Early Code Project

"Developing Teaching Materials for Preschool Teaching Undergraduates on Computational Thinking and Introduction to Coding

Project No: 2018-1-TR01-KA203-058832

Co-funded by the Erasmus+ Programme of the European Union

Training Manual

2018-2021

www.earlycoders.org earlycoderseu@gmail.com earlycoderseu earlycoderseu earlycoderseu

Project Coordinator
Kırşehir Ahi Evran University (Turkey)

IO-3 Responsible Partner
University of Latvia (Latvia)

Other Partners
Scuola di Robotica (Italy)
EarlyYears (Northern Ireland)
University of Bucharest (Romania)
Gazi University (Turkey)
A.P.E.C. (Turkey)

Contents

Introduction 5
Introduction to Computational Thinking...6
Computational thinking and Digital Literacy...7
Computational Thinking in Preschool Education...8
Introduction to Coding..9
Coding in Preschool Education..10
Introduction to Educational Robotics and SMART Pedagogy.......................................12
Educational Robotics...12
SMART Pedagogy...14
Educational Robotics and SMART Pedagogy in Preschool Education.....................15
Development Stages in Preschool...16
Screen-based and Screen-free Digital Technologies in Teacher Education.................20
Digital Technologies in Preschool Education..22
How to Improve Computational Thinking and Coding Skills?......................................24
Learning Styles...24
Assessment 27
Assessment and fairness...27
The Teaching-learning Process in Preschool..29
Assessment in Preschool Education..31
Preparing Instructional Materials and Games on Computational Thinking and Coding...33
Principles of Preparing Instructional Materials in Preschool...................................33
Assessment Tools for Preschool Materials..34
Learning Environment in Preschool...35
How to Bring Changes into Preschool Education?..37
Conclusion...39
Resources for Existing Preschool Computational Thinking Materials..........................40
References...41

Introduction

Teaching programing in a preschool education context provides a creative, active, and play-based learning environment for children to explore, enjoy, and construct their knowledge. Seymour Papert's (1980) constructionism framework stated that digital tools and environments are supported by an individual's active learning, which is done by providing new ways of constructions in the world. Computational thinking enables individuals to use new ways of representing and interacting with information and a new way of "objects to think with" (Berns et al., 2014; Papert, 1980).

Furthermore, digital environments enthrall children, by combining education and entertainment "edutainment" (Baird & Henninger, 2011, p. 5). The digital context can trigger motivation of young children to learn (Altun & Ulusoy, 2017; Chang, Lee, Chao, Wang & Chen, 2010; Hwang & Chang, 2011). In addition, computational skills foster children's problem solving, logical reasoning, visual memory, number sense, language skills, fine-motor skills, and hand-eye coordination (Chen, Quadir & Teng, 2011; Clements, 1999; Kazakoff, Sullivan, & Bers, 2013; Sugimoto, 2011; Toh, Causo, Tzuo, Chen, & Yeo, 2016; Zaranis, Kalogiannakis, & Papadakis, 2013). Therefore, preschool children should be introduced to precoding and computational skills activities for their future success so that they can fully participate in our digital world (NAEYC, 2017).

The importance of developing computational thinking and the coding skills of preschoolers, constitute the rationale for the EarlyCode project.

The Training Manual on Computational Thinking and Introduction to Coding for Preschool Education has been developed by the EarlyCode Project team in order to guide trainers and lecturers to implement the curriculum, (also an intellectual output of the project), in preschool education programs. This manual (IO-3) is compatible with the previous intellectual outputs of the project which are the curriculum (IO-1) and the handbook of teaching materials (IO-2). In the Training Manual you will find information about some key terms covered in the project such as computational thinking, coding, educational robotics, digital technologies in teacher education, teaching-learning process in preschool education, assessment and instructional material development. In addition, the manual includes information and guidance on how to implement the curriculum (Computational Thinking & Coding for Children) and how to improve computational thinking and coding skills.

Introduction to Computational Thinking

Computational thinking (CT) is the use of problem-solving methods, by means of formulating problems and solving them in the same manner as a computer would do. Computational thinking and programming are at the center of the debate on exploiting the full potential of ICT which emerged as a new concept to help prepare children for future challenges in an increasingly digital world. Indeed, these skills are now considered by many as being as fundamental as numeracy and literacy.

