Programming in grade 7-9: Action possibilities and constraints from the perspective of mathematics and technology teachers

Niklas Humble

Main supervisor: Lena-Maria Öberg

Co-supervisors: Peter Mozelius & Stefan Hrastinski

Faculty of Science, Technology and Media

Thesis for Doctoral Degree in Computer and System Science

Mid Sweden University

Östersund, 2023-01-19

Akademisk avhandling som med tillstånd av Mittuniversitetet i Östersund framläggs till offentlig granskning för avläggande av filosofie doktorsexamen onsdag, 19/1–2023, 10.00, F234, Mittuniversitetet Östersund. Seminariet kommer att hållas på engelska.

Programming in grade 7-9: Action possibilities and constraints from the perspective of mathematics and technology teachers

Printed by Mid Sweden University, Sundsvall

ISSN: 1652-893X

ISBN: 978-91-89341-85-2

Faculty of Science, Technology and Media
Mid Sweden University, SE-831 25 Östersund, Sweden
Phone: +46 (0)10 142 80 00

Mid Sweden University Doctoral Thesis 382

To the teachers.

Acknowledgement

This thesis has only one author, but there are many who deserve acknowledgement for making it a reality: Mid Sweden University and the Department of Computer and System Science, for believing in me and giving me this opportunity; colleagues at the Forum for Digitalisation, for insightful discussions and invaluable feedback on work in progress; fellow PhD students and researchers in the research collaboration and graduate school GRADE, for great doctoral courses and providing a sense of community; Dan Remenyi and the people at Academic Conferences International, for interesting conversations, seminars and conferences that helped me develop as a researcher; Marcelo Milrad, for comments and suggestions for improvements at the final stages of the thesis project; my thesis supervisors, Lena-Maria Öberg, Peter Mozelius and Stefan Hrastinski, for making this journey both enjoyable and an opportunity for self-development – I could not have asked for better supervisors; and my family, for being the best! Lastly, all the teachers who have participated in our programming courses and studies, this would not have been possible without you.

Thank you!

Table of contents

Abstract

The idea of using programming in fields outside of computer science is not new, but it has received renewed attention with the global trend of integrating programming in school curricula. In the Swedish context, mathematics and technology teachers for grades 7-9 have been significantly affected by the integration of programming. Previous research has highlighted the opportunities and challenges presented by this integration and shown that programming is a relevant skill for students to acquire, but it also shows that teachers often lack knowledge, skill and guidance for this integration. The knowledge object of the thesis is to develop knowledge about teachers' perceived affordances (action possibilities) and constraints in using programming in grade 7-9 mathematics and technology. The thesis includes five papers and uses a qualitative approach to identify action possibilities and constraints of using programming in grade 7-9 mathematics and technology. These action possibilities and constraints relate to three aspects of teaching and learning: subject content, motivation and engagement, and digital competence. Together, the findings provide a conceptual model for what programming can be used for in grade 7-9 mathematics and technology. This conceptual model, which incorporates action possibilities and constraints of programming related to the three aspects of teaching and learning mentioned above, represents the contribution of the thesis. The thesis provides new insights into the understanding of teachers' use and perceptions of programming for teaching and learning in grade 7-9 mathematics and technology (the study object). The thesis also has practical implications for the future design of professional development courses on programming for teachers and how programming should be used and integrated in an educational context.

Summary in Swedish

Idén att använda programmering i fler områden än datavetenskap är inte ny, men det har aktualiserats igen med den globala trenden att integrera programmering i skolans läroplan. I Sverige har årskurs 7–9 lärare i matematik och teknik varit särskild berörda av integreringen av programmering. Tidigare forskning visar både möjligheter och utmaningar, till exempel att programmering är en relevant färdighet men att lärare ofta saknar kunskap, kompetens och vägledning i att genomföra integreringen. Med affordances som teoretisk lins har kunskapsobjektet i avhandlingen varit att utveckla kunskap om lärares upplevda handlingsmöjligheter och begränsningar i att använda programmering för matematik och teknik i årskurs 7–9. Med kvalitativ ansats och fem inkluderade studier har avhandlingen identifierat flera handlingsmöjligheter och begränsningar med programmering för matematik och teknik i årskurs 7–9. Dessa relaterar till tre aspekter av undervisningen: ämnesinnehåll, motivation och engagemang, samt digital kompetens. Tillsammans bildar detta en konceptuell modell över vad programmering kan användas till i årskurs 7–9 matematik och teknik. Den konceptuella modellen, som inbegriper handlingsmöjligheter och begränsningar med programmering relaterat till tre aspekter av undervisning, utgör avhandlingens bidrag. Avhandlingen bidrar med nya insikter om lärares användning och upplevelse av programmering för matematik- och teknikundervisning i årskurs 7–9 (studieobjektet). Avhandlingen har även praktiska implikationer för design av fortbildningskurser i programmering, samt programmeringens användning och integration i skolan.

List of papers

The work and findings presented in this thesis are based on five studies. These studies (hereby referred to as Studies I-V) have been presented in five separate research papers.

Study I: Humble, N., Mozelius, P., & Sällvin, L. (2020). Remaking and reinforcing mathematics and technology with programming – Teacher perceptions of challenges, opportunities and tools in K-12 settings. *The International Journal of Information and Learning Technology*, *37(*5), 309-321. https://doi.org/10.1108/IJILT-02-2020-0021

Study II: Humble, N. (2021). The use of Programming Tools in Teaching and Learning Material by K-12 Teachers. In *Proceedings of the 20th European Conference on E-Learning (ECEL 2021)* (pp. 574-582). Academic Conferences and Publishing International Limited.

Study III: Humble, N. (2022). Teacher observations of programming affordances for K-12 mathematics and technology. *Education and Information Technologies, 27*(4), 4887-4904. https://doi.org/10.1007/s10639-021-10811-w

Study IV: Humble, N. (in third revision). A Conceptual Model of What Programming Affords Secondary School Courses in Mathematics and Technology.

Study V: Humble, N., & Mozelius, P. (2022). Making programming part of teachers' everyday life – Programming affordances and constraints for K-12 mathematics and technology. *The International Journal of Information and Learning Technology*, Vol. ahead-of-print No. ahead-of-print. https://doi.org/10.1108/IJILT-03-2022-0069

1 Introduction

> My plans for the lesson went out the window after 20 minutes, but the classroom sparked with creativity. I must admit that it was uncomfortable to lose control, but at the same time it was fun to see the enthusiasm in the classroom that was maintained throughout the lesson.
>
> (Teacher in Study I)

The history of programming can be traced back to Ada Lovelace, the world's first programmer and daughter of the poet Lord Byron, in the 19th century (Aiello, 2016). In the 1840s, Lovelace wrote a series of notes in collaboration with Charles Babbage as part of work on the Analytical Engine that are recognisable to us today as the first computer program (Aiello, 2016). The idea was to build a machine to do calculus, but Lovelace also discussed the possibilities of using machines for other applications, such as writing music and literature (Aiello, 2016).

Looking back at the history of computing in education, it is clear that the idea of integrating programming in mathematics and engineering is not a new one (Lee, 2004). In the 1950s, the computer was seen as an extension of the calculator, and mathematicians and engineers were retrained to solve problems using the programming language of computers (Lee, 2004). A similar occurrence is now being seen with the ongoing global trend of integrating programming in school curricula from kindergarten to grade 12 (K-12) (Floyd, 2019; Balanskat & Engelhardt, 2015; Rubio et al., 2015). A common argument made in support of this integration is that it will help develop students' digital competence (Nouri et al., 2020; Balanskat & Engelhardt, 2015). Digital competence can be defined as the ability to use information and communications technology (ICT) critically, confidently and creatively for work, leisure, learning or social participation to achieve individual goals (Ferrari & Punie, 2013). Digital competence is further considered to be transversal, meaning that it enables the development of other competences (Ferrari & Punie, 2013).

Further arguments given are the shortage of computing and informatics professionals on the labour market and that coding skills could help students navigate an increasingly digitalised society (Admiraal et al., 2019; Enders et al., 2019; Balanskat & Engelhardt, 2015). The integration of programming is also expected to support students to develop other useful skills, such as computational thinking, logical thinking, reasoning, self-efficacy, problem solving and creativity (Zhang & Nouri, 2019; Psycharis & Kallia, 2017; Balanskat & Engelhardt, 2015).

The integration of programming in K-12 education can be conducted in various ways (Bråting & Kilhamn, 2021). In England, the newly introduced subject of Computing includes programming, and in

Sweden and Finland, programming has been integrated both cross-curricula and in specific subjects (Bråting & Kilhamn, 2021; Bocconi, Chioccariello & Earp, 2018). In Sweden, the integration of programming was initiated with a new curriculum approved by the Swedish government in 2017 (Heintz et al., 2017). The new curriculum further specifies that programming should be included in the subjects of mathematics and technology (Heintz et al., 2017).

However, concerns have been raised about the ongoing integration of programming in K-12 education in many countries. It has been highlighted that teachers need guidance and support in how to use programming for teaching and learning (Pörn, Hemmi, & Kallio-Kujala, 2021; Szabo et al., 2019; Webb et al., 2017; Royal Society, 2017). In the UK, the integration of programming received criticism from the Royal Society (2017), which referred to it as "patchy and fragile" and stated the need for better teacher support and professional development. In Sweden, many teachers do not have the necessary knowledge and skills in programming to realise its potential and avoid pitfalls in creating teaching and learning activities that facilitate desired learning objectives (Bråting, Kilhamn & Rolandsson, 2020). The overall objective of this thesis is to address the issue that teachers lack guidance, knowledge and skills to see the potentials and possible pitfalls of programming for teaching and learning.

In the Swedish context, mathematics and technology teachers for grades 7-9 have been directly affected by the integration, with programming and related concepts explicitly mentioned in the new curriculum (Heintz et al., 2017). This was important for choosing the study object for the thesis. According to Askling (2004), a study object can be understood as the focus of research. The study object of this thesis is teachers' use and perceptions of programming for teaching

and learning in grade 7-9 mathematics and technology. However, a study object can be examined and understood using a variety of different approaches, theories and concepts (Vetenskapsrådet [Swedish Research Council], 2004). How a study object is studied is known as the knowledge object (Askling, 2004). Fransson and Lundgren (2003) describe it as such: if the study object is the territory that is being studied, then the knowledge object can be considered a map. The knowledge object of this thesis is based in computer and system sciences, specifically the discipline of technology-enhanced learning (this is further described in *2 Background*).

In computer and system sciences, such as human-computer interactions and information systems, a popular theory for understanding and examining the relationship between humans and technology is that of affordances (Stendal, Thapa & Lanamäki, 2016). Affordances can highlight action possibilities and constraints that are perceived in the use of technology (Bloomfield, Latham & Vurdubakis, 2010). The theory of affordances (this is further described in *2.1 Theoretical lens*) has been applied in the thesis for examining the study object and to develop knowledge based in computer and system science. The knowledge object in the thesis is to develop knowledge about teachers' perceived affordances (action possibilities) and constraints in using programming in grade 7-9 mathematics and technology.

The thesis provides an understanding of the study object through the application of affordances and offers guidance to teachers and other stakeholders in the ongoing integration of programming in K-12 education. Studies I-V, included in this thesis, provide data on teachers' use and perceptions of programming for teaching and learning in grade 7-9 mathematics and technology. The thesis summarises the

findings from the included studies and expands on them in the development of a conceptual model. The conceptual model addresses what programming can be used for in grade 7-9 mathematics and technology. The conceptual model (further described in *7 Concluding remarks*) incorporates teachers' perceived affordances (action possibilities) and constraints for programming and relates these to three aspects of teaching and learning: supporting subject content, facilitating motivation and engagement, and developing digital competence.

To the author's knowledge, a conceptual model that highlights affordances (action possibilities) and constraints of programming for K-12 education has not been presented in previous research (this is further described in *7.1 Theoretical contribution*). The findings presented in this thesis can be used in the design of professional development courses in programming for teachers, for teacher education and by K-12 teachers in their teaching and learning activities with programming (this is further described in *7.2 Implications for practice*).

1.1 Aim and research questions

The aim of the thesis is to examine teachers' use and perception of programming for teaching and learning in grade 7-9 mathematics and technology. The two research questions that guided the research carried out were:

> RQ1) What action possibilities do teachers perceive in programming for teaching and learning in grade 7-9 mathematics and technology?
>
> RQ2) What constraints do teachers perceive in programming for teaching and learning in grade 7-9 mathematics and technology?

Studies I-V, included in the thesis, all relate to the aim and research questions of the thesis. However, some of the included studies have a wider scope and/or apply a different theoretical lens. Further explanations of how this affects the studies' contributions to the thesis are provided in *5 Summary of studies*. Figure 1 provides an overview of the included studies in relation to the aim and research questions of the thesis.

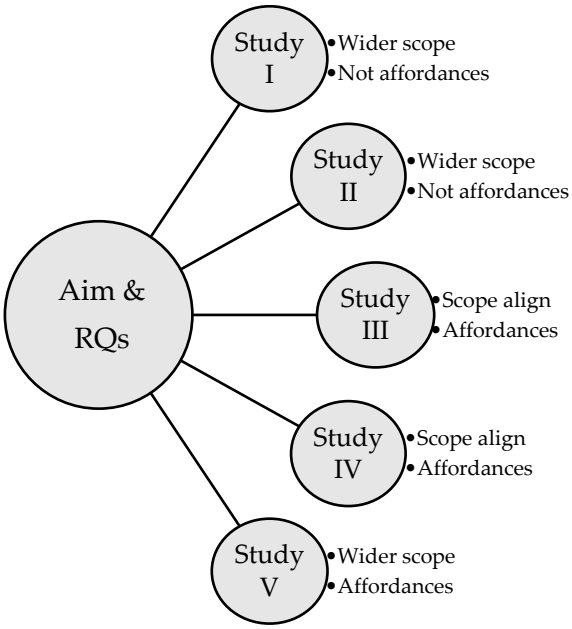

Figure 1. Overview of Studies I-V in relation to the aim and RQs of the thesis

2 Background

> "In the LOGO environment, children learn that the teacher too is a learner, and that *everyone* learns from mistakes."
>
> (Papert, 1993, p. 114)

The thesis and included studies are situated in technology-enhanced learning (TEL), which is described by Balacheff et al. (2009) as a multidisciplinary discipline. TEL, which incorporates the concept of E-learning, refers to the practice of using information and communication technologies (ICT) to enhance teaching and learning (Kirkwood & Price, 2014). TEL mainly concerns the use of computer-based technology to support learning, but historically it could also include other instructional technologies such as radio, television and film (Sen & Leong, 2020).

In computer and system sciences, such as information systems, a common focus of TEL research is higher education and organisational learning, for example how to redesign learning spaces in higher education using digital technology to accommodate students who are unable to participate on site (Kurian, John & Lang, 2021), or how novel technology, such as machine learning, can be coordinated with human learning in organisations to achieve more effective learning as a whole (Sturm et al., 2021). However, the focus of this thesis is on technology-enhanced teaching and learning for students in K-12 education, which will affect them both as students in higher education and in the labour market and is therefore considered an important focus for the research conducted in this thesis.

2.1 Theoretical lens

Several frameworks were considered for use as the theoretical lens for the thesis project. However, the ultimate choice was between computational thinking (CT) and affordances. CT was first applied by Seymour Papert in 1980 and later popularised by Jeanette Wing in 2006 (Lodi & Martini, 2021; Grover & Pea, 2013). There have been many definitions suggested for CT (Grover & Pea, 2013), which often include problem solving inspired by computer science and programming (Nouri et al., 2020; Shute, Sun & Asbell-Clarke, 2017; Yadav, Hong & Stephenson, 2016). In the early stages of the thesis project, CT was chosen as the main theoretical framework. CT has also commonly been used in related research on programming in K-12 education. However, CT was later abandoned for the thesis project in favour of the theory of affordances due to the challenges of supporting an analysis of teachers' use and perceptions of programming (this is further discussed in *3 Thesis journey*).

