in solving tasks with timers and counters than pupils who are not interested in technical studies.
- Premise 3: Pupils in the eighth and ninth grades of primary school who will have a professional orientation in the technical and electrical engineering fields, according to the B-I-T II Professional Interests Test, will be able to solve all assigned tasks (Moc, 2023).

3.2. Research Methodology

The following research methods were used in the research, which were used to verify the research assumptions:

Questionnaire survey—questionnaire to find out the respondents' attitude towards technical education (standardized B-I-T II Professional Interests Test (Irle et al., 2004));

Experiment—PLC Logo logic machine from Siemens (2003);

Statistical data processing—Likert scale, data statistics.

3.3. Research Tools in Detail

The following were used as research tools:

The B-I-T II Professional Interests Test (Irle et al., 2004), a simple questionnaire where pupils use a Likert scale to record their relationship to the subjects of Informatics and Technical Education. It is a standardized test used by school counselors in primary schools for the purpose of choosing the future profession of pupils.

Content of the monitored areas of interest:

Tř—technical crafts:

- Installation of machines in factories;
- Brown coal mining in an open-pit mine;
- Crankshaft turning for cars;
- Repair of diesel engines;
- Welding of bicycle frames;
- Mounting car headlights;
- Forging tools;
- Laying gas pipelines;
- Yacht construction.

TP—technical and natural sciences:

- Usability testing of plastics;
- Construction of television cameras;
- Measuring the radiation from a new star;
- Work in an electrical laboratory;
- Measurement of lightning electrical voltage;
- Developing new plastics;
- Rocket flight path calculations;
- Calculations of load-bearing capacity of iron structures;
- Work at the Meteorological Institute (Irle et al., 2004).

The set of tests itself offers two versions, AA + AB and BA + BB. The first is a forced-choice test, which, according to the authors, is more time-consuming but has a more accurate explanatory power. In individual questions, the respondent is always forced to answer multiple choices with a positive or negative answer. They always choose between several variants of activities that represent the content of selected areas of professional ac-

tivities. The respondent gradually goes through 81 questions where individual professional activities are intertwined. The respondent compares one profession with others more than once. The individual questions are placed on the field as a “chessboard”. The questions must be skipped both vertically and horizontally. It is very difficult for the respondent to read the questions in advance and try to deliberately influence the result.

In the case of the second version, BA + BB, it is free-choice. In this case, the test also contains 81 questions, which are sorted into columns, and the respondent goes through them one by one. For each question, they only answer on a scale of 1–5 how much they would like to perform a particular activity. Questions from individual areas are shuffled to exclude tendentious influence on the respondent’s answers. Nevertheless, the test does not prevent the questions from being read in advance with the subsequent possibility of intentionally influencing the result (Irle et al., 2004).

The PLC Logo logic controller from Siemens (2003) is a small logic controller that is now used for less extensive industrial applications (Figure 1). The programming itself is solved in an object-oriented manner using logic functions and special functions represented by timers, counters, and memory relays (Figure 2) (Moc & Soukup, 2024).



Figure 1. Siemens logo (Moc, 2023).

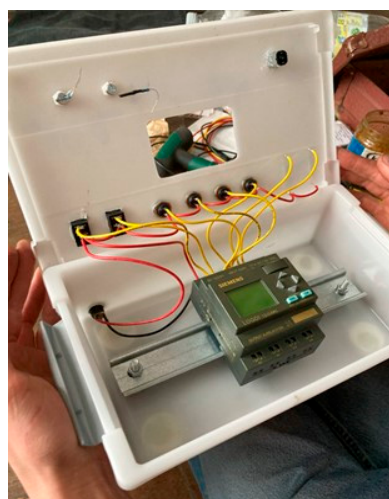


Figure 2. Siemens logo involvement (Moc, 2023).

3.4. Respondents

Pupils leaving primary school are faced with the decision of how to continue their studies and what direction to take in life. For this age group, the further focus of their

studies and, therefore, their relationship to technology are very important. Therefore, this research is focused on this age group. Four schools in the Pilsen region (Czech Republic) were selected for the research. These were pupils who were deciding on their future profession and choosing the appropriate vocational school or secondary school. The selected pupils were at the second level of primary school (eighth and ninth grade). A total of 320 pupils were randomly selected for the research with the consent of schools and parents.