According to Arfe (2020) from the University of Padova, computational thinking as a skill consists of the following problem-solving abilities: analyzing the problem space, reducing the problem difficulty by decomposing it into smaller units, developing an algorithm or plan, or more specifically a set of instructions or steps to solve the problem and finally, to verify that it has reached its goal.

There are four computational thinking concepts: algorithms, sequences, loops and conditional or if statements that correspond to certain abilities in children which refer to understanding and using abstraction, sequencing, decomposition and debugging.

CT learning activities are believed to encourage positive technological development through the enhancement of six behaviors (Bers, 2019) also known as the 6 “C”: Communication, Collaboration, Community-building, Content Creation, Creativity and Choices of Conduct.

Working with the aforementioned six behaviors and abilities will encourage children to have a better sense of confidence and competence and be equipped to participate in a digitally - literate community, and as a result, be better integrated socially in their adult life.

Ultimately, the goal of CT learning is to stimulate cognitive development such as abstract thinking and reflective reasoning in the early years in the hope of laying foundations for understanding more complicated computational processes later on in life, that will make future adults more technologically informed and to even consider a career in the domain as digital and software creators.

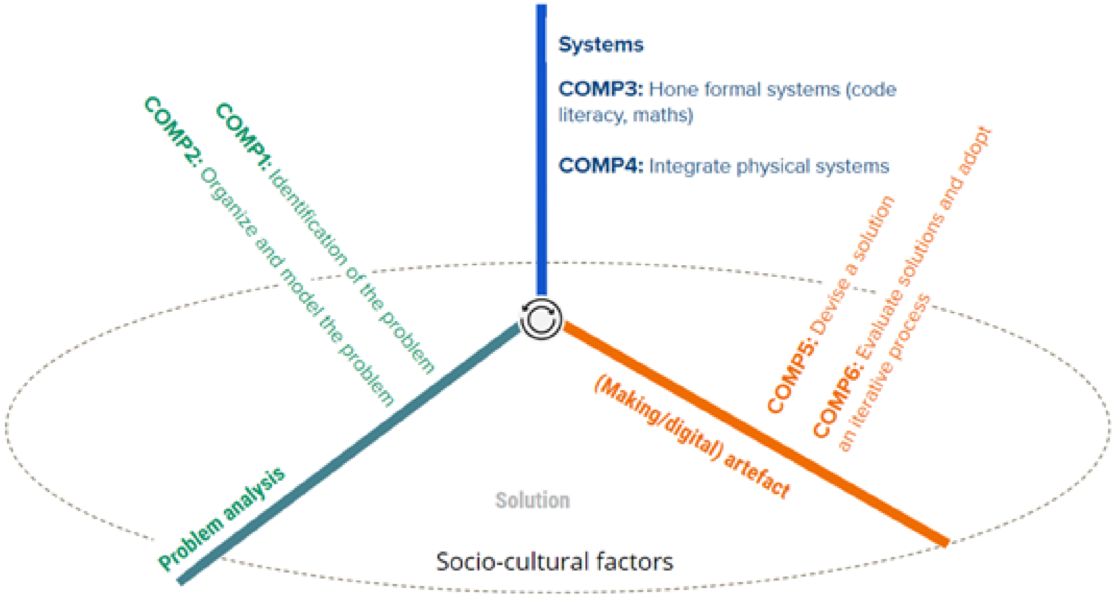

Figure 2. Computational thinking components (Romero, Lepage, & Lille, 2017)

Computational thinking engages components related to the analysis of the problem situation and the way in which the subjects organize and model the problem (problem analysis axis), honing formal systems with the use of a certain programming language and the integration of physical systems (systems axis) and the devices of an intermediate solution, its evaluation and improvement (creation axis). When learners are only engaged in coding, they develop knowledge related to the systems, but they do not engage in the full process of analysis, modeling and iterative creation of a solution (Romero, et al. 2017).

Computational thinking and digital literacy

Digital competence involves the confident and critical use of electronic media for work, leisure and communication. These competencies are related to logical and critical thinking, high-level information management skills, and well-developed communication skills (Ranieri, 2009). Digital competence is one of the eight key competences for lifelong learning developed by the European Commission (European Commission, 2005).