The theory and concept of affordances was coined by James Gibson (Stendal, Thapa & Lanamäki, 2016; Bower & Sturman, 2015) and is defined as 'action possibilities', or what an environment provides for an animal (Bower & Sturman, 2015; Gibson, 2015). The theory has been further developed and introduced in fields such as human-computer interaction and information systems, noticeably by the works of Donald Norman (Stendal, Thapa & Lanamäki, 2016). Norman (1999) argued that affordances that are of interest in the design of artefacts are mainly those perceived as possible by the user. Previous research in information systems provides examples of affordances being used both as pre-existing action possibilities in artefacts, intentionally

introduced by a designer, and as emerging action possibilities, perceived by users through their use of artefacts (Stendal, Thapa & Lanamäki, 2016). In this thesis, affordances are studied as emerging action possibilities in programming through the use and perceptions of teachers. For readability, the term 'action possibilities' is used instead of affordances in subsequent sections of the thesis. Action possibilities is a common definition of affordances that can be traced back to Gibson, who coined the concept (Stendal, Thapa & Lanamäki, 2016; Bower & Sturman, 2015; Gibson, 2015).

Two concepts related to affordances, which are often considered part of the theory of affordances, are 'constraints' and 'conceptual models'. According to Norman (1999), constraints are powerful traits of design that can guide users to meaningful actions. Norman (1999) specifies three types of constraints: physical, logical and cultural. Examples of these include the impossibility of moving a cursor outside of a computer screen (physical), knowing that scrolling down on a computer screen will show more of the page displayed (logical) and knowing that the scroll bar is situated on the right side of a computer screen (cultural) (Norman, 1999). Previous research shows that constraints can also be used to describe the ways in which technology can hinder individuals and organisations from achieving their goals (Leonardi, 2013; Majchrzak & Markus, 2012).

According to Norman, conceptual models are a simplified explanation of how something works (2013, p. 25) and often a crucial aspect of design (Norman, 1999). For example, folders displayed on a computer screen do not imply the presence of physical folders inside the hardware; they are representations for the usability of the system (Norman, 2013, p. 25). It is important to note that an artefact can have many conceptual models, and the designer's conceptual model is often

different from that of the user (Norman, 2013, p. 26). In this thesis, constraints are used to describe what teachers perceive to limit their action possibilities with programming. A conceptual model is used in *7 Concluding remarks* to offer a simplified explanation of how teachers use and perceive programming for teaching and learning in grade 7-9 mathematics and technology.

Lastly, it should be noted that there are many variations on affordances, and the theory has been applied in many different ways in previous research. Affordances can be combined with other theory, such as Heideggerian philosophy, to study the relationship between an actor and artefact (Osmundsen, Meske & Thapa, 2022). Affordances can also be used more as a concept than as a theory, for describing the possibilities presented by artefacts, such as in a study by Papavlasopoulou, Giannakos and Jaccheri (2017) on the affordances of tangible programming tools in K-12 education. In this thesis, the application of affordances is based mainly on the descriptions provided by Norman (1999; 2013) and Stendal, Thapa and Lanamäki (2016).

2.2 Related research

Research on programming in K-12 education often focuses on its application in science, technology, engineering and mathematics (STEM). An early pioneer in this field was Seymour Papert, by some considered the "father of educational computing" (Stager, 2016). In the late 1960s, Papert and his colleagues created Logo, a programming language that could be used to engage students in mathematics in a playful way through visualisation (Stager, 2016). In Logo, programming was used to control a turtle to draw shapes on a screen, enabling a concrete and purposeful understanding of geometry, calculus and algebra (Stager, 2016). This idea lives on in many modern programming tools for education, such as Scratch and the programming library Turtle for Python. In this section, a selection of related research is presented that supports the understanding of teachers' use and perceptions of programming for teaching and learning in grade 7-9 mathematics and technology. These are further discussed in relation to the findings of the thesis in *6 Discussion*. For a more thorough presentation of related research, see the included Studies I-V.

A common distinction between different programming tools used in the K-12 setting is textual programming tools versus block programming tools. This distinction is based on the method applied for creating programs, either visual blocks or text input (Szabo et al., 2019). Some examples of block programming tools are Scratch and App Inventor, whereas Logo and Python use textual programming (Szabo et al., 2019); see Figure 2. There are also examples of programming-inspired activities that do not use computers. This is sometimes

referred to as unplugged programming (Vinnervik, 2022). Programming can further be used to control external physical objects. This is sometimes referred to as tangible programming and can be conducted via, for example, a Micro:bit or Ardiuno (Cederqvist, 2022). In the thesis and included Studies I-V, teachers' use and perceptions of programming relate to textual programming, block programming, unplugged programming and tangible programming. The thesis highlights both the action possibilities and constraints of these different types of programming in grade 7-9 mathematics and technology.

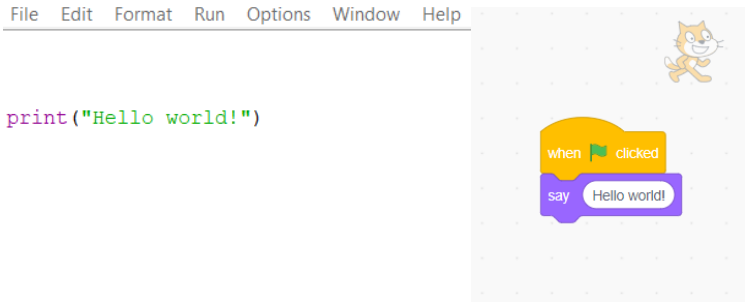

Figure 2. Example of "Hello world!" in Python and Scratch

Szabo et al. (2019) provide a global overview of initiatives to introduce programming in K-12 schools through a systematic literature review. The study shows that changes to national curricula were a driving force for many programming initiatives (Szabo et al., 2019). A common focus of the included papers in the review is to investigate and compare students' engagement, motivation and learning outcomes using different programming tools (Szabo et al., 2019). The results of the study indicate that younger students find the block programming

tool Scratch most appealing, while older students prefer tools that allowed for more advanced programming development (Szabo et al., 2019). Tangible elements of programming tools also seem to engage students (Szabo et al., 2019).

In the thesis and included Studies I-V, the focus is on teachers' use and perceptions rather than those of students. However, many of the findings presented by Szabo et al. (2019) are still relevant to discuss in relation to teachers' use and perceptions of programming, since they often include experiences of programming with students. The study by Szabo et al. (2019) further highlights that many of the included papers in the literature review report a need for supporting teachers for timetabling, workload and confidence amongst others (Szabo et al., 2019). This relates to the overall objective of the thesis, presented in *1 Introduction,* to address the issue of teachers lacking guidance, knowledge and skills to see the potentials and possible pitfalls of programming for teaching and learning.

Vinnervik (2022) reports on a study from 2018 with 19 teachers in grades 1-9 preparing for the integration of programming in Swedish schools. The study shows that teachers did not feel sufficiently prepared for the integration and expected school leadership to be more active in the process (Vinnervik, 2022). Teachers also highlighted challenges regarding IT infrastructure, ICT support and professional development in the schools (Vinnervik, 2022). However, the study also showed that teachers perceive the integration to be valid and timely due to its relevance for today's society and relevant to more subjects than just mathematics and technology (Vinnervik, 2022). Furthermore, they perceived programming to provide more variation in teaching and learning and more hands-on exploration of subject content (Vinnervik, 2022).

In a study of 20 teachers from kindergarten to grade 12, Kilhamn, Bråting and Rolandsson (2021) show that teachers perceive programming to be a powerful tool for both mathematics and other subjects (Kilhamn, Bråting & Rolandsson, 2021). Programming is perceived as fun and interesting by students and can therefore increase engagement in subject content (Kilhamn, Bråting & Rolandsson, 2021). Programming can further be used to develop skills in problem solving, through the deconstruction of problems and instructions, and for testing, debugging and modifying solutions (Kilhamn, Bråting & Rolandsson, 2021). Kilhamn, Bråting and Rolandsson (2021) relate their findings to the framework of computational thinking.

Korhonen et al. (2022) report on a 2015-2016 study that examined the perceptions, emotions and attitudes of 943 Finnish kindergarten to grade 9 teachers towards the integration of programming in the national curriculum. The study suggests that teachers perceive several advantages with the integration of programming, such as that it can be implemented across a number of subjects and that it allows teachers to look at their teaching from different perspectives (Korhonen et al., 2022). Furthermore, programming is perceived to be a relevant and useful skill that schools should equip their students with (Korhonen et al., 2022). Teachers also expressed that programming can support students to develop other useful skills, such as logical thinking, and make school subjects more interesting (Korhonen et al., 2022). However, there were also perceived challenges reported in the study. Teachers stated insufficient experience and skills in programming and a lack of clarity in integration planning over what should be taught, the tools that should be used and how much time should be allocated for programming (Korhonen et al., 2022).

Previous research has highlighted the use and perceptions of programming in K-12 education by both teachers and students, which can be related to the findings presented in this thesis and included Studies I-V (this is further discussed in *6 Discussion*). However, there is little related research that applies the theory of affordances in the study of programming in K-12 education, compared to computational thinking, which is commonly applied. Furthermore, to the author's knowledge, no previous research has presented a conceptual model that highlights the action possibilities and constraints of programming for K-12 education. The related research presented in this section is discussed and compared to the findings of the thesis in *6 Discussion*. A more thorough discussion of the findings in the included studies in relation to previous research is provided in Studies I-V. The developed conceptual model in this thesis is first presented in Study IV and further developed in *7 Concluding remarks*.

3 Thesis journey

> Because flawed code is instantly apparent from a program not working, students get instant feedback on their work without asking for support. Therefore, I believe that programming can be a good opportunity for students to exercise autonomy.
>
> (Teacher in Study V)

The thesis journey started in 2018 and the study object has remained relatively unchanged since then. It was decided early on that the thesis project should focus on programming in K-12 education, which was later refined to teachers' use and perceptions of programming for teaching and learning in grade 7-9 mathematics and technology. The author has a background as a teacher and IT coordinator and prior to starting the PhD studies worked with supporting teachers in learning and integrating programming in K-12 education. Some changes were made to the study approach, specifically the knowledge object, regarding for example, the choice of theoretical lens and which years (grades) of K-12 mathematics and technology should be included. In this section, the thesis journey is discussed in relation to the research process and contexts of Studies I-V and the changes that have been made to the thesis project over the years. An overview of the included studies is presented in Table 1.

Table 1. Overview of thesis work

	2018	2019	2020	2021	2022
Study I	Data collection	Data collection	Published		
Study II		Data collection		Published	
Study III		Data collection		Published	
Study IV				Data collection	Third revision
Study V			Data collection	Data collection	Published

The PhD study started in 2018, at which point the author was immediately involved in the newly introduced professional development courses in programming for teachers at Mid Sweden University, supported by the Swedish National Agency for Education. These courses were introduced in response to the updated curriculum a year prior, which introduced programming in K-12 education. The courses were directed towards teachers of mathematics and technology in grades 7-9 and teachers of mathematics in grades 10-12, and this became the initial focus for the thesis project. Between 2018 and 2019, three iterations of the professional development course were run, during which data were collected for Study I. The study was published in *The International Journal of Information and Learning*

Technology in 2020. During the period of data collection and analysis for Study I (further described in *4.1 Data collection* and *4.2 Data analysis*), the theoretical lens for the thesis project had not yet been decided and instead categories for programming and programming tools identified in previous research were used as a theoretical framework for analysis. The findings of Study I, and its contributions to the thesis, are presented in *5 Summary of studies*.

In 2019, two further studies for the thesis project started: Studies II and III. Study II used the same theoretical framework identified for Study I but applied a wider approach to programming in K-12 education. Data were collected from a Swedish website (Lektion.se) through which K-12 teachers can share teaching and learning materials with each other. All teaching and learning materials on programming that could be identified by the author, regardless of year (grade) and subject, were included in the study. The intention was to form a more general understanding of how programming can be used by teachers. Data collection and analysis for Study II are further described in *4.1 Data collection* and *4.2 Data analysis*, and the findings and contribution to the thesis are presented in *5 Summary of studies*. Study II was published in the *Proceedings of the 20th European Conference on E-Learning (ECEL)* in 2021.

Study III introduced a narrower scope (compared to Studies I and II), conducting focus group discussions with teacher teams in grade 7-9 mathematics and technology (further described in *4.1 Data collection*). A new theoretical lens was also chosen for Study III and the thesis project, computational thinking (CT). However, during focus group discussions and data analysis, the author felt that this was not an ideal theoretical lens. Most of the teacher teams' discussions did not relate to CT. Teachers seemed more interested in discussing the possibilities

and constraints of using and integrating programming and different programming tools than the problem-solving methods and skills used and developed by their students. Several definitions and frameworks for CT were tested but the analysis felt forced.

It was eventually decided that CT should be abandoned as the theoretical lens for Study III and the thesis project. Several alternative theories were tested on the recommendation of the thesis supervisors, finally resulting in affordances as the choice of theoretical lens. Affordances resonated with how the teachers spoke about programming and what they highlighted as important. When conducting the analysis (further described in *4.2 Data analysis*), the feeling was that affordances supported the interpretation of the collected data, rather than forcing it to fit a template. Affordances is also a common theory in computer and system sciences, such as human-computer interaction and information systems, but it is not as commonly applied in research on programming in K-12 education as CT. This also raised the possibility of providing additional contributions of interest to research on the study object. Affordances was therefore chosen as the new theoretical lens for Study III and the thesis project. Study III was published in the journal of *Education and Information Technologies* in 2021. The findings and contributions of Study III to the thesis are presented in *5 Summary of studies*.

Study IV commenced in 2021 and was significantly influenced by Study III. Study IV had the same scope as Study III, teachers in grade 7-9 mathematics and technology, but applied the theoretical lens of affordances from the start to allow for a more rigorous use of the theory. This also supported the development of a conceptual model, which was not done in Study III. Further, Study IV sought to focus more on the perceptions and use of programming by teachers who were more

experienced in programming. In Study III, levels of experience in programming were varied among teachers. Study IV collected data through interviews with teachers in grade 7-9 mathematics and technology from all over Sweden (further described in *4.1 Data collection*) and used affordances as the theoretical lens for analysis (further described in *4.2 Data analysis*). Study IV was submitted for publication in 2022 and is currently in the third round of revision. The findings of Study IV, and its contributions to the thesis, are presented in *5 Summary of studies*.

Study V commenced in 2020 with the redesign of the professional development course in programming that had been running at Mid Sweden University since 2018. It was decided by the Swedish National Agency for Education that courses should be 5 credits instead of the previous 7.5 and that the focus of courses should be specific to either textual or block programming. Mid Sweden University decided to focus on textual programming courses using the programming language of Python. Previously, the courses at Mid Sweden University had also included block programming with Scratch, tangible programming with Micro:bit and unplugged programming. Course assignments were also required to be changed or redesigned to accommodate the reduced workload for participants.

In redesigning the professional development course, the final course assignment was changed from creating a programming project to creating a lesson plan on programming. This was a result of both the need to reduce the workload for participants and analysing course evaluations from previous course iterations. The course evaluations showed that teachers wanted more concrete take-aways from the courses that could be immediately applied in teaching and learning activities. The lesson plans were collected for Study V from three

iterations of the new professional development course (further described in *4.1 Data collection*). The analysis was conducted with affordances as the theoretical lens (further described in *4.2 Data analysis*) and could be viewed as a development of Study II, which collected similar material but had a wider scope and did not use affordances as a theoretical lens. Study V offered a new opportunity to examine teaching and learning materials on programming that were more in line with the new focus of the thesis project. Study V was accepted for publication in the *International Journal of Information and Learning Technology* in 2022. The findings of Study V, and its contributions to the thesis, are presented in *5 Summary of studies*.

The process of the thesis project, and the relationship between the included studies, could be described as design-inspired or compared to design-based research (DBR). Although DBR was not applied in any of the included studies, each study was intended to further develop the findings for the thesis project and fill in gaps identified in previous studies. Studies I-V could therefore be viewed as iterations of testing used to fine-tune the answer to the overall objective of the thesis (presented in *1 Introduction*). For example, Study III developed on Study I and Study II by applying affordances as a theoretical lens and narrowing the scope of study. Study IV developed on Study III by applying a more thorough research design to the theoretical lens of affordances. Study V was the result of a redesigned professional development course on programming and could also be considered a development on Study II, applying affordances to a similar subject of study while narrowing the scope. However, the process of further developing the studies in the thesis project should not be viewed as having made the previous studies obsolete, but rather as a process of

developing new studies by critically reviewing those that came before and seeking to fill in the gaps.