4. Implementation of Research

To what extent does the ability to solve programming tasks correspond to the cognitive abilities of pupils and their professional orientation of interest? In answering this question, it is necessary to perceive the pupil's internal motivation and their ability to solve tasks as one of the criteria. Personal interests affect the motivation of an individual, but they do not have to correspond to the real professional typology of their personality, i.e., the ability to solve assigned tasks. On the other hand, the ability to solve tasks submitted on the basis of the B-I-T II Professional Interests Test may be different. Although the internal motivation of the individual probably plays a role here, the success of solving tasks may reflect more the true professional typology of the individual. This probably means a greater success rate in solving more complex tasks. This assumption can be seen in a personal orientation towards technical fields or electrical engineering, which is associated with a deeper sense of logical reasoning and object imagination ([Moc, 2023](#)).

The actual division of what the professional interests of the pupil are was carried out on the basis of the B-I-T Professional Interests Test. On this basis, technically oriented pupils will be considered suitable samples for the purpose of answering this assumption.

The diagnostic objective of the standardized B-I-T Professional Interests Test is to determine the professional orientation of an individual; this fact cannot be assessed solely on the basis of the individual's personal interests. The examined individual may show certain interests, e.g., according to the type of literature, sports, etc. In this case, it is only a motivational factor and may not correspond to the dispositions of the individual. An individual's curiosity in certain areas may not be sufficient to determine their future professional orientation. For this purpose, the incriminating test was created. The test is not intended for lay use to prevent misuse or an inappropriate interpretation of results. The diagnosis must be performed by a responsible person with an appropriate education. In this case, it is possible for the test to be administered by a primary school teacher. The results can then be used for subsequent professional orientation, i.e., career selection. An incorrect interpretation of the results can lead to an inappropriate choice of future profession and the resulting consequences. A suitable choice of future profession is also related to the motivation and dispositions of the individual, which have an impact on the success of their studies, the achievement of better results in education, and future employment. The evaluation of the tests used in our research was carried out in cooperation with a trained psychologist ([Irle et al., 2004](#)).

The organization of the research was carried out according to a pre-prepared methodology.

Organizationally, it was necessary to divide the pupils by ten for programming purposes. The other groups had a substitute program in the form of air gun shooting and workshop activities focused on woodworking.

It was essential to carry out the research first using questionnaires. In the case of the opposite procedure, there was a risk that some pupils would be enthusiastic about programming and could thus influence the completion of the Professional Interests Test. Subsequently, an experiment focused on programming was conducted.

Originally, 16 tasks were prepared for the actual programming, but based on pre-research, 10 were selected. The tasks themselves were conceived as worksheets (Kalhous & Obst, 2009). Each student had a separate computer on which the appropriate Logo Soft Comfort V8.2 software was installed. The program itself had a Czech version. For the subsequent verification of the created program based on the assigned task, a simulator located directly in Logo Soft was used, but a substantial part was verification of the PLC Logo. Logo Soft Comfort V8.2 is a Siemens program designed for Windows computers and is used to create a program on a computer for the PLC Logo. You can conveniently create your own program based on a given task. The created program can be simulated on the computer to verify its functionality. The created and verified program can then be uploaded to PLC Logo where it can be run. The created programs can be saved on the computer for further use. At the same time, the program in PLC Logo can be uploaded back to the computer for possible modifications (Siemens, 2003).

5. Course of Research

The pupils filled in the pupil questionnaire and the Professional Interests Test anonymously. Each set of tests had only a serial number, to which the result of the achieved programming results was then assigned. All results were then recorded in an MS Excel spreadsheet. The basic idea of this study was to find an answer to whether it is possible to use an industrial PLC to teach programming at primary schools, and not specifically modified didactic solutions. Another equally important task was to find answers to the above assumptions. Overall, the ability to program is perceived in the context of other variables; in our case, this was according to the Test of Professional Interests and Relationship to the Subject of Informatics and Technical Education. A record table was used for the purposes of evaluation. All records in the table were converted to % so that all results can be compared with each other.