The European Digital Competence Framework (DigComp) offers a matrix to understand the essential competencies need for all citizens to adapt in the digital world. Understanding where one is located on the DigComp index can provide guidance to improve a citizen's digital competence and to achieve goals related to work, employability, learning, leisure and participation in society. (European Commission, 2019; European Union, 2016). It describes 21 learning outcomes in 5 areas: 1) Information and data literacy; 2) Communication and collaboration; 3) Digital content creation; 4) Safety; 5) Problem-solving. (read more here: https://ces.to/nNNBhJ) The use of this framework means that citizens with a basic level of digital competence as well as everyone else should develop competencies in all areas of the DigComp framework.

Computational Thinking in Preschool Education

Even though nowadays, our children are surrounded by multiple complex and high-performing devices, they only learn a little about these resources during the first years of education. “For decades, the early childhood curriculum was focused on literacy and numeracy, with a particular focus on science, especially the natural world.” (Bers et al., 2013, p. 357). In addition to the knowledge related to the natural world and environment, the need to discover the artificial world, created by humans is more and more prominent.

Computational thinking in children’s early years contributes to a better understanding of using computer-based technologies which are necessary for today’s world and the future. Enhancing computational thinking and teaching coding, encourages children to create and develop new products instead of just being passive users of technology. Early computational thinking (ECT) forms the basis of social, emotional, physical and cognitive development of children and contributes to the lifelong learning process. Developing computational thinking skills will enable children to be effective decision makers, problem solvers and creative innovators in the future.

However, a curriculum that teaches CT skills and abilities has to evolve as the developmental stage of children evolves (Bers, 2019; Arfe, 2020). Furthermore, one has to also assume, that while some children may struggle with aspects of CT and learning, others may be well ahead of their classmates and simply be ready for more, such as moving on from loops or sequencing sheets to using a computer and even make games by using Scratch.

Consequently, this factors in the presumption that early education teachers can relay this type of teaching material efficiently and integrate it seamlessly into their class in a manner appropriate to the children’s development stage. We presume the teachers through their professional development, understand or are familiar with these basic computational thinking abilities and how to take it to the next step, if the class is ready to do so.

Introduction to Coding

This whole book is based on the concept of coding. In this chapter we will discuss the meaning of this term, and then to the concept applied in the field of informatics, then to the adaptation to the world of education, especially in pre-schools.

Semantically speaking, the verb "to code" means the process of assigning a code to something for classification or identification, so to create a set of rules that relate or convert a class of information to another form (Oxford English Dictionary, n.d.; Cambridge Dictionary, n.d.). This meaning is shared in the field of ICT, but here the process of creating the ruleset means, more precisely, the process or activity of writing computer programs.

The question then is what a computer program is? According to Rochkind (2004) a computer program is a collection of instructions, or rules, and data, written by a programmer. The program can be interpreted and executed by a computer to complete a task. Hoare (1969) and Dijkstra (1973), further, shows that a program can be intended on an axiomatic base. It means that, starting from some axioms – so a group of evident principles, without any need of proof – a set of inference rules that describes the program can be defined and proved.

A program is written in "programming language". This is a code that give instructions to a computer. The code can be directly executed by the machine, but usually higher-level programming languages are used, where the code is converted in something more understandable by humans (Hemmendinger, 2021).

Coding refers to the process of writing instructions for machines in such a way that they are able to understand them and behave or respond accordingly. A similar meaning is expressed by Ricketts (2018) who wrote "Coding involves an algorithm design, which is a set of instructions used to complete a task or to solve a given problem".

This, in the field of education, means not only learning how to manage ICT skills – directly related to the coding process – but also opening a wider approach to the learning process. As wrote by Papert in 1980, talking about tech-assisted education, "the child programs the computer and, in doing so, both acquires a sense of mastery over a piece of the most modern and powerful technology and establishes an intimate contact with some of the deepest ideas from science, from mathematics, and from the art of intellectual model building." This same meaning is shared by other researchers. As an example, Dagdilelis et al (2004) describe the use of information technology in education as relevant in three different ways:

1. Firstly, the technology may be directly the focus of the learning approach, so students will learn how to manage IT, as in a computer science course

2. On the other hand, the IT could also be used by the teacher as an instrument - a tool - that supports the learning of different subjects

3. Furthermore, technology may indirectly influence the learning process

In the K-12 curriculum, coding is introduced not so much to teach students how to become programmers, but to provide them with the skills in working and operating programmable, intelligent machines and robots. Similar skills relate to the responsible and appropriate use of digital and robotic devices. Furthermore, they can develop collateral skills in doing robotics and programming.