It should also be noted that other studies have been conducted during the thesis journey to support Studies I-V. Data collected for Study I were analysed, discussed and presented as papers at three conferences (Humble & Mozelius, 2019; Humble, Mozelius & Sällvin, 2019a; Humble, Mozelius & Sällvin, 2019b), in a book chapter (Humble, Mozelius & Sällvin, 2020) and in a popular science article (Humble, Sällvin & Mozelius, 2019), before being finalised and accepted for journal publication. Findings from Study I were further summarised and presented as a book chapter published in Swedish on the possibilities and challenges of programming in Swedish K-12 mathematics (Humble, Mozelius & Sällvin, 2021a). To support the intended use of CT in Study III, two literature studies were conducted and presented as conference papers on block programming for CT (Humble, 2019) and textual programming for CT (Humble, 2020).

In the redesign of the professional development course that enabled the new course assignment used for Study V, previous course iterations were analysed to facilitate a redesign based on lessons learnt and participant needs. This work was presented as two separate conference papers (Humble & Mozelius, 2021; Mozelius & Humble, 2020). To better understand and apply the methods used for data analysis in the thesis project (content analysis and thematic analysis), these have been studied and presented as a conference paper (Humble & Mozelius, 2022a) and a journal article (Humble & Mozelius, 2022b). Several more studies have been conducted and published between 2018 and 2022 that are less closely related to the thesis. These have influenced the author's development as a researcher. A full list of publications during the thesis journey is provided in Table 2.

Table 2. Full list of publications

	Author(s)	Year	Title	In	Description
#1	Hrastinski et al.	2019, May	Critical Imaginaries and Reflections on Artificial Intelligence and Robots in Postdigital K-12 Education	Postdigital Science and Education	Journal article
#2	Humble, Mozelius & Sällvin	2019, June	Teacher challenges and choice of programming tools for teaching K-12 technology and mathematics	International Conference on Education and New Developments (END 2019)	Paper in proceedings
#3	Humble & Mozelius	2019, July	Teacher perception of obstacles and opportunities in the integration of programming in K-12 settings	International Conference on Education and New Learning Technologies (EDULEARN 2019)	Paper in proceedings
#4	Humble, Mozelius & Sällvin	2019, Nov	On the Role of Unplugged Programming in K-12 Education	European Conference on e-Learning (ECEL 2019)	Paper in proceedings
#5	Humble & Mozelius	2019, Nov	Artificial Intelligence in Education – A Promise, a Threat or a Hype?	European Conference on the Impact of Artificial Intelligence and Robotics (ECIAIR 2019)	Paper in proceedings
#6	Humble & Mozelius	2019, Nov	Teacher-Supported AI or AI-Supported Teachers?	European Conference on the Impact of Artificial Intelligence and Robotics (ECIAIR 2019)	Paper in proceedings
#7	Humble	2019, Nov	Developing computational thinking skills in K-12 education through block programming tools	International Conference of Education, Research and Innovation (ICERI 2019)	Paper in proceedings
#8	Humble & Mozelius	2019, Nov	Learning analytics for programming education: Obstacles and opportunities	International Conference of Education, Research and Innovation (ICERI 2019)	Paper in proceedings

#9	Humble, Sällvin & Mozelius	2019, Nov	Programmering i matte och teknik	Pedagogiska magasinet (Lärarförbundet)	Summary of research in teacher union magazine
#10	Mozelius & Humble	2020, March	Lessons learnt from teacher professional development in programming	International Technology, Education and Development Conference (INTED 2020)	Paper in proceedings
#11	Humble, Mozelius & Sällvin	2020, June	The introduction of programming in K-12 technology and mathematics: Teacher choice of programming tools and their perceptions of challenges and opportunities	Education Applications & Developments V	Book chapter
#12	Humble	2020, July	Using textual programming tools to develop computational thinking skills in K-12 education	International Conference on Education and New Learning Technologies (EDULEARN 2020)	Paper in proceedings
#13	Humble	2020, July	Developing a web application for auto-generating grammar tests	International Conference on Education and New Learning Technologies (EDULEARN 2020)	Paper in proceedings
#14	Jandrić et al.	2020, Aug	Teaching in the Age of Covid-19	Postdigital Science and Education	Journal article
#15	Humble, Mozelius & Sällvin	2020, Aug	Remaking and reinforcing mathematics and technology with programming – Teacher perceptions of challenges, opportunities and tools in K-12 settings	International Journal of Information and Learning Technology	Journal article
#16	Humble	2021, Aug	The Development of Computational Thinking Concepts in Course Participants' Programming Solutions	Nordic Learning Analytics (Summer) Institute (NLASI 2021)	Paper in proceedings
#17	Jandrić et al.	2021, Aug	Teaching in the Age of Covid-19—1 Year Later	Postdigital Science and Education	Journal article

#18	Humble, Mozelius & Sällvin	2021, Sep	You can't escape learning, but maybe you can get out of the room! – Game-based learning for programming education	European Conference on Game Based Learning (ECGBL 2021)	Paper in proceedings
#19	Humble	2021, Oct	The use of Programming Tools in Teaching and Learning Material by K-12 Teachers	European Conference on e-Learning (ECEL 2021)	Paper in proceedings
#20	Humble & Mozelius	2021, Oct	Enhancing Pedagogy to Andragogy in the Redesign of Teacher Training Courses on Programming	European Conference on e-Learning (ECEL 2021)	Paper in proceedings
#21	Humble & Mozelius	2021, Nov	Computerised Consequentialism to Support Moral Reasoning and Decision Making in Crisis Management	European Conference on the Impact of Artificial Intelligence and Robotics (ECIAIR 2021)	Paper in proceedings
#22	Humble, Mozelius & Sällvin	2021, Nov	Olika programmeringsverktyg och lärares fortbildning: Hinder och möjligheter	Programmering i skolmatematiken: möjligheter och utmaningar	Book chapter
#23	Mozelius & Humble	2022, March	Programming in K-12 mathematics – A two-step rocket	International Technology, Education and Development Conference (INTED 2022)	Paper in proceedings
#24	Humble & Mozelius	2022, June	Content Analysis or Thematic Analysis: Similarities, Differences and Applications in Qualitative Research	European Conference on Research Methodology for Business and Management Studies (ECRM 2022)	Paper in proceedings
#25	Jandrić et al.	2022, Sep	Teaching in the Age of Covid-19—The New Normal	Postdigital Science and Education	Journal article
#26	Humble & Mozelius	2022, Oct	Refurbishing the Educational Escape Room for Programming: Lowering the Threshold and Raising the Ceiling	European Conference on Games Based Learning (ECGBL 2022)	Paper in proceedings

#27	Mozelius et al.	2022, Oct	How to get the girls Gaming: A Literature Study on Inclusive Design	European Conference on Games Based Learning (ECGBL 2022)	Paper in proceedings
#28	Humble	2022, Nov	Teacher observations of programming affordances for K-12 mathematics and technology	Education and Information Technologies	Journal article
#29	Humble & Mozelius	2022, ahead of print	Making programming part of teachers' everyday life – Programming affordances and constraints for K-12 mathematics and technology	International Journal of Information and Learning Technology	Journal article
#30	Humble & Mozelius	2022, Nov	The threat, hype, and promise of artificial intelligence in education	Discover Artificial Intelligence	Journal article
#31	Humble & Mozelius	2022, Nov	Content Analysis or Thematic Analysis: Doctoral Students' Perceptions of Similarities and Differences	Electronic Journal of Business Research Methods	Journal article
#32	Mozelius & Humble	2022, Nov	Design Factors for an Educational Game where Girls and Boys Play Together to Learn Fundamental Programming	EAI ArtsIT 2022	Paper in proceedings
#33	Bergman & Humble	2022, Dec	Sentiment Analysis for Emotional Navigation in Written Communication: What Support do Autistic People Need?	European Conference on the Impact of AI and Robotics (ECIAIR 2022)	Paper in proceedings

4 Methodology

> With the help of the Turtle module, write code that draws an equilateral triangle. Write the code in three different ways to achieve the same result.
>
> (Lesson material in Study II)

The thesis and included studies apply a qualitative research approach. Silverman (2020, p. 3) notes that there is no general agreement on the nature of qualitative research. However, there are some descriptions of the qualitative approach that are more common than others. According to Bryman (2016, p. 33), the qualitative approach typically emphasises induction and individuals' interpretation and creation of the social world. In this thesis, induction is applied in the analysis of collected data. However, induction is combined with deduction in a deductive-inductive or inductive-deductive approach, in which previous research or theory is applied as a theoretical lens (further described in *4.2 Data analysis*). Teachers' interpretation and creation of programming for teaching and learning in grade 7-9 mathematics and technology are central in both the thesis and included studies.

According to Kaplan and Maxwell (2005, p. 30), the goal of qualitative research is to investigate the behaviours and perspectives of people to understand a specific situation or issue. The overall objective of the thesis is to address the issue of teachers lacking guidance, knowledge and skills to see the potentials and possible pitfalls of programming for teaching and learning. To understand and address this issue, teachers' perceptions and behaviours (as understood through perceived action possibilities, constraints and use of programming) are studied in different situations using different qualitative methods, as presented in the included Studies I-V of the thesis (further described in *4.1 Data collection*).

4.1 Data collection

Data collected in Studies I-V contributed to answering the aim and research questions of the thesis. The same study object is approached in the studies with different qualitative methods used for data collection. This can be compared to the idea of triangulation, which is described by Denzin (2015) as a multi-method approach to studying a phenomenon. Flick (2004) notes that there are several types of triangulation, such as data triangulation, investigator triangulation, theory triangulation and methodological triangulation. Data collection for this thesis draws on the idea of data triangulation, which combines data from different sources, times, places and peoples (Flick, 2004). A summary of the methods used for data collection in Studies I-V is provided at the end of this section; see Table 3.

Study I applies a case study approach to address teachers' perceptions on integrating programming in the teaching and learning of mathematics and technology and their perceptions of different programming tools. This connects to the aim and research questions of the thesis. However, Study I had a wider focus compared to this thesis (grades 7-12 instead of grades 7-9) and did not apply the theoretical lens of affordances (this is further described in *3 Thesis journey* and *5.1 Study I*). Data were collected in the context of three iterations of a 7.5-credit professional development course on programming directed at grade 7-12 teachers in mathematics and technology. Participants were mainly located in the mid Sweden region, since the courses were held on campus in Östersund and Sundsvall. Participants in the study were evenly distributed between male and female teachers.

Data were collected between 2018 and 2019 and consist of 55 course essays, 33 postings on the online course forum and 20 workshop observations. In the essays, teachers reflected on their perceptions of textual programming with Python and block programming with Scratch and the opportunities and challenges they perceived with learning programming. In the forum postings, teachers reflected on and discussed their experiences of unplugged programming during course workshops and opportunities and challenges for classroom application. Workshops were conducted during the courses and comprised collaborations and discussions on programming assignments and the application of programming for teaching and learning activities in mathematics and technology. Python and Scratch were the most commonly used programming tools, but three of the workshops also included unplugged programming.

Study II was conducted as a systematic document review, inspired by systematic literature reviews, to analyse teachers' use of programming for teaching and learning in K-12 education. This included the programming tools they used, how they used them and for which subjects and years (grades) they used them. This connects to the aim and research questions of the thesis. However, Study II had a wider focus compared to the thesis (including several additional years and subjects compared to the thesis) and did not apply the theoretical lens of affordances (this is further described in *3 Thesis journey* and *5.2 Study II*). Data were collected in late 2019 from a Swedish website (Lektion.se) through which teachers can share teaching and learning materials.

A systematic three-stage process was used to select the teaching and learning materials included in the study: 1) a database search for potentially relevant teaching and learning materials (332 identified), 2) a title and summary screening of the teaching and learning materials

with inclusion criteria (62 potentially relevant identified) and 3) a screening of all teaching and learning materials with inclusion criteria (26 selected for inclusion). Selected teaching and learning materials were published between 2008 and 2019 and included a number of subjects across all years (grades) between kindergarten and grade 12. The gender distribution and geographical location of teachers in Study II is difficult to determine since the only information provided by respondents was their name, which may or may not have been their real name.

Study III was conducted as a focus group study to address teachers' perceptions of integrating programming in teaching and learning for grade 7-9 mathematics and technology and their use of programming for teaching and learning in these subjects. This study aligns with the aim and research questions of the thesis, has the same scope as the thesis (mathematics and technology in grades 7-9) and applies the theoretical lens of affordances (this is further described in *3 Thesis journey* and *5.3 Study III*). Teachers were contacted through invitation e-mails sent to all K-12 schools in two big municipalities in the mid Sweden region and invitation posts on a forum for a professional development course in programming for teachers. Three teacher teams, from three different schools in the mid Sweden region, accepted the invitation and met the criteria of teaching grade 7-9 mathematics or technology.

Focus group discussions were conducted during 2019 with the three teacher teams during a team meeting at their workplace. A total of 21 teachers (8 female and 13 male) participated in the focus groups; they were all familiar with programming but only 8 considered themselves to have competence in programming prior to its integration with the curriculum. The focus group discussions lasted about 48 minutes on

average and used the following questions as guidelines: How far have you come in the integration? How do you integrate programming in mathematics and technology? What programming tools do you use? What challenges and opportunities do you perceive?

Study IV was conducted as an interview study to address teachers' perceptions of what programming provides for grade 7-9 mathematics and technology. This study aligns with the aim and research questions of the thesis, has the same scope as the thesis (mathematics and technology in grades 7-9) and applies the theoretical lens of affordances (this is further described in *3 Thesis journey* and *5.4 Study IV*). Teachers were contacted through invitation posts on course forums for profession development courses for teachers, social media and by email; about 700 e-mails were sent to teachers, principals and administrators at 64 municipalities across Sweden. Twenty-five teachers registered their interest in participating in the study and 20 met the criteria of teaching grade 7-9 mathematics or technology and had experience in using programming for teaching and learning. Of these 20 teachers, 1 did not reply to the final interview invitation.

Semi-structured interviews were conducted during 2021 with the remaining 19 teachers (5 female and 14 male) selected for the study. The teachers were located all over Sweden, had teaching experience ranging from 3 to 40 years and experience using programming in education ranging from 1 to 30 years. Interviews were conducted using video conference tools (Zoom and Teams) and had an average duration of about 36 minutes. The interviews included questions on how programming is used in each subject, what programming tools are used, what programming can and cannot provide for teaching and learning, similarities/differences in programming use between

different subjects and years (grades), and recommendations for other teachers.

Study V was conducted as a document analysis to address teachers' use of programming in teaching and learning for mathematics and technology. This connects to the aim and research questions of the thesis. Study V had a wider focus compared with the thesis (grades 7-12 instead of grades 7-9) but did apply the theoretical lens of affordances (this is further described in *3 Thesis journey* and *5.5 Study V*). Data were collected from three iterations of a 5-credit professional development course on programming directed at grade 7-12 teachers in mathematics and technology. These courses were redesigned versions of a previous professional development course on programming used in Study I. This development is further described in *3 Thesis journey*.

Participants in the professional development courses used for Study V were located all over Sweden. Since these courses were conducted online via the video conference tool Zoom, participants were no longer constrained by having to be located (relatively) close to Östersund or Sundsvall to attend the course, which had been the case for the earlier version of the course (used for Study I). Data were collected between 2020 and 2021 and consists of 31 lesson plans. Of these lesson plans, 13 were produced by female teachers and 18 by male teachers. The instructions for the assignment were to create teaching and learning material on programming to suit their teaching and learning activities and to include reflections and motivations for the choices made in the material.