The set of tasks for the actual programming contained 10 tasks. Completing each task was considered to be 10%. For example, if a pupil achieved task 7, he received a level of 70%, etc. Any started and unfinished tasks were no longer evaluated in any way (Moc, 2023).

In the case of the Professional Interests Test, the test results are in the form of a %. In the case of values approaching 100%, the result is considered the maximum and vice versa. Pupils' interest in the subject of Informatics and Technical Education was evaluated by a mark; for example, 1 means 100%, 3 is 50%, etc. There are many professional specializations in the Professional Interests Test, but for our purposes, only two parts that focus on technical fields were essential. More precisely, the test indicates the following:

TR—technical crafts; TP—technical and natural science professions.

Furthermore, the Professional Interests Test contains other professional specializations that are irrelevant for this research.

Based on the obtained and statistically processed results of the experiment, it was possible to start verifying the assumptions.

5.1. Premise No. 1

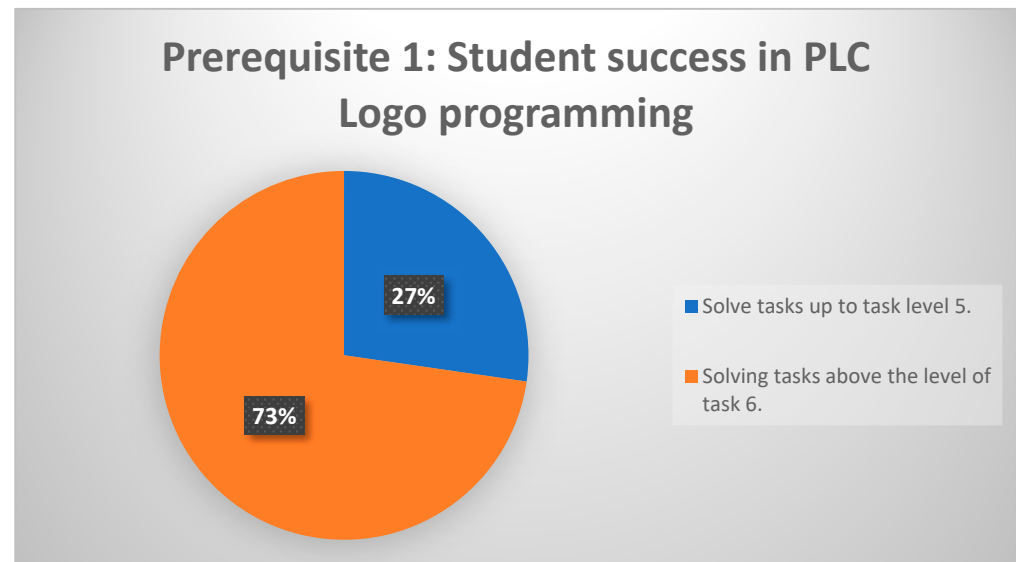
Pupils in the eighth and ninth grades of primary school will be able to use basic logical functions and solve assigned tasks, while time functions and counters will need the help of a teacher to solve them.

Verification of the Premise No. 1

For the purpose of determining whether students can solve individual tasks and successfully program PLC Logo, a limit of at least task 6 was set. Tasks up to task level 5

are only basic tasks that require basic logic to be solved using logical functions. Only after task 6 are the tasks more complex, requiring special functions to be used. In the case of task 6, the memory circuit solution is used. Follow-up tasks include other advanced functions, timers, counters, and their combinations.

The following graph shows that 73% of pupils were able to manage at least task 6 or more complex ones. Only 27% of pupils were able to solve tasks up to task level 5 (Scheme 1).



Scheme 1. Results of Premise No. 1 (Moc, 2023).

The result of the first assumption is that almost three-quarters of pupils can program advanced functions; i.e., they can be successfully used in primary school for programming and industrial solutions (Moc, 2023).

5.2. Premise No. 2

Pupils in the eighth and ninth grades of primary school who have a personal interest in technical studies at secondary schools will achieve greater success in solving tasks with timers and counters than pupils who are not interested in technical studies.