Coding in Preschool Education

As seen before, the coding process stimulates a very broad learning area. Recent studies prove the positive effects of using coding in early childhood on children's attitudes, knowledge, and skills in various areas such as problem-solving and computational thinking (Bers et al., 2014; Sullivan and Bers, 2016; Çiftci & Bildiren, 2020).

According to Lee (2020) an appropriate manner must be adopted in how to expose children to technology. In the introduction we discussed the role of computers in programming - this educational approach is called "plugged coding". However, another approach could be used. It is called "unplugged coding". In unplugged coding there is no computer to be coded. Instead the focus of the activities is on the algorithmic solution of problems, and it can be expressed by a code that could be interpreted without the use of computers. For example, an unplugged coding activity could be expressing the sequence of washing hands: it is not directly programming, but it is working on algorithms, sequences and codification: basis of coding (Lee & Junoh, 2019).

In the same research Lee and Junoh show how the focus must be on sequencing and timing, so how we can express actions and activities in sequence. Then they suggest introducing movements and directional commands, for example giving children instructions about when and how they have to move. This leads children to explore both coding and motor skills. These kind of activities, then, can also be done using the "plugged" approach, inserting robots like Cubetto, Bee Bot, Lego WeDo, Thymio.

The domain of robotics is multidisciplinary, including various subjects such as engineering, electronics systems, finite automata, control technology, communication, vision, computing, and systems design. In early childhood education many of the previous concepts can be introduced and delivered through the curriculum (Komis & Misirli, 2016). Komis and Misirli also report how robotics could support the first approach to basic programming concepts, and how to set up a proper educational environment, where computational and algorithmic thinking are also developed.

Another option is to introduce screen-based programming, like the programming of animations and games. An example of this is the software Scratch Jr, developed for children from 5 to 7 (scratchjr.org). It is a free digital coding playground that introduces powerful ideas of computer science into early childhood education (Bers, 2018a). These ideas are aligned with educational computer science frameworks utilized in schools, such as the K-12 Computer Science Framework, the CSTA K-12 Computer Science Standards, and the ISTE Standards for Computer Science Educators. It is a software used in classrooms and homes worldwide. It enables children, who might or might not know how to read, to create interactive stories and games by snapping together graphical programming blocks (Bers, 2018b).

Introduction to Educational Robotics and SMART Pedagogy

Nowadays, ideas for incorporating robotics activities into the learning process are no longer a novelty, but there is still the question of how to use them to promote the development of certain competencies, and which pedagogical principles should be considered in order to improve students' motivation to look for new innovative solutions.

The aim of the activities planned in the EarlyCode Project is to ensure that students are supported to become technology literates and to develop competencies needed to become creators of new innovative ideas. The project team believes that if teachers in the future can provide children with specific knowledge, they will be able to understand robotics at an early age. Therefore, the project developed a curriculum for preservice preschool teachers, which combines the ideas of SMART pedagogy and educational robotics to support computational thinking development in preschools.

Educational Robotics

The first researcher who started talking about the possibility of using robots for educational purposes was Papert (1984), who developed the idea that students construct their knowledge in mathematics and understand the basic principles of physics if there are computers used in the learning process, where students are active participants in programming. Later on, he developed the programming language LOGO and Turtle robots to allow students to participate in the construction process and to discover the knowledge by themselves. He defined his idea as constructionism, which was widely developed and used in educational settings.

In the previous section, we introduced the idea of robots and noted how they could be used in education. In this section, we will try to formalize it, discussing the concept of Educational Robotics. Defining the concept of a robot, however, is not that simple because there are many different definitions, and this broad class of devices makes it difficult to give one unique definition (Guizzo, 2018).

Merriam-Webster (n.d.) in his definitions of robots focuses to the concept that they are machines, that may perform – somewhat autonomously – a series of tasks, event complex ones. Another definition is "A robot is a machine which is programmed to move and perform certain tasks automatically" (HarperCollins, n.d.).

Ben-Ari and Mondada (2018) show that automation in carrying out activities is a key element in robotics. They also write, "The difference between a robot and a simple automaton like a dishwasher is in the definition of what a "complex series of actions" is.". They also note that only few definitions of robots include the concept of sensors, but robots – instead of simpler automata – have sensors and use them to adapt their actions to the environment.

Putting all these elements together, we can move nearer to the definition given by Guizzo (2018): “A robot is an autonomous machine capable of sensing it’s environment, carrying out computations to make decisions, and performing actions in the real world.” In this manual we will use this definition to describe a robot.