Table 3. Summary methods for data collection

	Method	Data
Study I	Case study	55 essays, 33 forum postings and 20 workshop observations
Study II	Systematic document review	26 teaching and learning materials
Study III	Focus group interviews	3 recorded focus group interviews with a total of 21 teachers
Study IV	Semi-structured interviews	19 recorded interviews with teachers
Study V	Document analysis	31 lesson plans

4.2 Data analysis

Data analyses for Studies I-V were conducted using qualitative methods to interpret collected data and identify patterns of meaning. The two methods used for qualitative analysis were content analysis and thematic analysis. Hsieh & Shannon (2005) describe qualitative content analysis as a method "to interpret meaning from the content of text data" and state that three distinctive approaches can be identified: conventional content analysis, directed content analysis and summative content analysis. Content analyses, directed by either theory or categories identified in previous research, were applied for Studies I, II and V. Clarke and Braun (2017) describe thematic analysis as a method to identify, analyse and interpret patterns of meaning (themes) in data and that this can be driven by theory (deductively) or by data (inductively). Thematic analyses, with a mixture of deduction and induction, were applied in Studies III and IV. A summary of the methods used for analysis in Studies I-V is provided at the end of this section, in Table 4.

The choice between content analysis and thematic analysis in the included studies was guided by the nature of the collected data. Content analysis was used for studies in which data consisted mainly of documents or written text, while thematic analysis was used for studies in which data consisted of interviews or verbal data. Content analysis, compared to thematic analysis, allows for quantifying data and measuring the frequency of categories and themes (Vaismoradi, Turunen & Bondas, 2013). The frequency of different themes and categories was considered more interesting for studies comprised of documents and text data (Studies I, II and V), where data collection is

often larger than for studies with interviews or verbal data (Studies III and IV).

Study I applied content analysis to interpret meanings in course essays, online postings on course forums and workshop observations from three iterations of a teacher professional development course in programming (further described in *4.1 Data collection*). Analysis was conducted with a mixture of deductive and inductive coding and can be described as a series of steps. Assarroudi et al. (2018) suggest 16 steps for conducting a directed content analysis, the last 9 of which concern the analysis phase and are used in this case to describe the analysis in Study I:

1. Categorisation matrix
2. Definition of categories
3. Coding rules
4. Pre-testing
5. Anchor samples
6. Data analysis
7. Inductive abstraction
8. Establish links
9. Report steps and findings

First, a formative categorisation matrix was developed in a spreadsheet document with the chosen categories for analysis. Second, the categories were defined through previous research and presented in summarised form as the two research questions for Study I and in more detail in the background section of Study I. Third, the coding

rules were determined for identifying codes and extracts that emphasise the challenges and opportunities of learning and integrating programming and of different programming tools. Fourth, pre-testing of the categorisation matrix was integrated into Study I through discussions between the three authors and conference presentations and discussions of early preliminary results (further described in *3 Thesis journey*). Fifth, previous research that was used to develop the categorisation matrix and presented in the background section of Study I was applied as anchor samples for the categories.

Sixth, the main data analysis was conducted by adding identified codes from the course essays and forum postings to the relevant categories in the spreadsheet document. Seventh, with an inductive approach the preliminary codes were clustered into themes (or subcategories) of specific challenges and opportunities within each category. The themes were also compared to workshop observations for further development. Eighth, to establish a link between the identified themes and the categories for analysis, extracts from the collected data were chosen to represent the themes and to be used in the results and analysis section of the study. In the ninth and final step, the study was reported by describing steps for data collection and analysis in the method section and presenting the identified themes in the results and analysis section with the categories for analysis as subheadings (further described in *5 Summary of studies*).

Study II applied content analysis to interpret meanings in teaching and learning materials on programming for K-12 education from a Swedish website, Lektion.se (further described in *4.1 Data collection*). The analysis was conducted with a mixture of deductive and inductive coding in a series of steps but also included counting and frequency of programming tools, school subjects and year groups (grades) in the

collected material. Counting and comparisons of different keywords and content to interpret the underlying context are usually conducted in summative content analysis (Hsieh & Shannon, 2005). Therefore, the analysis in Study II could be described as a combination of directed and summative content analysis. Assarroudi et al.'s (2018) last 9 steps for directed content analysis are used here to describe the analysis in Study II.

First, a formative categorisation matrix was developed in a spreadsheet with the chosen categories for analysis. Second, the categories were defined through previous research and presented in the theoretical background section of Study II. Third, coding rules were decided: identifying codes and extracts that exemplify the use of different programming tools and counting the frequency of programming tools, school subjects and year groups (grades) in the collected material. Fourth, pre-testing of the categorisation matrix for Study II was conducted through discussions with thesis supervisors. Fifth, previous research presented in the theoretical background section of Study II was applied as anchor samples for the categories.

Sixth, the main data analysis was conducted by adding identified codes of programming use in the relevant categories of the spreadsheet document and by counting the frequency of different programming tools, year groups (grades) and subjects being addressed in the material. Seventh, using an inductive approach, the preliminary codes of programming use were clustered into themes (or subcategories) based on similarities, differences and meanings within each category. Eighth, extracts from the collected data were chosen to establish a link between identified themes and categories for analysis and to represent the themes in the results section of the study. In the final step, the study was reported by describing steps for data collection and analysis in the

method section and presenting the identified themes and frequencies in the results section with the categories for analysis as subheadings (further described in *5 Summary of studies*).

Study III applied thematic analysis to interpret patterns of meanings in focus group discussions with teacher teams on programming in grade 7-9 mathematics and technology (further described in *4.1 Data collection*). The analysis used a mixture of deductive and inductive approaches and was conducted in a series of steps. Braun and Clarke (2012) suggest six phases for conducting a thematic analysis: 1) familiarisation with the data, 2) generating initial codes, 3) searching for themes, 4) reviewing potential themes, 5) defining and naming themes and 6) producing the report. These phases are used here to describe the analysis in Study III. However, an initial step of developing a categorisation matrix was added since the study used affordances as a theoretical lens. The categorisation matrix consisted of two categories in a text document: action possibilities and constraints.

0. Categorisation matrix
1. Familiarisation with the data
2. Generating initial codes
3. Searching for themes
4. Reviewing potential themes
5. Defining and naming themes
6. Producing the report

First, focus group discussions were moderated and recorded by the author to ensure familiarity with the data. Recordings were listened to repeatedly and notes taken on topics that the teachers emphasised in the discussions for data immersion. Second, initial codes were generated in Study III by highlighting extracts from the discussions and collecting them in the relevant categories in the categorisation matrix of the text document. Third, the identified codes were reviewed using an inductive approach within the categorisation matrix for similarities and overlap in the search for themes. This phase was repeated for each of the identified themes to search for new and more specific categories for the categorisation matrix. This resulted in five new categories that replaced the initial two.

Fourth, the themes were reviewed in relation to extracts from the focus group discussions and compared to the theory and categories in the categorisation matrix for quality checking. Fifth, themes and categories were defined and named based on comparisons with the theory and extracts from the focus group discussions with the idea of identifying a singular focus to address the aim and research questions of the study. Lastly, data collection and analysis were described in the method section of the study and identified themes were reported in the results and analysis section, with the new categories in the categorisation matrix as subheadings (further described in *5 Summary of studies*).

Study IV applied thematic analysis to interpret patterns of meanings in interviews with teachers on programming for mathematics and technology in grades 7-9. The analysis was conducted in a series of steps and used a mixture of deductive and inductive approaches. The six phases for conducting a thematic analysis, suggested by Braun and Clarke (2012), are used here to describe the analysis. However, as with Study III, Study IV used affordances as a theoretical lens, and an initial

step of developing a categorisation matrix was added. The categorisation matrix consisted of two categories in a spreadsheet document: action possibilities and constraints.

First, the interviews were conducted and recorded by the author to ensure familiarity with the data for analysis. Recordings were listened to repeatedly and notes were taken on the topics emphasised by teachers in the interviews to achieve data immersion. Second, initial codes were generated in Study IV by identifying codes of interest directly after each interview and collecting them in a text document, separate from the categorisation matrix. This was done inductively and focused on identifying extracts in the interviews that the teachers had emphasised as important, recurred in conversation or related to the aim and research questions of the study. Third, deductive and inductive approaches were used to search for themes. A deductive approach was used to review the relevance of identified codes to theory and either include in the categorisation matrix or discard. An inductive approach was used to review the included codes for similarities and overlap within each category in searching for themes. The identified themes were then reviewed again using an inductive approach when searching for new and more specified categories for the categorisation matrix. Lastly, a deductive approach was used to analyse each interview again in search for additional codes that could support, rephrase or form new themes and categories. These steps in the third phase resulted in ten new categories that replaced the initial two.

Fourth, the themes in Study IV were reviewed in relation to extracts from the interviews and compared to theory and the categorisation matrix for quality checking. Fifth, themes and categories were defined and named based on comparisons with the theory and extracts from

interviews, with the idea of identifying a singular focus that addressed the aim and research questions of the study. Sixth, data collection and analysis were described in the method section and identified themes were reported in the results and analysis section with the new categories in the categorisation matrix as subheadings (further described in *5 Summary of studies*). Finally, in what could be considered the final step of analysis, categories and themes were compared to related research in the discussion section of the study. This identified three aspects of teaching and learning related to the themes and categories, forming a potential conceptual model. The conceptual model was presented in the conclusion section of Study IV and has been further developed in this thesis (presented in *7 Concluding remarks*).

Study V applied content analysis to interpret meanings in lesson plans on programming from three iterations of a professional development course in programming for grade 7-12 teachers in mathematics and technology (further described in *4.1 Data collection*). Analysis was conducted in a series of steps using a mixture of deductive and inductive coding. The last nine steps for directed content analysis in Assarroudi et al. (2018) are used here to describe the analysis.

First, a formative categorisation matrix was developed in a spreadsheet with the chosen categories for analysis. Second, the categories were defined through literature and previous research on the theory of affordances and presented in the study. Third, coding rules were determined: identifying codes and extracts that exemplify action possibilities and constraints of programming for mathematics and technology. Fourth, pre-testing of the categorisation matrix was conducted through discussions with the second author and supervisors of the thesis. Fifth, concept definitions and examples from

previous research in the theory section of Study V were chosen as anchor samples for the categories.

Sixth, the main data analysis in Study V was conducted by adding identified codes for action possibilities and constraints of programming to the relevant categories in the spreadsheet. Seventh, using an inductive approach, the preliminary codes were clustered within each category into themes (or subcategories) based on their similarities, differences and meanings. To create more specific categories, the seventh step was repeated with the subcategories instead of the preliminary codes. This resulted in five new categories that replaced the initial two in the categorisation matrix. Eighth, extracts from the collected data were chosen to establish a link between identified themes and categories and to represent the themes in the results and analysis section of the study. Lastly, steps for data collection and analysis were reported in the method section of the study and identified themes were presented in the results and analysis section with the new categories in the categorisation matrix as subheadings (further described in *5 Summary of studies*).

Table 4. Summary methods for data analysis

	Method	Coding approach
Study I	Directed content analysis	9 steps for directed content analysis
Study II	Directed summative content analysis	9 steps for directed content analysis
Study III	Thematic analysis	6 phases for thematic analysis + initial phase of developing categorisation matrix
Study IV	Thematic analysis	6 phases for thematic analysis + initial phase of developing categorisation matrix
Study V	Directed content analysis	9 steps for directed content analysis

4.3 Trustworthiness

According to Guba (1981), trustworthiness in research is a question of addressing several concerns: How can confidence in the truth of the findings be established (truth value)? How can the degree of applicability to other contexts be determined (applicability)? How can consistency in replication be determined (consistency)? And how can the degree of neutrality be established (neutrality)? Guba (1981) suggests that these questions can be addressed by applying four criteria of trustworthiness:

1. Credibility, to address the concern of truth value
2. Transferability, to address the concern of applicability
3. Dependability, to address the concern of consistency
4. Confirmability, to address the concern of neutrality

These four criteria are discussed here in relation to the thesis. According to Shenton (2004), the researcher can promote credibility in research using scientific methods that are well established in qualitative or quantitative research and by developing a familiarity with the organisation prior to data collection. All included studies in this thesis use established qualitative research methods for data collection and data analysis. Furthermore, the author has a background as a teacher and IT coordinator in K-12 education and has an understanding and familiarity with the organisational setting for the thesis and included studies.

Qualitative research is often conducted with small samples, and the quest for transferability can therefore be difficult (Shenton, 2004). As suggested by Bassey (1981), transferability in research does not necessarily need to be limited to whether or not it can be generalised. Transferability could also be measured by the extent to which a reader can relate findings to their situation (Bassey, 1981). Since the integration of programming in K-12 education is a global phenomenon, it is likely that teachers and researchers interested in the field can relate to the study object of the thesis and included studies. To facilitate transferability of study results, the processes for data collection and analysis have been described in both the thesis and included studies. The findings presented in the thesis and included studies should not be understood as representative of teachers' perceptions and use of programming in general. Data have been collected about teachers in a context in which interest in programming for teaching and learning is central, and the findings should therefore be understood in relation to that.

Dependability is about ensuring similar results would be obtained if the research were repeated under similar conditions; this can be difficult for qualitative research because of the changing nature of the research context (Guba, 1981; Shenton, 2004). Shenton (2004) suggests that to address dependability in qualitative research, sufficient details about the research process should be provided that future researchers could repeat the study, even if they would not obtain the same results. It has been a goal of Studies I-V and the thesis to describe the process of data collection and data analysis in sufficient detail for the studies to be repeatable. In the case that details are missing in the included studies, either due to the word limit set by the publication or for other

reasons, these have been added and clarified in the methodology section of the thesis.

Confirmability is about the objectivity of the research, i.e. to what degree the research is derived from the studied subject and not researcher bias (Guba, 1981; Shenton, 2004). This can be difficult to ensure since the research design is conducted by the researcher (Guba, 1981; Shenton, 2004). Therefore, confirmability in qualitative research means taking steps to ensure freedom from the researcher's biases as far as possible (Shenton, 2004). Suggested approaches for this are triangulation, collecting data from multiple sources and perspectives, and providing clarity and openness on the underlying assumptions that have affected the design of the research (Guba, 1981; Shenton, 2004). Steps have been taken in the included Studies I-V to ensure confirmability as far as possible by describing the underlying motivation and aim of the research in relation to research gaps, problems identified in previous research, and theory or previous research used for analysis. In Study I, triangulation was used as part of the case study approach. Triangulation is also applied in the thesis by addressing the aim and research questions through multiple studies to derive a comprehensive understanding of the study object.

4.4 Ethical considerations

Resnik (2020) states that the reason for professions, institutions and disciplines having ethical norms or codes for their members is to help them coordinate actions and establish trust in the eyes of the public. Specific policies, rules and codes have been developed and adopted by government agencies, professional associations and universities for research ethics (Resnik, 2020). According to Shaw (2003), principles that are important to address in qualitative research include: informed consent, confidentiality and privacy, social justice and practitioner research. These principles, together with the rules provided by the Swedish Research Council (2017) and Sveriges riksdag [the Swedish Parliament] (2003), are used here to present the ethical considerations of the thesis and included studies.

The principle of informed consent is applied in research to protect participants from abuse (Shaw, 2003). The principle is also addressed in the Swedish Ethical Review Act (Sveriges riksdag [Swedish Parliament], 2003) and in the rules for good research practice (Swedish Research Council, 2017). A problem that qualitative researchers face with informed consent is that it is difficult to determine to what extent a situation affects or hinders participants from giving informed consent or understanding the circumstances for giving informed consent (Dewing, 2002; Shaw, 2003). One such example is when children participate in educational research (Moore, McArthur & Noble-Carr, 2018). Studies I-V collected data on teachers' use and perceptions of programming, and no children or K-12 students are therefore included in any of the studies.

Informed consent was sought from teachers who participated in the studies (Studies I, III, IV and V). One exception is Study II, which was based on teaching and learning materials created by teachers and shared on a public website (Lektion.se). The author never contacted the teachers who created the material, and because the material is published under the licence of Creative Commons, informed consent was never sought. Furthermore, ethical reviews were not conducted for any of the included studies since none of them presented a risk of harm to participants.

The principle of confidentiality and privacy is especially important in qualitative research since it does not provide, for example, the same kinds of safeguards in the sampling process as quantitative research (Shaw, 2003). This is also addressed in the rules for good research practice (Swedish Research Council, 2017) and the Swedish Ethical Review Act (Sveriges riksdag [Swedish Parliament], 2003) regarding the handling of personal data. This principle could be raised to challenge Study I and Study V, since the teachers who participated could be identified through the information provided about the professional development courses. However, since data collection in both studies was conducted over three instances of the course (six instances in total) with a large number of course participants and focused on themes or categories of use and perception, it should be very difficult to identify individual teachers in either study.