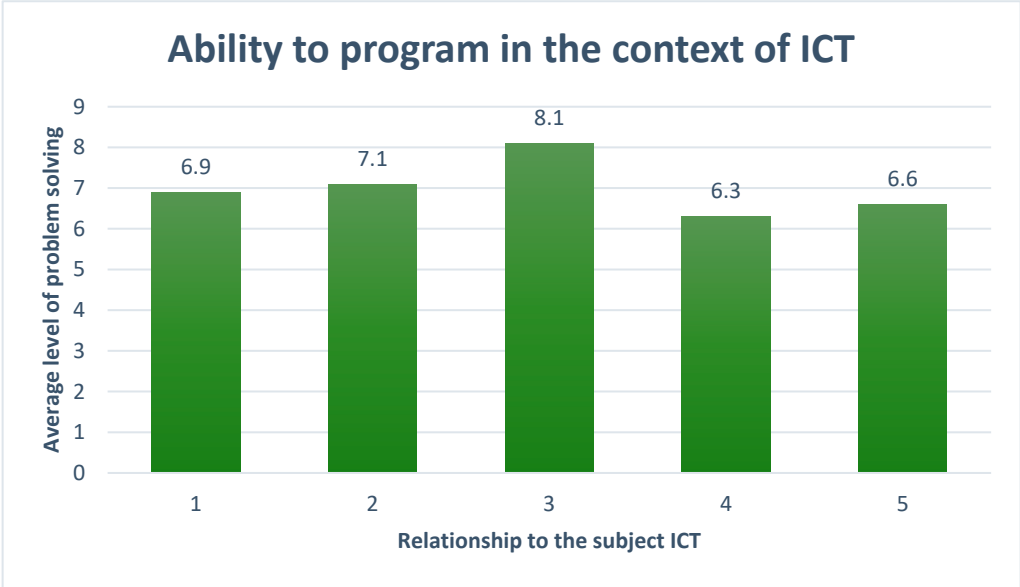
Verification of Premise No. 2

This assumption is based on the internal motivation of pupils. Pupils who are more interested in teaching Informatics and Technical Education, and who will also achieve better grades here, will be more successful in programming. This is a result based on the personal preferences of the students. These preferences were rated in the questionnaire with a grade of 1–5. In the case of a grade of 1, this is a very close relationship to the subject of Technical Education (excellent), while in the case of a grade of 5, it was a complete lack of interest in the subject.

In the case of the relationship to Computer Science, this assumption has not been fully proven, as can be seen from the graph in Scheme 2. Here it can be seen that average pupils achieved a level of 8.13 in solving tasks, i.e., an 81.3% success rate. In the case of pupils with an excellent relationship with Informatics, an average result of 6.95 was achieved, i.e., a 69.5% success rate. At the complete opposite end are students who rated their approach to Computer Science as the worst, achieving a result of 6.55 tasks, i.e., 65.5% (Moc, 2023).

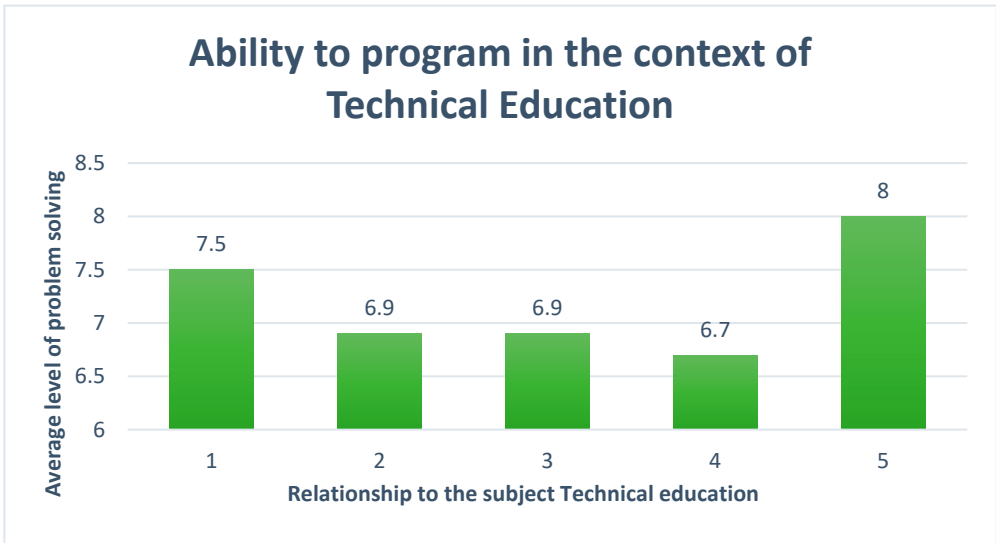
The premise implies that the best results should be achieved by pupils who perceive their relationship to Informatics as excellent. It can be stated that this assumption did not