Educational Robotics is then a branch of education, which uses robots as didactic tools to support the teaching/learning approach. For this reason, some robots have been explicitly designed to be used as a teaching tool (EARLY, n.d.).

Sant’Anna School of Advanced Studies writes “Educational Robotics is … a method to increase the quality of scientific and technical education … it increases involvement, develops problem-solving skills, promotes an interdisciplinary approach and supports the development of teamwork”.

Angel-Fernandez and Vincze (2018) note how educational robotics may involve several uses of the robots. They suggest the following definition: “Educational Robotics is a field of study that aims to improve the learning experience of people through the creation, implementation, improvement, and validation of pedagogical activities, tools (e.g. guidelines and templates) and technologies, where robots play an active role and pedagogical methods inform each decision.”

The e-Media project Consortium (2019) writes that children learn quickly and easily if they can use concrete and physical objects, like robots – 3D devices – are, following the constructivism and constructionism theories (Piaget, 1974; Papert, 1980). Furthermore, robots act as facilitators for learning and can evolve into a rich experience for children, who will take different skills and competencies from different subjects.

According to Alimisis (2013), educational robotics can improve the learning approach, but the solution is not only to introduce robots in didactics. Learning robotics is not the goal to be achieved by educational robotics. Instead, a didactic curriculum update, to include robots as tools, has to be made. This manual follows this vision, where robots, and in general other devices, are not the learning goal, butthey act as supporting instruments for the teachers.

SMART Pedagogy

SMART pedagogy is characterized by the idea of a learning process that takes into account all philosophical and psychological aspects of education and involving digital technologies in their implementation (Daniela, 2019). The idea is based on organizing a student-centered, effective, engaging, experience-based and technology-enriched learning environment that promotes student self-directed activity (Zhu, Yu, & Riezebos, 2016).

In context of technology enhanced learning, the role of pedagogy becomes more important for finding the ways to incorporate technology into education. The concept of Smart Pedagogy is triangular, where the important cornerstones are:

1. Human developmental regularities, which include the conditions for the development of cognitive processes, the conditions for sensory development, as well as the conditions for socio-emotional development.
2. The taxonomy of the educational process, which includes the goals to be achieved and the regularities of the learning process needed to achieve these goals.
3. Technological progress, which entails the need for changes in teachers' pedagogical competence, where one of the most important components of this competence is Predictive Analytical Competence (Daniela, 2019).

Extensive literature analysis, researching scientific articles that have analyzed teacher digital competences, digital literacy, media literacy and computer literacy acquisition processes, summarized the eight most frequently used approaches / methods of digital competence development for future teachers (Røkenes, Krumsvik, 2014):

- **Collaboration** that ensures the exchange of experience and knowledge by learning new concepts or deepening knowledge about them.
- **Metacognition** to analyze the learning process of oneself, as well as to reflect on the meaning and effectiveness of the use of digital technologies.
- **Blending,** combining face-to-face and e-learning activities, or encouraging students to construct / summarize / demonstrate knowledge using digital technologies that can be implemented in video format, collaborating on a specific digital platform, etc.
- **Modeling** – demonstrations of digital technologies by lecturers, practicing teachers or other experts when modeling or implementing learning situations in a real school environment.
- **Authentic learning,** giving students the opportunity to apply and test their theoretical knowledge of the use of digital technology in the classroom, by modeling lesson plans, developing curricula and then implementing it in the classroom or discussing it with an

expert.

- **Student-active learning,** encouraging students to develop their own digital solutions for specific learning situations.
- **Assessment** as an important part of the part of learning, research and modeling using digital technologies identifying different assessment situations and applying the most appropriate ways to obtain feedback using digital solutions.
- **Bridging theory/practice gap,** is based on deepening knowledge on a specific topic that is closely linked to active and authentic learning and using digital technologies to offer students situations in which they may not end up in just visiting schools or in teacher practicum. This means using technologies such as virtual stimulators, video analysis, and more to provide students with challenging and unexpected situations.

Educational Robotics and SMART Pedagogy in Preschool Education

Young children must approach technologies gradually using educational robotics tools and kits such as Lego, WeDo or Beebot or organizing technology free activities, learning about sequences, enhancing logical thinking skills, modeling or creating constructions with different objects. Over exposure to digital devices, or wrong pedagogical approaches, may lead to harmful results. For this reason, it is important to start digital activities, in Early Childhood Education, with analogue ones, so unplugged activities. These kind of activities can both help children become familiar with the basis of coding, programming, and computational thinking and also with physical, motor and socio-relational and emotional skills.