Study III could also be questioned in relation to this principle since the material was collected through focus groups with a relatively small number of participants. Liamputtong (2011, p. 25) states that confidentiality should be ensured for the participants in focus groups but that this can be difficult in practice since participants can reveal each other's identities. Liamputtong (2011, p. 26) therefore

recommends that the researcher remains mindful that the discussion does not move into any sensitive domains that could pose personal harm to any of the participants. These recommendations were followed in the moderation of the focus groups in Study III. Furthermore, no personal data that could be considered sensitive, according to the definition provided by the Swedish Authority for Privacy Protection (2021), were collected or stored for the included studies.

The principle of social justice in qualitative research raises the question of power dynamics between a researcher and participant, and results should be reported with contextual relevance (Shaw, 2003). This principle is also addressed in the rules for good research practice (Swedish Research Council, 2017), which state that a researcher should be truthful in presenting conflicts of interest. The thesis and included studies have provided information about the contexts for the studies and sought to be truthful about the results by presenting the application of established qualitative methods for data collection and analysis. Power dynamics could be considered an ethical issue in Studies I and V, in which the author acted as both researcher and course facilitator. However, given that the context for data collection was professional development courses for teachers with course outcomes having no influence on their future careers, the double role of the author should not be considered an issue. Furthermore, teachers who participated in the included studies were informed that they could withdraw consent before, during or after data collection.

The ethical issues of practitioner research are especially relevant in Studies I and V, relating to the issue of power dynamics for the principle of social justice. In Study I and Study V, the author had the dual role of researcher and course facilitator. According to Shaw (2003),

the principle of informed consent relies on participation being voluntarily, but this can be difficult to ensure if participants are dependent on the researcher(s), for example in the case of a teacher-student relationship. This issue was addressed in Studies I and V by raising the question of participation and data collection at the end of the courses when most assignments had already been submitted and graded. However, it still cannot be guaranteed that fear of retribution did not influence individuals' decisions on participation. This is a pressing ethical issue that the researcher needs to be mindful of when conduction practitioner research (Shaw, 2003). However, involuntary participation in the included studies due to fear of retribution is considered unlikely. The participating teachers had a much higher average age than typical higher education students and were often older than the author. Furthermore, teachers who participated in the professional development courses did so on their own initiative; the course was not mandatory and fear of retribution should be little to none.

5 Summary of studies

> I think that it stimulates their joy of discovery in some way. To get something to make a sound or write or display something.
>
> (Teacher in Study III)

The thesis includes five studies that have been conducted as separate research papers. All the included studies have contributed to the understanding of the study object of the thesis – teachers' use and perceptions of programming for teaching and learning in grade 7-9 mathematics and technology. Furthermore, the included studies have all contributed to answering the aim and research questions of the thesis – to examine teachers' use and perceptions of programming and identify action possibilities and constraints for programming in grade 7-9 mathematics and technology. An overview of identified action possibilities and constraints in Studies I-V is provided in *6 Discussion*. In this section, the included studies are summarised and presented under separate subheadings. Study I, Study III and Study V have been published in international journals. Study II has been published in the proceedings of a European conference. Study IV is in the third stage of revision for journal publication.

5.1 Study I

Humble, N., Mozelius, P., & Sällvin, L. (2020). Remaking and reinforcing mathematics and technology with programming – Teacher perceptions of challenges, opportunities and tools in K-12 settings. *The International Journal of Information and Learning Technology*, 37(5), 309-321. https://doi.org/10.1108/IJILT-02-2020-0021

Study I addresses teachers' perceptions of integrating programming in the teaching and learning of mathematics and technology and of different programming tools. Through a case study approach, the study provides insights into what teachers perceive to be the challenges and opportunities of integrating programming in their teaching and learning activities and using the different programming tools Python, Scratch and unplugged programming. Data were collected from three instances of a professional development course on programming directed at grade 7-12 teachers in the subjects of mathematics and technology in the form of workshop observations, essay assignments and forum postings (further described in *4.1 Data collection*). The collected data were analysed with content analysis to identify codes of interest and develop meaningful categories to answer the study's aim and research questions (further described in *4.2 Data analysis*).

The study shows that teachers perceive both challenges and opportunities in integrating programming in teaching and learning

activities for mathematics and technology. Regarding the challenges, teachers perceive programming to be difficult to learn and integrate due to a lack of directives from the school leadership and time for their own learning and development of teaching and learning activities. Regarding the opportunities, teachers perceive programming to be fun, that there are many accessible teaching and learning materials available and that programming can easily be linked to other teaching and learning activities in their subjects such as problem solving and logic.

The study further shows that teachers perceive both challenges and opportunities for different programming tools in teaching and learning activities for mathematics and technology. Regarding the challenges of different programming tools, teachers perceive Python to be less accessible than other alternatives, that Scratch is limited compared to Python and that unplugged programming is mainly directed towards younger students. Regarding opportunities for different programming tools, teachers perceive Python to provide greater freedom and to be a more useful skill in relation to education and the labour market compared to other alternatives, that Scratch is easier to learn than Python and has a lower threshold for novice programmers and that unplugged programming provides fun and easy support for other programming tools and activities such as mathematics, computational thinking and logical thinking.

Study I contributes to the thesis by identifying challenges and opportunities that teachers perceive for programming in mathematics and technology for grades 7-12. The findings contribute to an understanding of how teachers perceive programming in general and how they perceive different programming tools. Study I did not apply the theory of affordances when it was conducted. However, the

identified perceptions on integration, challenges and opportunities are closely connected to teachers' perceptions of how programming can and cannot be used for teaching and learning activities in their subjects. Therefore, these findings can be translated into action possibilities and constraints and are considered to align with the theoretical lens of affordances used in the thesis.

In *6 Discussion*, findings from Study I are related to action possibilities and constraints identified in Studies III-V (which applied affordances as a theoretical lens) and used to further develop the conceptual model presented in *7 Concluding remarks*. Furthermore, the study has a wider scope than the thesis, including mathematics teachers for grades 10-12. However, this is not considered an issue since no crucial differences relevant to the thesis aim and research questions were identified between teachers' perceptions for grades 7-9 and 10-12.

5.2 Study II

Humble, N. (2021). The use of Programming Tools in Teaching and Learning Material by K-12 Teachers. In *Proceedings of the 20th European Conference on E-Learning (ECEL 2021)*, (pp. 574-582). Academic Conferences and Publishing International Limited.

Study II addresses how teachers use programming for teaching and learning activities in K-12 education. Through a document review, the study provides insights into the programming tools teachers use, how they use them and potential relationships between programming tools, school subjects and student grades. Data were collected in the form of teaching and learning materials that address programming in K-12 education from a website (Lektion.se) through which teachers can share their teaching and learning materials with each other (further described in *4.1 Data collection*). The collected data were analysed with content analysis to identify codes of interest, connect the use of programming tools to school subjects and student grades and answer the study's aim and research questions (further described in *4.2 Data analysis*).

The study shows a variety of uses for programming in teaching and learning activities for K-12 education, and the programming tools identified can be categorised as textual, block, unplugged and tangible. Textual programming tools are mostly used in teaching and learning materials for grades 10-12, although they are also used for grades 7-9,

and in the subjects of mathematics and computer engineering. The main use of textual programming is as a teaching and learning tool in other subjects (such as mathematics and technology) and to teach programming. Block programming tools are mostly used in teaching and learning activities for the subjects of mathematics and technology, and the examples of it being used are evenly distributed across year groups (grades), except for grades 10-12, for which there are no use examples. All examples of block programming include the practice of controlling objects in virtual environments, often combined with content visualisation within the specific subject.

Unplugged programming is mostly used for teaching and learning activities for grades 7-9, but there are also use examples from kindergarten up to grade 3, and in the subject of mathematics. The main use of unplugged programming is centred on the practice of giving clear instructions and providing step-by-step solutions to problems. This is combined with geometry and controlling imagined objects using a game-based approach. Tangible programming tools are mostly used in teaching and learning activities for grades 2-3, but there are also examples of its use in all other grades, and in the subjects of technology and mathematics. The main use of tangible programming is to control physical objects, such as robots. For younger grades, this is combined with mathematics to visualise, for example, calculations.

Study II contributes to the thesis by examining the use of programming in teaching and learning activities for K-12 education. The study examines programming using a wider scope than addressed in this thesis, including more grades and subjects than mathematics and technology in grades 7-9. Furthermore, the study contributes to an understanding of how teachers use different types of programming tools in education.

As with Study I, the theory of affordances was not applied at the time the study was conducted and published. However, the identified use for the different programming tools in the teaching and learning materials should be considered representative of teachers' perceptions of possible and meaningful actions with programming. Therefore, selected findings in Study II are considered translatable into action possibilities for programming in grade 7-9 mathematics and technology, in line with the theoretical lens of affordances used in the thesis. In *6 Discussion*, findings from Study II are related to action possibilities and constraints identified in Studies III-V (which applied affordances as a theoretical lens) and used to further develop the conceptual model presented in *7 Concluding remarks*.

5.3 Study III

Humble, N. (2022). Teacher observations of programming affordances for K-12 mathematics and technology. *Education and Information Technologies*, *27*(4), 4887-4904. https://doi.org/10.1007/s10639-021-10811-w

Study III addresses teachers' perceptions of integrating programming in grade 7-9 mathematics and technology and their use of programming as a tool for teaching and learning in these subjects. Through focus group interviews, the study provides insights into what teachers perceive to be the action possibilities and constraints of different programming tools, and with using and integrating programming in grade 7-9 mathematics and technology. Data were collected from three focus groups, consisting of teachers in grade 7-9 mathematics and technology (further described in *4.1 Data collection*). A thematic analysis using the theoretical lens of affordances was applied to the collected data to identify themes of action possibilities and constraints with regard to the use and integration of programming in grade 7-9 mathematics and technology (further described in *4.2 Data analysis*).

The study shows that teachers perceive both action possibilities and constraints for the use and integration of programming in teaching and learning of grade 7-9 mathematics and technology. Teachers are generally positive towards the use of programming for teaching and learning activities but perceive constraints in its integration, including

insufficient planning from school leaders and other stakeholders for the integration. Furthermore, teachers believe that there is not enough time for them to learn programming and integrate it in their teaching and learning activities for mathematics and technology. Teachers also perceive the guidelines and instructions for integration to be unclear.

The study further shows that teachers perceive several action possibilities and constraints for different programming tools used. These action possibilities and constraints are summarised by the following categories: play, discovery, adaptation, control and freedom. Block programming and unplugged programming are mentioned as examples of tools that can provide an element of play within mathematics and technology, with both being highly accessible and enabling the inclusion of games in teaching and learning activities. Tangible technology using a block-programming interface is mentioned as an example of a tool that can facilitate the joy of discovery in teaching and learning, allowing students to experiment without being constrained by programming syntax. However, unplugged programming and block programming tools that are perceived to offer a too limited programming experience are not believed to facilitate discovery.

Block and tangible programming tools that can offer different levels of complexity are considered to provide an element of adaptation to teaching and learning activities. Textual programming that was considered too complex and unplugged programming that was considered too simple were not perceived to provide the element of adaptation. Block programming tools that allow a teacher to plan, create and monitor students' use are considered to provide control over the teaching and learning activities. Textual programming tools, which often lack this feature, are considered to constrain teachers'

control. However, textual programming tools that provide a complete programming experience are considered to provide the freedom that is often needed in teaching and learning activities for more advanced content, especially in mathematics. There is a concern among teachers that block programming tools constrain this freedom and prevent the development of a deeper understanding of programming by hiding code in graphical blocks.

Study III contributes to the thesis by examining teachers' perceptions of the action possibilities and constraints of using and integrating programming in teaching and learning in grade 7-9 mathematics and technology. The study further contributes to the thesis by identifying action possibilities and constraints of different programming tools in grade 7-9 mathematics and technology. Study III applied affordances as a theoretical lens. The scope of Study III, teachers' use and perceptions of integrating programming in teaching and learning for grade 7-9 mathematics and technology, aligns with the scope of the thesis. Therefore, the findings presented in the study are relevant for answering the aim and research questions of the thesis. In *6 Discussion*, the findings from Study III are related to those of the other included studies to further develop the identified action possibilities and constraints. The findings from Study III are further used to develop the conceptual model presented in *7 Concluding remarks*, which was first introduced in Study IV.

5.4 Study IV

Humble, N. (in third revision). A Conceptual Model of What Programming Affords Secondary School Courses in Mathematics and Technology.

Study IV addresses teachers' perceptions of the potential contributions of programming to grade 7-9 mathematics and technology. The interview study provides insights into what teachers perceive to be action possibilities and constraints of programming for grade 7-9 mathematics and technology. Data were collected through semi-structured interviews with teachers in grade 7-9 mathematics and technology who have experience in programming (further described in *4.1 Data collection*). A thematic analysis using the theoretical lens of affordances was applied to analyse the collected data and identify themes of action possibilities and constraints for programming in grade 7-9 mathematics and technology (further described in *4.2 Data analysis*).

The study shows that teachers perceive several action possibilities that programming can provide for grade 7-9 mathematics and technology but also a number of constraints. The identified action possibilities and constraints are summarised in the following categories: flexibility, creativity, efficiency, visualisation, fun, curiosity, play, holistic views, fearlessness and interdisciplinary collaborations. Study IV further shows that the identified action possibilities and constraints address three aspects of teaching and learning: 1) supporting subject content, 2)

facilitating engagement and motivation and 3) fostering developmental skills.

With regard to supporting subject content, teachers perceive that programming offers greater flexibility for teaching and learning by providing a new tool and context for problem solving and different perspectives on subject content. Compared to textual programming, block programming is perceived as constraining what can be achieved with programming. Programming is perceived to encourage creativity by offering multiple ways of looking at a problem using different programming tools. Compared to block programming, textual programming can be perceived as overwhelming by novices and therefore a constraint on creativity. Programming is perceived as a powerful tool for problem solving as well as for reinforcing routine and potentially boring practices that are efficient for teaching and learning. However, it is perceived that textual programming can reduce efficiency among students with writing difficulties because accurate spelling is important to avoid errors in code. Teachers perceive that graphical interfaces and programming libraries can be used as a visualisation tool in teaching and learning, such as for automation, construction, geometry and probability theory. However, potential constraints are perceived in textual programming as being too difficult and block programming as not always providing an intuitive programming interface.

Regarding engagement and motivation, teachers perceive programming to bring an element of fun to teaching and learning by offering new and tangible approaches to a subject. However, the tangible element is often missing in mathematics, which is perceived as a potential constraint. Programming is perceived to promote curiosity in teaching and learning, offering a new tool for sparking

interest in a subject. However, negative preconceptions about programming and programmers are perceived to be a constraint on the number and type of students for whom programming will spark interest. Programming is perceived to have the potential to make teaching and learning more playful. However, a perceived constraint is that block programming, compared to textual programming, is sometimes seen as being only for play.

Regarding fostering developmental skills, teachers perceive that programming provides holistic views on teaching and learning by situating a subject within a wider context, connecting subjects to the real world and helping to develop an understanding of how technology works. A perceived constraint for block programming, compared to textual programming, is that it is not used in a professional setting, limiting its usefulness outside of education. Programming is perceived to reduce the fear of making errors, which is useful in teaching and learning. Fear of making errors is perceived as a common problem, especially in mathematics, but handling errors is part of the programming mindset. A perceived constraint is that programming does not automatically foster fearlessness with regard to mistakes, rather it is a practice that must be actively practiced and fostered. Teachers perceive programming to be a tool for teaching and learning through which subjects such as mathematics, technology, language and social sciences can be combined for interdisciplinary knowledge development. A perceived constraint is that block programming, which is often most accessible for beginners, can be seen as childish by those with experience in programming.