come true, as the results in programming do not correspond to a positive attitude towards Computer Science (Scheme 2). Why students who have a good relationship with computer science do not achieve good results in programming would be a topic for further research. We might conclude that for many students, computer science (ICT) is interesting only at the user level. Therefore, programming and algorithm design should be included in the curriculum.



Scheme 2. Ability to program in the context of the subject Computer Science (Moc, 2023).

In the case of the relationship to Technical Education, the result is also evident from the following graph in Scheme 3. Here, if we ignore the result in the case of pupils with the worst relationship with Technical Education, where the success rate in programming, i.e., in achieving results, is 8 on average (80%), all other results with increasing interest in Technical Education are better. The best results were achieved by pupils with an excellent relationship with Technical Education, where they achieved an average of 7.5, i.e., 75%, in solving tasks. Subsequently, the result in the solution decreases as their interest in the corresponding object decreases (Moc, 2023).



Scheme 3. Ability to program in the context of the subject Technical Education (Moc, 2023).

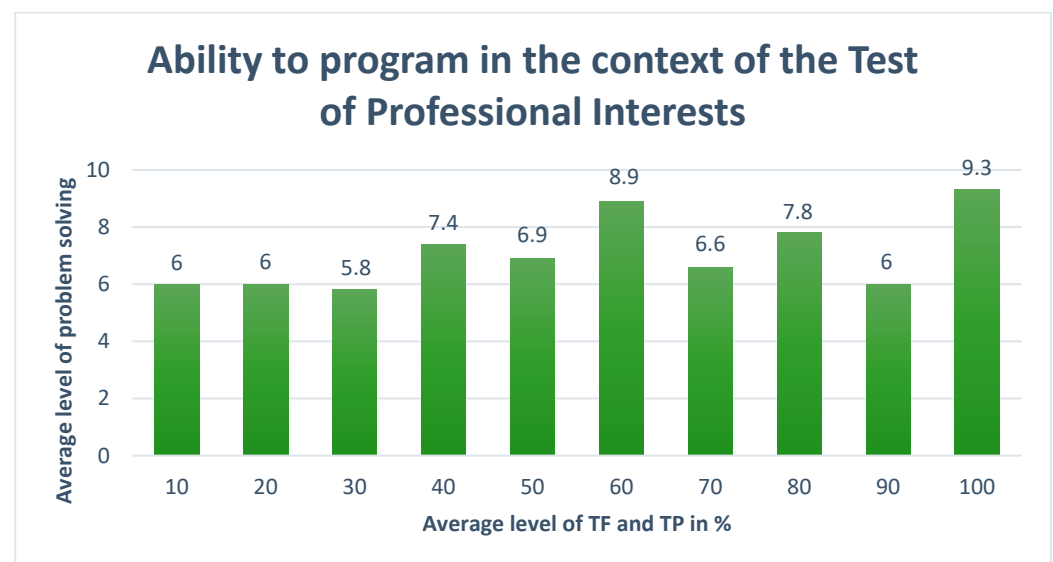
In the case of this assumption, it can be partly argued that the ability to program may be partly influenced by interest in Technical Education. Other authors have reached the same conclusions (Morris & Trushell, 2014). Unfortunately, in the context of Computer Science, this cannot be claimed. Overall, a positive attitude towards Informatics or Technical Education as a motivational factor can only be attributed a minimal influence in the context of the ability to program. It can be argued that this assumption has not been confirmed (Scheme 3).

5.3. Premise No. 3

Pupils in the eighth and ninth grades of primary school who will have a professional orientation in the technical and electrical engineering fields, according to the B-I-T II Professional Interests Test, will be able to solve all assigned tasks.