Study IV contributes to the thesis by examining teachers' perceptions of the action possibilities and constraints of different programming tools, and with using programming for teaching and learning in grade

7-9 mathematics and technology. Study IV applied affordances as a theoretical lens during the planning, execution and presentation stages. The scope of Study IV, teachers' perceptions of the possible contributions of programming to grade 7-9 mathematics and technology, aligns with the scope of the thesis. Therefore, the findings presented in the study are relevant for answering the aim and research questions of the thesis.

The study further contributes to the thesis by providing an early version of a conceptual model, based in Norman's (2013; 1999) description of affordances and conceptual models, for the possible contributions of programming for grade 7-9 mathematics and technology. This version of the conceptual model includes action possibilities and constraints of programming for grade 7-9 mathematics and technology related to three aspects of teaching and learning: supporting subject content, facilitating engagement and motivation, and fostering developmental skills. In *6 Discussion*, the findings from Study IV are related to those of the other included studies to further develop the identified action possibilities and constraints. It is also used to further develop the conceptual model, presented in *7 Concluding remarks*, by specifying action possibilities and constraints (see summary in *6 Discussion*) and renaming 'developmental skills' to 'digital competence' for better accuracy and consistency with related research.

5.5 Study V

Humble, N., & Mozelius, P. (2022). Making programming part of teachers' everyday life – Programming affordances and constraints for K-12 mathematics and technology. *The International Journal of Information and Learning Technology*, Vol. ahead-of-print No. ahead-of-print. https://doi.org/10.1108/IJILT-03-2022-0069

Study V addresses how teachers use programming for teaching and learning in mathematics and technology. Through a document analysis, the study provides insights into the action possibilities and constraints that teachers perceive with programming for mathematics and technology. Data were collected from three instances of a professional development course on programming directed at grade 7-12 teachers in the subjects of mathematics and technology in the form of lesson plans created by the teachers for their final course assignment (further described in *4.1 Data collection*). Content analysis using the theoretical lens of affordances was applied to the collected data to identify codes of interest and develop meaningful categories to answer the study's aim and research questions (further described in *4.2 Data analysis*).

The study shows that teachers perceive several action possibilities and constraints in using programming for teaching and learning activities in grade 7-12 mathematics and technology. Action possibilities and constraints for teaching and learning activities were identified by

examining how teachers apply and reflect on programming within the collected lesson plans. The identified action possibilities for programming are summarised in the following categories: programming to support subject content, programming to foster motivation, and programming to develop general skills. The identified constraints for programming are summarised in the categories: organisational constraints for programming and student-related constraints for programming.

With regard to supporting subject content, programming is used to adapt the difficulty of subject content to suit different student groups and provide opportunities for repeated problem solving to facilitate learning. Programming activities are used and re-used for different subjects and provide a foundation for interdisciplinary collaborations and to visualise subject content such as mathematics and geometry. Programming is also used to increase accessibility by applying graphical-block programming tools. Programming is used to provide variety and creativity of subject content and give students the opportunity to apply their knowledge independently to find their own solutions. Programming is used to draw on the power of computing to address subject content that can be difficult to teach and learn, such as statistics, prime numbers and number sequences. It is also used to generalise subject content to other domains and build on students' prior knowledge. It can provide instant feedback on students' work through error messages and serve to complement the support and feedback provided by a teacher.

With regard to fostering motivation, multiple programming tools were used in the collected lesson plans and this variety was perceived to support students' understanding and facilitate engagement. Reflections in the lesson plans further indicate that programming is

perceived to be a fun activity that can boost both students' and teachers' motivation for mathematics and technology. The lesson plans also show that programming is used to incorporate problem-based learning in teaching and learning activities. This is also perceived to provide a real-life orientation to subject content and visible learning benefits for students. Programming was also used in the lesson plans to stimulate students' curiosity for mathematics and technology and develop a mindset of free reflections.

With regard to developing general skills that go beyond teaching and learning in mathematics and technology, programming is used to support students' in reflecting on their work and finding and correcting their own errors. It is also used to support students in deconstructing problems and assignments into smaller, more manageable parts that can be methodically addressed, as well as foster their independence and ability to develop their own solutions. Programming is used to foster holistic views of mathematics, technology and the outside world by demonstrating how it is used in society and connecting subject content to re-usable and generalised programming algorithms.

With regard to organisational constraints of programming, school hardware is perceived to be a potential constraint on teachers' use of programming. School computers do not always give teachers and students full system access and control and the freedom to use certain applications. A lack of time is perceived as another organisational constraint that limits what the teachers can do with programming. With regard to student-related constraints, wide disparities in prior knowledge of programming among students is perceived as a potential problem. Teaching and learning activities that rely on programming may be perceived as easy by some and too difficult by

others. Student resistance is perceived as another potential constraint for programming, since not all students will be excited by the new learning opportunity. The lesson plans also indicate that students' lack of focus and patience are perceived as potential constraints. However, the lesson plans also reflect that teachers perceive programming to be an activity that can support students in developing these traits.

Study V contributes to the thesis by examining the use of programming for teaching and learning activities in grade 7-12 mathematics and technology, as reflected by teachers in their lesson plans. The study further contributes to the thesis by identifying action possibilities and constraints for programming in teaching and learning activities of mathematics and technology. Study V applied affordances as a theoretical lens. The scope of the study was wider than that of this thesis and included mathematics teachers in grades 10-12. However, this is not considered an issue since no crucial differences in the study's results relevant to answering the thesis aim and research questions were identified between grade 7-9 and 10-12 teachers.

Although Study V did not use the conceptual model developed in Study IV, there are clear similarities between the categories of identified action possibilities and constraints in Study V and the three aspects of teaching and learning in the conceptual model of Study IV. In *6 Discussion*, findings from Study V are related to those of the other studies to further specify the action possibilities and constraints. Similar to Study IV, the category 'developing general skills' in Study V is renamed to 'developing digital competence'; as part of the further development of the conceptual model (presented in *7 Concluding remarks*) for better accuracy and consistency with related research.

6 Discussion

> It is simply not possible for me to put the amount of time into this that I think it requires. Where am I supposed to find time in an already full schedule?
>
> (Teacher in Study I)

In the studies included in this thesis, action possibilities and constraints can be identified that support the understanding of teachers' use and perceptions of programming for teaching and learning in grade 7-9 mathematics and technology. In this section, the findings from Studies I-V are discussed in relation to each other and previous research to summarise and further specify the identified action possibilities and constraints. This discussion also contributes to the development of the conceptual model, which was first introduced in Study IV and is presented in more detail in *7 Concluding remarks*.

Therefore, the discussion of action possibilities and constraints of programming for grade 7-9 mathematics and technology is related to the three aspects of teaching and learning included in the revised conceptual model in the thesis: supporting subject content, facilitating motivation and engagement, and developing digital competence. It should be noted that action possibilities and constraints are seldom isolated and should be understood in relation to each other. However, for the purpose of presenting the findings from the included studies in relation to previous research, action possibilities and constraints are discussed under two separate subheadings. The identified action possibilities and constraints are summarised in Table 5.

Table 5. Summary of action possibilities and constraints

	Support subject content	Facilitate motivation and engagement	Develop digital competence
Action possibilities	Variety Creativity Powerful Repeat Re-use Generalise Interdisciplinary collaborations Problem solving Less fear of errors Visualisation	Fun Playful Interesting Curiosity Adaptive Hands-on Discovery	Holistic views Deconstruction Problem-based learning Confidence Independence Error handling Technology understanding From consumer to producer
Constraints	Lack of tangible elements in subject Too difficult Too easy Too limited Prevent deeper learning Overwhelming Writing difficulties	Difficult to learn Play instead of learning Professional relevance Too childish	Application outside educational context Relevance Usefulness Potential misconceptions about skill development

6.1 Action possibilities

Previous research shows positive perceptions of the integration of programming in education by both Swedish (Vinnervik, 2022) and Finnish teachers (Korhonen et al., 2022). According to Vinnervik (2022), the integration of programming is perceived as valid, timely and relevant for today's society. This is also supported by a study by Korhonen et al. (2022), which shows that programming is perceived as a relevant and useful skill that schools should provide for their students. This is further emphasised by the included studies in the thesis. Teachers are generally positive towards programming for teaching and learning in grade 7-9 mathematics and technology and perceive programming to be both fun and easily linked to subject content (Studies I, III, IV, V).

The included studies show action possibilities for programming to support subject content in grade 7-9 mathematics and technology. In Studies IV and V, it is highlighted that programming can be used to provide variety and creativity in teaching and learning activities by offering different ways of looking at subject content and enabling students to find their own solutions to problems. Similar findings have been presented in previous research. According to Vinnervik (2022), teachers perceive programming to provide variation in teaching and learning. Korhonen et al. (2022) report that teachers perceive programming to potentially support them in approaching their teaching from different perspectives. Study V also suggests that programming should allow students to use their knowledge in new ways and that the use of multiple programming tools can support this.

According to Kilhamn, Bråting and Rolandsson (2021), teachers perceive programming to be a powerful tool for problem solving in mathematics and other subjects. This notion is supported by the included studies. Programming is used to repeat, re-use and generalise problem-solving strategies for different subject content and interdisciplinary collaboration (Study V). Programming is further used for problem solving and to reduce the fear of making errors, especially in mathematics, inspired by error handling in programming (Study IV). The included studies also show that teachers use programming to visualise subject content, which shows a clear connection to the early educational adaptation of Papert's Logo (Stager, 2016). Programming is used for visualisation in mathematics (Study V), geometry, automation, construction and probability theory through both the textual programming library Turtle and block programming (Study IV). The material collected in Study II further highlights that block programming is commonly used for visualisation in mathematics and technology.

The included studies show action possibilities for programming to facilitate motivation and engagement in grade 7-9 mathematics and technology. This is also a common focus for research on programming in K-12 education (Szabo et al., 2019). Programming is perceived as a fun activity (Study I) that can introduce a playful element to teaching and learning and provide interest, curiosity and motivation for both students and teachers (Studies IV, V). The potential for programming to increase engagement in subject content by being perceived as fun and interesting is also supported in previous research (Korhonen et al., 2022; Kilhamn, Bråting & Rolandsson, 2021). Study III further highlights that for some programming tools the level of complexity can be adapted. This could allow more students to enjoy the tools,

regardless of their previous knowledge and experience in programming.

Vinnervik (2022) also highlights that programming is perceived by teachers to provide more hands-on exploration of subject content. This is supported by the studies included in the thesis and in Study IV programming is described as fun and providing new and tangible ways of teaching. The hands-on approach is especially apparent in the controlling of a tangible object, such as with the use of a Micro:bit or an Arduino (Cederqvist, 2022). Study II shows that the use of tangible programming is relatively common in the collected material. It is used in both mathematics and technology, across all years (grades) of K-12 education, mainly for controlling physical objects (Study II). The included studies show that the tangible elements of programming can facilitate motivation and engagement. Study III shows that programming with tangible technology and tinkering can facilitate the joy of discovery among students.

The included studies show action possibilities for programming to develop digital competence in grade 7-9 mathematics and technology. These are skills that go beyond specific teaching and learning situations and are transversal and useful across many contexts. Teachers induce a holistic view of teaching and learning by connecting different aspects of subject content, the outside world and other subject fields through interdisciplinary collaborations, re-usable programming algorithms, deconstructions of problems and instructions and problem-based learning with visible benefits (Studies IV, V). This can be compared to teacher perceptions presented in a study by Kilhamn, Bråting and Rolandsson (2021), in which programming is perceived to support problem solving and

deconstruction, and in Vinnervik (2022), in which programming is perceived as relevant across several subjects.

Programming is further used in the included studies to encourage confidence and independence among students. This is achieved through supporting students to reflect on their work, progress independently, handle errors and understand how technology works (Studies IV, V). This can be compared to Korhonen et al. (2022), in which teachers perceive programming as having the potential to allow students to develop other useful skills, such as logical thinking. Study V further highlights that programming can be used to support students to move from a consumer standpoint on technology to having a producer relationship with technology.

6.2 Constraints

Previous research shows several constraints for integrating programming in K-12 education. Szabo et al. (2019) highlight that several studies have reported the need to support teachers in timetabling, workload and building confidence during the integration of programming. Korhonen et al. (2022) also highlight that integration planning is perceived as unclear and that teachers lack experience and skill in programming. These concerns are also highlighted in the studies included in the thesis. Studies I and III show that teachers perceive school leaders' planning for the integration to be unclear, as well as a lack of guidance, directives and instructions for the use of programming in their subjects. Teachers further express that they perceive a lack of time for integration (Studies I, III, V).

Vinnervik (2022) highlights that teachers do not feel sufficiently prepared for the integration, expect more active school leadership and have concerns regarding IT infrastructure, support and professional development. These concerns are also highlighted in the included studies. Study V shows that school computers can be a potential constraint for integrating programming, since teachers and students often do not have full system access and cannot use certain applications. Furthermore, negative preconceptions about programming and programmers could affect which students are interested in the integration (Study IV). Other potential constraints on the integration are students' prior knowledge in programming, potential resistance towards programming, focus and patience (Study V). However, Study V also highlights that these constraints can be addressed through programming.

The included studies show constraints for programming to support subject content in grade 7-9 mathematics and technology. Previous research highlights that programming can increase variation and support hands-on exploration of subject content (Vinnervik, 2022). Study IV show that hands-on exploration could be a potential constraint for subject content in mathematics, which often lacks tangible elements compared to subject content in technology. The studies included in the thesis also show that there are potential constraints for several programming tools to support variation in teaching and learning. Textual programming is perceived as being too complex (Study III), more so than block and unplugged programming (Study I) and not adaptable (Study III). Block programming is perceived as limited compared to textual programming (Study I) and a constraint on programming freedom (Study III). Unplugged programming is perceived as being too simple, not adaptable (Study III) and mainly for younger children (Study I).

Previous research further highlights that programming can be used as a powerful tool for problem solving (Kilhamn, Bråting & Rolandsson, 2021). The included studies support the notion that textual programming is used and perceived as a powerful tool for problem solving. However, Study IV also shows that textual programming could be perceived as overwhelming by novices and potentially constrain students with writing difficulties. Study II also indicates that textual programming could be too difficult for some students in grades 7-9 since it is mainly used in grades 10-12 and often focused on the practice of writing code. Block programming could be considered a good compromise between textual programming and unplugged programming in terms of complexity. However, by hiding code in blocks, block programming could potentially prevent a deeper

understanding of programming (Study III) and thereby limit its usefulness as a tool.

The included studies show constraints for programming to facilitate motivation and engagement in grade 7-9 mathematics and technology. As stated in previous research, programming in the K-12 context is often studied with regard to facilitating engagement and motivation among students (Szabo et al., 2019). However, as in the case of supporting subject content, constraints on different programming tools to facilitate motivation and engagement have been identified in the included studies. Textual programming is often considered more difficult for novices to learn compared to other programming tools (Studies I-IV), which could be a constraint on facilitating motivation and engagement among students.

Kilhamn, Bråting and Rolandsson (2021) highlight that programming is perceived as fun and interesting with the potential to engage students. This is also supported by the included studies. Unplugged and block programming are two tools that are described as fun by teachers (Studies I, IV). Block programming is often perceived as more accessible than other programming tools, but this comes at the potential cost of it only being used for play (Study IV). Study IV highlights that block programming, compared to textual programming, is not used in professional settings, which could affect students' motivation to engage in using the tools for learning. Both unplugged and block programming could also be perceived as childish by students (Studies I, III), which could affect their effectiveness at facilitating motivation and engagement for teaching and learning.

The included studies show constraints for programming to develop digital competence in grade 7-9 mathematics and technology.

Previous research highlights programming's relevance for today's society and a number of subjects besides mathematics and technology (Vinnervik, 2022). This notion is supported by the included studies. However, Studies I and IV show that relevance to society and future work life may be a constraint of block and unplugged programming since, unlike textual programming, they are not used in professional settings. Block programming could still be considered a good option for interdisciplinary collaboration since it is often perceived as more accessible (Study IV). This comes at the potential cost of it being perceived as childish by those more experienced in programming and used more for play than learning (Study IV).

Previous research further highlights that programming can be used to develop other useful skills (Korhonen et al., 2022) such as problem solving, deconstruction, testing and debugging (Kilhamn, Bråting & Rolandsson, 2021). The included studies support the argument that programming can be used to develop skills that are useful outside grade 7-9 mathematics and technology. However, as highlighted before, the use of certain programming tools outside the educational context could potentially affect the relevance and usefulness of developed skills (Studies I, IV). The included studies further support the argument that programming can be used for problem solving, deconstruction, testing and debugging. However, it is a possible misconception that these skills can be developed by simply engaging in programming. As highlighted in Study IV, a programmer mindset can reduce the fear of making errors in mathematics, but this is not developed automatically and needs to be actively addressed and fostered by teachers through programming (Study IV).

7 Concluding remarks

> You shouldn't limit yourself to one way of solving programming assignments or to one specific programming language.
>
> (Teacher in Study IV)

The aim of this thesis has been to examine teachers' use and perception of programming for teaching and learning in grade 7-9 mathematics and technology. This has been investigated through five separate but interrelated studies (Studies I-V), which have been presented and discussed in this thesis. The thesis shows that teachers mainly use programming to address three aspects of teaching and learning: to support subject content, to facilitate motivation and engagement, and to develop digital competence. Further, the included studies have highlighted that there are several action possibilities and constraints that affect how teachers use and perceive programming in relation to these aspects of teaching and learning. All this can be summarised using a conceptual model of the uses of programming in grade 7-9 mathematics and technology from the perspective of teachers (Figure 3).

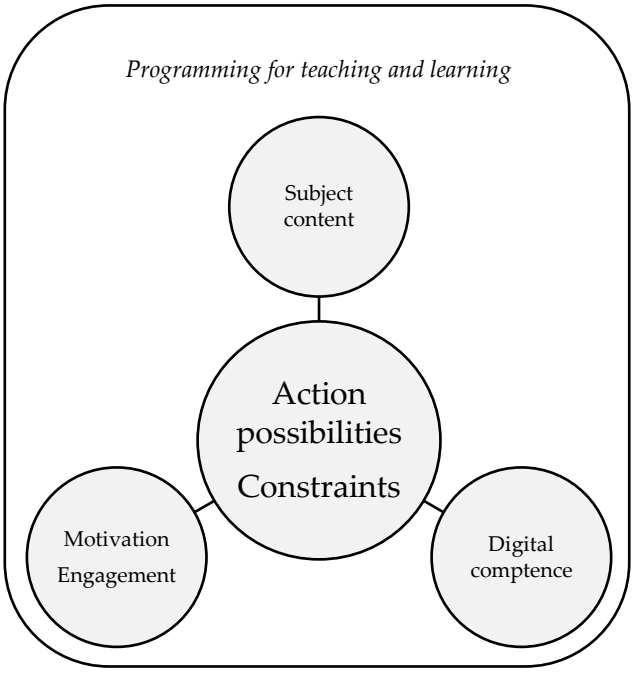

Figure 3. Teachers' conceptual model of programming

The most important contribution of the thesis project is in identifying the action possibilities and constraints of programming for grade 7-9 mathematics and technology; these were identified using the theoretical lens of affordances and tied together within the conceptual model. The findings presented in this thesis provide new insights into the study object, teachers' use and perceptions of programming for teaching and learning in grade 7-9 mathematics and technology, which are further described in *7.1 Theoretical contribution*. The findings of the thesis also provide insights and suggestions for the overall objective of the thesis: to address the issue of teachers lacking guidance, knowledge and skills to see the potentials and possible pitfalls of programming for teaching and learning. The findings of the thesis should have practical implications for the use and integration of

programming in K-12 education and how teachers are supported and prepared for the use of programming in their teaching and learning activities. This is further described in *7.2 Implications for practice*.

However, the thesis and included Studies I-V have also highlighted challenges and the need for future research on the use and integration of programming in K-12 education. The most salient challenge that has emerged from this research is: how should teachers address the action possibilities and constraints of programming for teaching and learning when there seems to be no individual programming tool that checks all boxes? This is an important question that requires more research. However, guided by the findings presented in this thesis and inspired by the professional setting of programming, the author suggests a multi-tool approach to programming be applied in K-12 education. In a professional setting, programming tools are chosen and combined based on the desired outcome. A similar mindset should be applied for programming in the educational setting, with programming tools chosen and combined based on needs to support, facilitate and develop teaching and learning activities. This work could be guided by the findings presented in this thesis to take advantage of individual action possibilities of programming while potentially avoiding the constraints.

7.1 Theoretical contribution

The study object of the thesis was teachers' use and perceptions of programming for teaching and learning in grade 7-9 mathematics and technology. The knowledge object of the thesis was to develop knowledge about teachers' perceived affordances (action possibilities) and constraints in using programming for grade 7-9 mathematics and technology. Through the included studies of the thesis, it has been highlighted that teachers mainly use and perceive programming in relation to three aspects of teaching and learning: to support subject content, facilitate motivation and engagement, and develop digital competence. The included studies have further identified several action possibilities and constraints for programming to address these aspects of teaching and learning in grade 7-9 mathematics and technology. Guided by the theory of affordances, these have been combined in a conceptual model (Figure 3) of the uses of programming in grade 7-9 mathematics and technology. To the author' knowledge, a conceptual model that highlights action possibilities and constraints of programming for K-12 education has not been presented in previous research and should be considered an important contribution to the understanding of the study object.

The thesis has also provided new understanding of the study object by applying the theoretical lens of affordances, which appears less commonly in related research than computational thinking. Previous research that applies affordances to the study of programming in the K-12 setting does exist. However, these studies are often more focused on specific technologies or programming tools, such as tangible programming, while more general studies of programming for

teaching and learning tend to apply the theory of computational thinking. This thesis has shown that the theory of affordances can be a valid alternative to computational thinking in research on the study object. As discussed in *3 Thesis journey*, affordances can support in developing an understanding of teachers' use of programming since it highlights action possibilities and constraints related to what teachers perceive as important or meaningful, rather than relating their use of programming to the facets of a computational thinking framework that may or may not resonate with their perspectives.

7.2 Implications for practice

The starting point and overall objective of the thesis was to address the issue of teachers lacking guidance, knowledge and skills to see the potentials and possible pitfalls of programming for teaching and learning. The thesis and included studies provide practical examples of action possibilities and constraints for programming, summarised in a conceptual model (Figure 3) that relate these findings to three aspects of teaching and learning: supporting subject content, facilitating motivation and engagement, and developing digital competence. These findings are based on teachers' use and perceptions of programming for mathematics and technology in grades 7-9 and can be used to address the overall objective of the thesis. The findings of the thesis should have implications for how teachers use programming in K-12 education as well as how other stakeholders support teachers in the ongoing integration of programming.

As highlighted in the thesis and included studies, programming can be used to address different aspects of teaching and learning (subject content, motivation and engagement, and digital competence). But programming presents both action possibilities and constraints in trying to achieve this. This should, to a large extent, be addressed, for example, in the design of professional development courses on programming or by preparing future teachers during teacher education. If the support and guidance provided for teachers does not include the versatility of programming (which has been shown in this thesis), teachers will likely not have access to the full range of opportunities that programming can provide or be aware of the constraints.

Professional development courses in programming and teacher education should incorporate multiple programming tools and approaches that can be used in the K-12 setting. This would enable teachers to make conscious decisions on the tools best suited for their teaching and learning activities. Furthermore, programming should be presented and taught to teachers in relation to how it can support subject content, facilitate motivation and engagement, and develop digital competence, as well as the action possibilities and constraints of programming in achieving this. This would enable teachers to make conscious decisions on the use of programming in their teaching and learning activities, while still being mindful that both action possibilities and constraints can affect teaching and learning outcomes.

Below are some examples of how programming can be used by teachers in the process of, or considering, integrating programming in their teaching and learning activities related to the findings of the thesis. When using programming to support subject content with students that already have an interest or are engaged in programming, textual programming tools such as Python can be a good choice. Professional programming is most commonly associated with textual programming and is often considered more powerful than tools such as block programming and unplugged programming. This can be especially important if the supported subject content is complex or advanced.

However, if programming is mainly intended to facilitate motivation and engagement with students that are less familiar with programming, block programming tools such as Scratch or unplugged programming may be a more effective choice. Block and unplugged programming are often perceived as fun, engaging and more accessible than textual programming since they do not require accurate

spelling and syntax. Block and unplugged programming both often also include a game-based learning approach that students may find appealing.

If programming is being used to develop digital competence of students with varied experience of programming, tangible programming can be a good choice. Tangible programming tools such as Micro:bit are often considered exciting since they incorporate hands-on tinkering. Tangible programming also commonly offers a choice between a textual or block programming interface, which makes it suitable for students with varied experience of programming. Furthermore, tangible programming has obvious relevance to the real world since it provides students with the opportunity to explore how technology works.

The examples provided above should be viewed as potential applications for programming in K-12 education. Some students may find textual programming too difficult, while others may find the challenge engaging. Some students may find the playful nature of block and unplugged programming motivating, while others may find it childish. Teachers and students have different backgrounds, preference and experience in programming, which should be taken into consideration. It is important to remember that programming is versatile and includes many different types of tools and approaches that can be used to address different aspects of teaching and learning and to accommodate students (and teachers) with different experiences of programming.

8 Limitations and future research

> I'm excited to test this lesson plan … I believe that it could be really fun.
>
> (Teacher in Study V)

The integration of programming in education is an ongoing process and will likely continue with developments in educational technology and programming tools. It will be up to the readers of this thesis to determine to what extent the presented findings stand the test of time. Therefore, the study object addressed in this thesis is and should be an ongoing endeavour for research. Not all teachers affected by the change in curriculum will integrate programming in their teaching and learning activities and are unlikely to show interest in participating in studies or professional development courses on programming. This was not the case for the studies included in the thesis, in which most teachers were positive towards programming and made efforts to implement it in their subjects. As such, the absence of critical voices could be considered a limitation of the thesis.

Therefore, future research and professional development should strive to find ways to reach teachers who are sceptical towards programming and choose not to implement it in their subjects. Highlighting their perspectives would enrich the understanding of teachers' use and perceptions of programming for teaching and learning in grade 7-9 mathematics and technology. It could further support the design of professional development courses on programming and encourage more teachers to participate.

Lastly, a challenge that many teachers highlighted for the ongoing integration of programming, both in the conducted studies and during the professional development courses, is the lack of time that many of them experience. While some teachers report good support from school leadership in providing time for professional development, this has not been the case for all. A specific challenge that teachers have mentioned is that students with limited experience of programming first need to be taught the basics before it can be used for teaching and learning in their subject, which is time consuming. The author hopes to continue to explore whether the teaching of programming can be supported and made more effective with game-based learning.

An early version of an educational game for programming has already been developed and evaluated by teachers (Humble, Mozelius & Sällvin, 2021b) and students (Humble & Mozelius, 2022c). A new and updated version of this game is planned for development and further testing in education for 2023-2024 through the project SPEDAT (SPEl för DAtalogiskt Tänkande [Games for computational thinking]). The design of the game is further guided by the work and research in the Erasmus+ project Gaming4Coding, in which a programming game with inclusive design will be developed. This project has already investigated the criteria for an inclusive design (Mozelius et al., 2022)

and developed a potential framework for the design of the game (Mozelius & Humble, 2022). More studies are planned for the years to come on the design, development, evaluation and learning outcomes of the games.

References

Admiraal, W., Post, L., Guo, P., Saab, N., Makinen, S., Rainio, O., Vuori, J., Bourgeois, J., Kortuem, G., & Danford, G. (2019). Students as Future Workers: Cross-border Multidisciplinary Learning Labs in Higher Education. *International Journal of Technology in Education and Science, 3*(2), 85-94. Retrieved July 19, 2022 from https://www.learntechlib.org/p/207262/

Aiello, L. C. (2016). The multifaceted impact of Ada Lovelace in the digital age. *Artificial Intelligence, 235*, 58-62. https://doi.org/10.1016/j.artint.2016.02.003

Askling, B. (2004). Några tankar om utbildningsvetenskap och pedagogisk forskning [Some thoughts on educational science and research]. *Pedagogisk forskning i Sverige, 9*(1), 53-57. Retrieved October 23, 2022 from https://open.lnu.se/index.php/PFS/article/view/1173/1022

Assarroudi, A., Heshmati Nabavi, F., Armat, M. R., Ebadi, A., & Vaismoradi, M. (2018). Directed qualitative content analysis: The description and elaboration of its underpinning methods and data analysis process. *Journal of Research in Nursing, 23*(1), 42-55. https://doi.org/10.1177%2F1744987117741667

Balacheff, N., Ludvigsen, S., De Jong, T., Lazonder, A., & Barnes, S. A. (2009). Technology-enhanced learning. In N. Balacheff, S. Ludvigsen, T. De Jong, A. Lazonder, & S. Barnes (Eds.). *Technology-Enhanced Learning: Principles and Products.* Springer.

Balanskat, A., & Engelhardt, K. (2015). Computing our future: Computer programming and coding – Priorities, school curricula and initiatives across Europe. European Schoolnet. Retrieved July 19, 2022 from http://www.eun.org/documents/411753/817341/Computing+our+future_final_2015.pdf

Bassey, M. (1981). Pedagogic research: On the relative merits of search for generalisation and study of single events. *Oxford Review of Education, 7*(1), 73-94. https://doi.org/10.1080/0305498810070108

Bloomfield, B. P., Latham, Y., & Vurdubakis, T. (2010). Bodies, technologies and action possibilities: When is an affordance?. *Sociology, 44*(3), 415-433. https://doi.org/10.1177%2F0038038510362469

Bocconi, S., Chioccariello, A. & Earp, J. (2018). The Nordic approach to introducing Computational Thinking and programming in compulsory education. *Report prepared for the Nordic@BETT2018 Steering Group*. https://doi.org/10.17471/54007

Bower, M., & Sturman, D. (2015). What are the educational affordances of wearable technologies?. *Computers & Education, 88*, 343-353. https://doi.org/10.1016/j.compedu.2015.07.013

Braun, V., & Clarke, V. (2012). Thematic analysis. In H. Cooper, P. M. Camic, D. L. Long, A. T. Panter, D. Rindskopf, & K. J. Sher (Eds.). *APA Handbook of Research Methods in Psychology, Vol. 2. Research Designs: Quantitative, Qualitative, Neuropsychological, and Biological* (pp. 57-71). American Psychological Association. https://psycnet.apa.org/doi/10.1037/13620-004

Bryman, A. (2016). *Social research methods*. Oxford University Press.

Bråting, K., & Kilhamn, C. (2021). Exploring the intersection of algebraic and computational thinking. *Mathematical Thinking and Learning, 23*(2), 170-185. https://doi.org/10.1080/10986065.2020.1779012

Bråting, K., Kilhamn, C., & Rolandsson, L. (2020). Integrating programming in Swedish school mathematics: Description of a research project. In *MADIF12: The twelfth research seminar of the Swedish Society for Research in Mathematics Education* (pp. 101-110). Retrieved July 19, 2022 from http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-410946

Cederqvist, A. M. (2022). Designing and coding with BBC micro:bit to solve a real-world task – A challenging movement between contexts. *Education and Information Technologies 27*, 5917-5951. https://doi.org/10.1007/s10639-021-10865-w

Clarke, V. & Braun, V. (2017). Thematic analysis. *The Journal of Positive Psychology, 12*(3), 297-298. http://dx.doi.org/10.1080/17439760.2016.1262613

Denzin, N. K. (2015). Triangulation. *The Blackwell Encyclopedia of Sociology*. https://doi.org/10.1002/9781405165518.wbeost050.pub2

Dewing, J. (2002). From ritual to relationship: A person-centred approach to consent in qualitative research with older people who have a dementia. *Dementia, 1*(2), 157-171. https://doi.org/10.1177%2F147130120200100204

Enders, T., Hediger, V., Hieronimus, S., Kirchherr, J. W., Klier, J., Schubert, J., & Winde, M. (2019). Future skills: Six approaches to close the skill gap. World Government Summit. Retrieved July 19, 2022 from https://www.mckinsey.com/~/media/mckinsey/industries/public%20and%20social%20sector/our%20insights/closing%20the%20future%20skills%20gap/wgs-future-skills-jan-28-2019.pdf

Ferrari, A., & Punie, Y. (2013). DIGCOMP: A framework for developing and understanding digital competence in Europe. Joint Research Centre, Institute for Prospective Technological Studies. https://data.europa.eu/doi/10.2788/52966

Flick, U. (2004). Triangulation in qualitative research. In U. Flick, E. von Kardorff, & I. Steinke (Eds). *A companion to qualitative research* (pp. 178-183). Sage.