Verification of Premise No. 3

The last prerequisite is the ability to program in the context of the Professional Interests Test, which is standardized and used in primary schools as an advisory for the choice of future education at secondary school. The graph in Scheme 4 compares the results of the test, but only the results that relate to the technical focus are taken into account: TR—technical crafts and TP—technical and natural science professions. From the TR and TP, the arithmetic mean was created, which is shown horizontally in the table. For simplicity, pupils are divided into ten categories according to their interests, based on a figure of 10%. The vertical column again shows the ability to solve (program) the assigned tasks. The right-hand column shows pupils whose aptitude test results indicated interest in the future study of technically oriented fields, while the leftmost column shows pupils who had no prerequisites for further technical studies (Scheme 4).



Scheme 4. Ability to program in the context of the Professional Interests Test in the field of Tr and TP (Moc, 2023).

It is clear from the results of the graph (Scheme 4) that, although the results are not exactly linear, the ability to successfully program increases with the increasing prerequisite for technical study (Professional Interests Test). It can, thus, be argued that this assumption has been verified to be valid. Pupils who have a disposition to study at a technical secondary school have better results in programming.

5.4. Research Limits

The main limitation in the selection of the sample of respondents was the reluctance of schools to participate in this research.

The results cannot be considered generalizable, as several independent variables may have influenced the level of programming ability during the research. One of them may have been the initial entry level of knowledge and skills of pupils from different schools, as technical subjects were taught by different teachers.

6. Discussion and Conclusions

The indicated number of respondents of 320 is sufficient to answer the assumptions, although a broader survey would be needed to generalize. A possible extension to a larger number of respondents would probably enable us to answer the second assumption more accurately. At the same time, it was proven that pupils' ability to program at primary school is at a high level. In the literature, this issue of programming control units with a connection to computer-controlled machines has already been addressed ([English et al., 2013](#)). Rather than determining what devices should be used for programming in education, researchers are looking for ways to support them. This is not only about financial resources but, above all, about supporting one's own teaching. This support focuses directly on teachers and their knowledge, skills, motivation, etc. Due to the approaching Industry 5.0, it is necessary to move from the prevailing theoretical teaching approach of programming to a more practical teaching solution ([Crick, 2017](#)).

Our own research combined with the practical experiment confirmed that, in technical education, it is possible to use industrial equipment for programming and algorithm design, even in the subject of Informatics in primary schools. The same conclusion was reached by [Wing \(2006\)](#). In that study, most respondents were able to solve at least one more advanced task. However, the relationship between pupils' interest in the subjects of Informatics and Technical Education was not confirmed; here, the results are rather random and do not correspond to the assumption. In the case of the Professional Interests Test, it can be stated that if a pupil has a technical disposition, they also have better results in programming.

Due to the demonstration of the possibility of programming industrial equipment at primary schools, it is possible to recommend the inclusion of similar devices in teaching. These do not necessarily have to be a PLC, but elements of some programmable home appliances, intelligent electrical installations, etc., can be used. There is considerable room for preparing pupils for future life. In the case of special technical solutions, it is more the role of secondary schools. Dohnal already came to similar conclusions in 2009, arguing that in primary school, information about programming should be presented gradually and in logical units, with tasks being appropriate. [Panek \(2024\)](#) writes about the importance of programming in STEM and mentions the importance of online programming environments. His research concludes that students' attitudes towards programming in primary school are very positive and that all students should have the opportunity to learn programming in primary school. [Patel \(2024\)](#) argues the same in his article and shows that students who learn programming in primary school are more likely to continue their education through STEM. He believes that as technology grows in all sectors, the demand for STEM skills will continue to increase. However, this means that it is not necessary for students to show too much engagement in only technology education in primary school because STEM subjects are interrelated. The same view is shared by [Panek \(2024\)](#), who emphasizes the importance of programming in primary schools because the integration of each new technology into everyday life requires the adaptation of programming and coding. This idea already appeared in Krotký's work in 2009 ([Krotký, 2009](#)).