Floyd, S. P. (2019, February). Historical high school computer science curriculum and current K-12 initiatives. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education* (pp. 1287-1287). https://doi.org/10.1145/3287324.3293772

Fransson, K., & Lundgren, U. P. (2003). *Utbildningsvetenskap – ett begrepp och dess sammanhang* [Educational science – a concept and its context]. Vetenskapsrådet [The Swedish Research Council]. Retrieved October 23, 2022 from https://www.vr.se/analys/rapporter/vara-rapporter/2003-03-03-utbildningsvetenskap.html

Gibson, J. J. (2015). *The Ecological Approach to Visual Perception*. Psychology Press.

Grover, S., & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. *Educational researcher, 42*(1), 38-43. https://doi.org/10.3102%2F0013189X12463051

Guba, E. G. (1981). Criteria for assessing the trustworthiness of naturalistic inquiries. *Ectj, 29*(2), 75-91. https://doi.org/10.1007/BF02766777

Heintz, F., Mannila, L., Nordén, L. Å., Parnes, P., & Regnell, B. (2017, November). Introducing programming and digital competence in Swedish K-9 education. In *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives* (pp. 117-128). Springer, Cham. https://doi.org/10.1007/978-3-319-71483-7_10

Hsieh, H. F., & Shannon, S. E. (2005). Three approaches to qualitative content analysis. *Qualitative Health Research, 15*(9), 1277-1288. https://doi.org/10.1177%2F1049732305276687

Humble, N. (2019). Developing computational thinking skills in K-12 education through block programming tools. In *Proceedings of the 12th annual International Conference of Education, Research and Innovation 2019 (ICERI 2019),* (pp. 4865-4873). The International Academy of Technology, Education and Development.

Humble, N. (2020). Using textual programming tools to develop computational thinking skills in K-12 education. In *Proceedings of the 12th annual International Conference on Education and New Learning Technologies (EDULEARN 2020),* (pp. 7188-7195). The International Academy of Technology, Education and Development.

Humble, N., & Mozelius, P. (2019). Teacher perception of obstacles and opportunities in the integration of programming in K-12 settings. In *Proceedings of the 11th International Conference on Education and New Learning Technologies (EDULEARN 2019),* (pp. 350-356). The International Academy of Technology, Education and Development.

Humble, N., & Mozelius, P. (2021, October). Enhancing Pedagogy to Andragogy in the Redesign of Teacher Training Courses on Programming. In *Proceedings of the 20th European Conference on e-Learning (ECEL 2021),* (pp. 210-217). Academic Conferences and Publishing International Limited.

Humble, N., & Mozelius, P. (2022a, May). Content analysis or thematic analysis: Similarities, differences and applications in qualitative research. In *Proceedings of the 21st European Conference on Research Methodology for Business and Management Studies (ECRM 2022), 21*(1) 76-81. Academic Conferences and Publishing International Limited. Retrieved July 19, 2022 from https://papers.academic-conferences.org/index.php/ecrm/article/view/316/320

Humble, N., & Mozelius, P. (2022b). Content Analysis or Thematic Analysis: Doctoral Students' Perceptions of Similarities and Differences. *Electronic Journal of Business Research Methods, 20*(3), 89-98. https://doi.org/10.34190/ejbrm.20.3.2920

Humble, N., & Mozelius, P. (2022c, September). Refurbishing the educational escape room for programming: Lowering the threshold and raising the ceiling. In *Proceedings of the 16th European Conference on Games Based Learning (ECGBL 2022)*, *16*(1) 280-289. Academic Conferences and Publishing International Limited. Retrieved October 28, 2022 from https://papers.academic-conferences.org/index.php/ecgbl/article/view/603

Humble, N., Mozelius, P., & Sällvin, L. (2019a). Teacher challenges and choice of programming tools for teaching K-12 technology and mathematics. In *Proceedings of the International Conference on Education and New Developments (END 2019),* 1, (pp. 431-435). inScience Press. Retrieved July 19, 2022 from http://end-educationconference.org/wp-content/uploads/2019/06/Education-and-New-Developments_2019_Vol_I.pdf

Humble, N., Mozelius, P., & Sällvin, L. (2019b, November). On the Role of Unplugged Programming in K-12 Education. In *Proceedings of the 18th European Conference on E-Learning (ECEL 2019),* (pp. 224-230). Academic Conferences and Publishing International Limited.

Humble, N., Mozelius, P., & Sällvin, L. (2020). The introduction of programming in K-12 technology and mathematics: Teacher choice of programming tools and their perceptions of challenges and opportunities. In M. Carmo (Eds.). *Education Applications & Developments V* (pp. 117-126). inScience Press. Retrieved July 19, 2022 from https://insciencepress.org/education-applications-developments-v/

Humble, N., Mozelius, P., & Sällvin, L. (2021a). Olika programmeringsverktyg och lärares fortbildning: Hinder och möjligheter. In K. Bråting, C. Kilhamn, & L. Rolandsson (Eds.). *Programmering i skolmatematiken: Möjligheter och utmaningar*. Studentlitteratur.

Humble, N., Mozelius, P., & Sällvin, L. (2021b, September). You can't escape learning, but maybe you can get out of the room! – Game-based learning for programming education. In *Proceedings of the 15th European Conference on Game Based Learning (ECGBL 2021),* (pp. 359-367). Academic Conferences and Publishing International Limited.

Humble, N., Sällvin, L., & Mozelius, P. (2019, November). Programmering i matte och teknik. *Pedagogiska Magasinet*. Lärarförbundet.

Kaplan, B., & Maxwell, J. A. (2005). Qualitative research methods for evaluating computer information systems. In J. G. Anderson, & C. E. Aydin (Eds.). *Evaluating the organizational impact of healthcare information systems* (pp. 30-55). Springer. https://doi.org/10.1007/0-387-30329-4_2

Kilhamn, C., Bråting, K., & Rolandsson, L. (2021). Teachers' arguments for including programming in mathematics education. In *NORMA 20, The Ninth Nordic Conference on Mathematics Education, Oslo, Norway, 1-4 June 2021* (pp. 169-176). Svensk förening för MatematikDidaktisk Forskning-SMDF.

Kirkwood, A., & Price, L. (2014). Technology-enhanced learning and teaching in higher education: What is 'enhanced' and how do we know? A critical literature review. *Learning, Media and Technology, 39*(1), 6-36. https://doi.org/10.1080/17439884.2013.770404

Korhonen, T., Salo, L., Laakso, N., Seitamaa, A., Sormunen, K., Kukkonen, M., & Forsström, H. (2022). Finnish teachers as adopters of educational innovation: Perceptions of programming as a new part of the curriculum. *Computer Science Education*, 1-23. https://doi.org/10.1080/08993408.2022.2095595

Kurian, J. C., John, B., & Lang, A. (2021, December). Redesigning Learning Spaces During a Pandemic. In *Proceedings of the 2021 AIS SIGED International Conference on Information Systems Education and Research*, 13. Retrieved October 24, 2022 from https://aisel.aisnet.org/siged2021/13

Lee, J. A. N. (2004). History of computing in education. In Impagliazzo, Lee J. A. N. (Eds.), *History of Computing in Education. 2004. IFIP Advances in Information and Communication Technology*, vol. 145. Springer. https://doi.org/10.1007/1-4020-8136-7_1

Leonardi, P. M. (2013). When Does Technology Use Enable Network Change in Organizations? A Comparative Study of Feature Use and Shared Affordances. *MIS Quarterly, 37*(3), 749–775. Retrieved July 19, 2022 from http://www.jstor.org/stable/43825998

Liamputtong, P. (2011). *Focus Group Methodology: Principle and Practice*. Sage.

Lodi, M., & Martini, S. (2021). Computational thinking, between Papert and Wing. *Science & Education, 30*(4), 883-908. https://doi.org/10.1007/s11191-021-00202-5

Majchrzak, A., & Markus, M. L. (2012). Technology affordances and constraints in management information systems (MIS). In E. Kessler (Ed.). *Encyclopedia of Management Theory*. Sage.

Moore, T. P., McArthur, M., & Noble-Carr, D. (2018). More a marathon than a hurdle: Towards children's informed consent in a study on safety. *Qualitative Research, 18*(1), 88-107. https://doi.org/10.1177%2F1468794117700708

Mozelius, P., & Hoff, C. (2019). The Introduction of Programming in Compulsory School: A Multi-Stakeholder Perspective. In *Proceedings of the 12th annual International Conference of Education, Research and Innovation (ICERI 2019),* (pp. 2032-2040). The International Academy of Technology, Education and Development.

Mozelius, P., & Humble, N. (2020). Lessons learnt from teacher professional development in programming. In *Proceedings of the 14th International Technology, Education and Development Conference (INTED 2020),* pp. 3903-3908. The International Academy of Technology, Education and Development.

Mozelius, P., & Humble, N. (2022). Design Factors for an Educational Game where Girls and Boys Play Together to Learn Fundamental Programming. In *EAI ArtsIT 2022, 11.*

Mozelius, P., Humble, N., Sällvin, L., Öberg, L. M., Pechuel, R., & Fernández-Manjón, B. (2022). How to get the girls Gaming: A Literature Study on Inclusive Design. In *Proceedings of the 16th European Conference on Games Based Learning (ECGBL 2022), 16*(1), 396-402. Academic Conferences and Publishing International Limited. Retrieved October 28, 2022 from https://papers.academic-conferences.org/index.php/ecgbl/article/view/658

Norman, D. (2013). *The design of everyday things: Revised and expanded edition*. The MIT Press.

Norman, D. A. (1999). Affordance, conventions, and design. *interactions, 6*(3), 38-43. https://doi.org/10.1145/301153.301168

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. https://doi.org/10.1080/20004508.2019.1627844

Osmundsen, K. S., Meske, C., & Thapa, D. (2022). Familiarity with digital twin totality: Exploring the relation and perception of affordances through a Heideggerian Perspective. *Information Systems Journal*. https://doi.org/10.1111/isj.12383

Papavlasopoulou, S., Giannakos, M. N., & Jaccheri, L. (2017, April). Reviewing the affordances of tangible programming languages: Implications for design and practice. In *2017 IEEE Global Engineering Education Conference (EDUCON),* (pp. 1811-1816). IEEE. https://doi.org/10.1109/EDUCON.2017.7943096

Papert, S. (1993). *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books.

Psycharis, S., & Kallia, M. (2017). The effects of computer programming on high school students' reasoning skills and mathematical self-efficacy and problem solving. *Instructional Science, 45*(5), 583-602. https://doi.org/10.1007/s11251-017-9421-5

Pörn, R., Hemmi, K., & Kallio-Kujala, P. (2021). Inspiring or Confusing—A Study of Finnish 1-6 Teachers' Relation to Teaching Programming. *LUMAT: International Journal on Math, Science and Technology Education, 9*(1), 366-396. https://doi.org/10.31129/LUMAT.9.1.1355

Resnik, D. B. (2020). What is ethics in research & why is it important. National Institute of Environmental Health Sciences. Retrieved July 19, 2022 from https://www.niehs.nih.gov/research/resources/bioethics/whatis/index.cfm

The Royal Society. (2017). After the reboot: Computing education in UK schools. Policy Report. Retrieved July 19, 2022 from https://royalsociety.org/-/media/policy/projects/computing-education/computing-education-report.pdf

Rubio, M. A., Romero-Zaliz, R., Mañoso, C., & Angel, P. (2015). Closing the gender gap in an introductory programming course. *Computers & Education, 82*, 409-420. https://doi.org/10.1016/j.compedu.2014.12.003

Sen A., & Leong C. K. C. (2020). Technology-Enhanced Learning. In A. Tatnall (Eds.). *Encyclopedia of Education and Information Technologies*. Springer. https://doi.org/10.1007/978-3-319-60013-0_72-1

Shaw, I. F. (2003). Ethics in qualitative research and evaluation. *Journal of Social Work, 3*(1), 9-29. https://doi.org/10.1177%2F1468017303003001002

Shenton, A. K. (2004). Strategies for ensuring trustworthiness in qualitative research projects. *Education for Information, 22*(2), 63-75. https://doi.org/10.3233/EFI-2004-22201

Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review, 22*, 142-158. https://doi.org/10.1016/j.edurev.2017.09.003

Silverman, D. (Ed.). (2020). *Qualitative research*. Sage.

Stager, G. S. (2016). Seymour Papert (1928–2016). *Nature, 537*, 308. https://doi.org/10.1038/537308a

Stendal, K., Thapa, D., & Lanamäki, A. (2016, January). Analyzing the concept of affordances in information systems. In *2016 49th Hawaii International Conference on System Sciences (HICSS),* (pp. 5270-5277). IEEE. https://doi.org/10.1109/HICSS.2016.651

Sturm, T., Gerlach, J. P., Pumplun, L., Mesbah, N., Peters, F., Tauchert, C., Nan, N., & Buxmann, P. (2021). Coordinating Human and Machine Learning for Effective Organizational Learning. *MIS Quarterly, 45*(3), 1581-1602. https://doi.org/10.25300/MISQ/2021/16543

Swedish Authority for Privacy Protection [Integritetsskyddsmyndigheten] (2021). Sensitive personal data [Känsliga personuppgifter]. Retrieved July 19, 2022 from https://www.imy.se/en/organisations/data-protection/this-applies-accordning-to-gdpr/the-purposes-and-scope-of-gdpr/personal-data/sensitive-personal-data/

Sveriges riksdag [The Swedish Parliament] (2003). Lag (2003:460) om etikprövning av forskning som avser människor [The law (2003:460) on ethical review for research with humans]. Retrieved July 19, 2022 from https://www.riksdagen.se/sv/dokument-lagar/dokument/svensk-forfattningssamling/lag-2003460-om-etikprovning-av-forskning-som_sfs-2003-460

Swedish Research Council [Vetenskapsrådet] (2017). Good research practice [God forskningssed]. Retrieved July 19, 2022 from https://www.vr.se/english/analysis/reports/our-reports/2017-08-31-good-research-practice.html

Szabo, C., Sheard, J., Luxton-Reilly, A., Becker, B. A., & Ott, L. (2019, November). Fifteen years of introductory programming in schools: A global overview of K-12 initiatives. In *Proceedings of the 19th Koli Calling International Conference on Computing Education Research* (pp. 1-9). https://doi.org/10.1145/3364510.3364513

Vaismoradi, M., Turunen, H., & Bondas, T. (2013). Content analysis and thematic analysis: Implications for conducting a qualitative descriptive study. *Nursing & Health Sciences, 15*(3), 398-405. https://doi.org/10.1111/nhs.12048

Vetenskapsrådet [Swedish Research Council] (2004). Internationella arenor för utbildningsvetenskaper [International Arenas for Educational Sciences]. Retrieved July 23, 2022 from https://www.vr.se/analys/rapporter/vara-rapporter/2004-09-10-internationella-arenor-for-utbildningsvetenskaper.html

Vinnervik, P. (2022). Implementing programming in school mathematics and technology: Teachers' intrinsic and extrinsic challenges. *International Journal of Technology and Design Education 32*, 213-242. https://doi.org/10.1007/s10798-020-09602-0

Webb, M., Davis, N., Bell, T., Katz, Y. J., Reynolds, N., Chambers, D. P., & Sysło, M. M. (2017). Computer science in K-12 school curricula of the 2lst century: Why, what and when?. *Education and Information Technologies, 22*(2), 445-468. https://doi.org/10.1007/s10639-016-9493-x

Yadav, A., Hong, H., & Stephenson, C. (2016). Computational thinking for all: Pedagogical approaches to embedding 21st century problem solving in K-12 classrooms. *TechTrends, 60*(6), 565-568. https://doi.org/10.1007/s11528-016-0087-7

Zhang, L., & Nouri, J. (2019). A systematic review of learning computational thinking through Scratch in K-9. *Computers & Education, 141*, 103607. https://doi.org/10.1016/j.compedu.2019.103607

Papers