Engaging students in programming does not necessarily mean that they will all be programmers. The ability to algorithmize manifests itself in practice as the ability to independently solve tasks that pose demands in practice. This can include different working procedures, searching for a suitable workflow, etc. (Finger & Houguet, 2009). In this context, it is worth mentioning the results of research that presents a systematic approach to designing products regarding their environmental impact throughout their entire life cycle (Tomášková et al., 2024). The concept of life cycle assessment (LCA) and ecodesign can be effectively integrated into the teaching of technical subjects and programming tasks, allowing students to understand not only the creation of algorithm design but also the impacts of technological decisions on sustainability. This interdisciplinary approach supports the development of critical thinking, creativity, and responsibility, which are key competencies for the era of Industry 5.0. These are also not only important for working with artificial intelligence but also for robotics and automation.

Author Contributions: Conceptualization, P.M., J.H.; methodology, P.M., J.H.; resources, P.M., T.T., writing—original draft preparation, P.M., J.H. and T.T., writing—review and editing, P.M., J.H. and T.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded within the budget of the Faculty of Pedagogy; no special funds were allocated.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and approved by the Ethics Committee of Faculty of Pedagogy, University of West Bohemia in Pilsen (30 January 2023).

Informed Consent Statement: Written consent to participate in the research was obtained from the students or their legal representatives.

Data Availability Statement: The original contributions presented in the study are included in the article; further inquiries can be directed to the corresponding authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Adolfsson, C.-H., & Alvunger, D. (2018). The selection of content and knowledge conceptions in the teaching of curriculum standards in compulsory schooling. In C. Sundberg, & D. Alvunger (Eds.), *Transnational curriculum standards and classroom practices: The new meaning of teaching* (pp. 98–115). Springer.
- Åkerfeldt, A., Kjällander, S., & Selander, S. (2018). *Programmering: Introduktion till digital kompetens i grundskolan* [Programming: Introduction to digital competence in primary school] (Första uppl.). Liber.
- Balanskat, A., & Engelhardt, K. (2015). *Computing our future: Computer programming and coding—Priorities, school curricula and initiatives across Europe*. European Schoolnet. Available online: https://www.dzs.cz/sites/default/files/2020-08/Computing%20our%20future_final.pdf (accessed on 7 July 2024).
- Berg, C. (2022). *What is industry 5.0? (In-depth guide with examples)*. Clarify.io. Available online: <https://www.clarify.io/learn/industry-5-0> (accessed on 31 March 2025).
- Century, J., Ferris, K. A., & Zuo, H. (2020). Finding time for computer science in the elementary school day: A quasi-experimental study of a transdisciplinary problem-based learning approach. *International Journal of STEM Education*, 7, 20. [CrossRef]
- Crick, T. (2017). *Computing education: An overview of research in the field* (pp. 1–38). Cronfa-Swansea University. Open Access Repository. Available online: <http://cronfa.swan.ac.uk/Record/cronfa43589> (accessed on 7 January 2025).
- Croft, D. (2024). *Guide: Industry 5.0—Learn lean sigma*. Available online: <https://www.learnleansigma.com/guides/industry-5-0/> (accessed on 7 July 2024).
- Dostál, J. (2009). *Programování na základních školách* [Programming in elementary schools] [Qualification thesis, Masarykova Univerzita].
- Dostál, J. (2018). *Člověk a technika—podkladová studie k revizím RVP* [Humans and technology—Background study for RVP revisions]. Podkladová Studie. NVU.
- English, L. D., Hudson, P., & Dawes, L. (2013). Engineering-based problem solving in the middle school: Design and construction with simple machines. *Journal of Pre-College Engineering Education Research (J-PEER)*, 3(2), 5. [CrossRef]

- Finger, G., & Houguet, B. (2009). Insights into the intrinsic and extrinsic challenges for implementing technology education: Case studies of Queensland teachers. *International Journal of Technology and Design Education*, 19(3), 309–334. [CrossRef]
- Hargreaves, A. (2005). Educational change takes ages: Life, career and generational factors in teachers' emotional responses to educational change. *Teaching and Teacher Education*, 21(8), 967–983. [CrossRef]
- Irle, M., Wolfgang, A., & Muhic, J. (2004). *Test profesních zájmů B-I-T II* [Test of professional interests B-I-T II]. Testcentrum.
- Ivanov, D. (2022). The Industry 5.0 framework: Viability-based integration of the resilience, sustainability, and human-centricity perspectives. *International Journal of Production Research*, 61(5), 1683–1695. [CrossRef]
- Kalhous, Z., & Obst, Z. (2009). *Školní didaktika* [School didactics]. Portál.
- Kaufman, J. C., & Beghetto, R. A. (2009). Beyond big and little: The Four C model of creativity. *Review of General Psychology*, 13(1), 1–12. [CrossRef]
- Krotký, J. (2009). Information technologies and technical education. In M. Chraska, & M. Havelka (Eds.), *Trends in education* (Vols. 1 & 2, pp. 472–475). Upol.
- Marković, D. (2012). Creative education and new learning as means of encouraging creativity, original thinking and entrepreneurship. In Humanities and the contemporary world (*International Conference proceedings*). Available online: <https://www.semanticscholar.org/paper/Creative-Education-and-New-Learning-as-Means-of-%2C-Markovic/f22041355a870bf25fff31cf149d54383021b2d3> (accessed on 7 January 2025).
- Medeiros, R. P., Ramalho, G. L., & Falcao, T. P. (2019). A systematic literature review on teaching and learning introductory programming in higher education. *IEEE Transactions on Education*, 62(2), 77–90. [CrossRef]
- Moc, P. (2023). *Podpora technického vzdělání pomocí počítačem řízených strojů* [Support for technical education using computer-controlled machines] [Doctoral dissertation, Západočeská Univerzita v Plzni].
- Moc, P., & Soukup, V. (2024). Manipulační automat v primárním vzdělávání. In *Sborník abstraktů z mezinárodní studentské odborné konference Olympiáda techniky Plzeň 2024* [Proceedings of the International Student Technical Conference, Pilsen 2024] (p. 15). Západočeská Univerzita v Plzni.
- Morris, D., & Trushell, J. (2014). Computer programming, ICT and gender in the classroom: A male-dominated domain or a female preserve? *Research in Teacher Education*, 4(1), 4–9.
- Nouri, J., Zhang, L., Mannila, L., & Norén, E. (2020). Development of computational thinking, digital competence and 21st century skills when learning programming in K-9. *Education Inquiry*, 11(1), 1–17. [CrossRef]
- Ou, O. (2023). Investigation and analysis of the current situation of programming education in primary and secondary schools. *Heliyon*, 9(4), e15530. [CrossRef] [PubMed]
- Panek, S. (2024). *Benefits of coding in the elementary classroom* [Master's thesis, Concordia University]. Available online: https://digitalcommons.csp.edu/teacher-education_masters/106 (accessed on 20 January 2025).
- Patel, L. (2024, April 17). *Importance of coding education in elementary schools*. Tynker. Available online: <https://www.tynker.com/blog/importance-of-coding-education-in-elementary-schools/> (accessed on 20 January 2025).
- Redakce časopisu. (2023). *Co přinese pátá průmyslová revoluce—Průmysl 5.0?* [What will the fifth industrial revolution bring—Industry 5.0?]. ElektroPrůmysl.cz. Available online: <https://www.elektroprumysl.cz/technologicke-novinky/co-prinese-pata-prumyslova-revoluce-prumysl-5-0> (accessed on 30 July 2024).
- Siemens. (2003). *Logo!: Manual edition* (6th ed., p. 295). Available online: https://cache.industry.siemens.com/dl/files/461/16527461/att_82564/v1/Logo_e.pdf (accessed on 3 August 2023).
- Sosna, T., Vochozka, V., Šerý, M., & Blažek, J. (2025). Developing pupils' creativity through 3D modelling: An experimental study. *Frontiers in Education Secondary Teacher Education*, 10, 1583877. [CrossRef]
- Tomášková, T., Duda, D., Krotký, J., & Honzíková, J. (2024). Ecodesign as a new possibility in teaching technical subjects in secondary schools. *Education and New Developments*, 199–203. [CrossRef]
- Vinnervik, P. (2023). Programming in school technology education: The shaping of a new subject content. *International Journal of Technology and Design Education*, 33, 1449–1470. [CrossRef]
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35. [CrossRef]
- Zaharija, G., Mladenović, S., & Boljat, I. (2013). Introducing basic programming concepts to elementary school children. *Procedia-Social and Behavioral Sciences*, 106, 1576–1584. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.