Once young children can properly manage unplugged activities, and once they are old enough to deal with digital devices, actual robots can be introduced. According to several studies, such as health Domingues-Montanari, 2017 and Stewart et al, 2019, screen-time exposure to young children may have negative effects on their health. For this reason, it is better to firstly introduce digital off-screen devices, such as electronic robots and didactic toys that can be used and programmed without the use of screen-based devices, such as computers and tablets. These kinds of devices may include didactic tools, such as Bee Bot, Cubetto, mTiny robots and other similar systems. These activities may act as a bridge from unplugged to screen-based ones. These kinds of without-screen systems may let children concentrate more on the didactic aspects.

Older children, who are able to properly relate with others, and can sufficiently concentrate on the previous activities, can be introduced to screen-based lessons. A wide educational software library exists, and some tools such as using ScratchJR, code.org, Blue Bot and others, can be chosen. These types of lessons lead children directly to the actual coding, and let them concentrate more on computational thinking skills.

Also, independently from the system that you adopt, a gamification approach can be useful in the teaching model. A review of more than 40 studies found that play is significantly related to creative problem solving, co-operative behaviour, logical thinking, IQ scores, and peer group popularity (Lamrani & Abdelwahed, 2020). According to Mitchel Resnick's view (2017), a play-based approach can support learning. He writes about children who play "They play games, they play sports. They play musical instruments, they play songs. They play the odds; they play the stock market. They play with toys; they play with ideas". Then he suggests adopting a "playground" setup, where children may "play", intended as actively explore, experiment, collaborate. Doing this, children will be actively engaged, and the learning approach will be supported.

Development Stages in Preschool

During the preschool years children rapidly grow and develop new skills. Child developmental stages are important indicators for teachers and parents to follow and better understand children and their interests. Nevertheless, it is important to remember that each child develops uniquely. Therefore, we have divided all preschool age in 3 wide sections to generally describe children's development in preschool (see Table 1).

	1 to 2 years	3 to 4 years	5 to 7 years
Social emotional development	The child begins to explore the world around him, gradually moving away from adults and becoming more independent (Levine, Munsch, 2017). Remembers and recognizes teachers. Emotionally responds to the feelings of other children or adults. Learning the rules. Self-esteem and awareness of one's identity are growing. Around the age of two, a child may show a very stubborn attitude - frequent and insistent use of "no". Children may have tantrums about simple things that are very important to them. A two-year-old child is only able to follow an adult's instructions 45% of the day (Centrs Dardedze, 2020).	The child strongly feels the concept of "one's people" and "strangers". At this age, children begin to identify the differences that exist between group members and other people. The child's morality is based on society's assumptions about how something – the child acts based on what is accepted in the family / educational institution. (Bourn, Hunt, Ahmed, 2017) It is important for a child at this age to feel that he is able to act independently, thus gaining confidence in his own strength for the future (Erikson, 1950). Children begin to follow the rules in games, according to them they perform various activities and roles. One of the most important tasks is to learn to play both together and next to others.	The notion of justice and equality is beginning to take shape, that comes together with expecting other peers to follow the same rules as well (Bourn, Hunt, Ahmed, 2017). The child should be allowed to take the initiative, thus encouraging self-determination (Erikson, 1950). It is difficult for a child to understand such emotions as pride, gratitude, jealousy. The child is able to talk to himself to calm down. As the age of 7 approaches, child begins to realize that he or she may have conflicting emotions about an event or person. Begins to realize that others also have their own thoughts, feelings and emotions. Can share and often shares with others (Centrs Dardedze, 2020).
Physical development (small finger muscles)	As the child learns to walk and is able to comprehend more and more space, there is an active exploration of the physical world, which is realized using all the senses (sight, touch, taste, smell, ...) (Bartolotta, Shulman, 2013). The child learns self-help skills by putting his or her belongings in place, trying to eat and take off some items of clothing that require coordinated and purposeful movements.	Significant time for the development of fine motor skills, which is also closely related to the development of eye-hand coordination. Therefore, it is important to activate both small motor skills by learning to tear, cut, button and to develop large motor skills by training to throw, roll and catch objects.	Able to imitate complex movements. Gross motor skills: - Jump with a jump rope, change legs, swing, learn to swim, dance to the rhythm. Fine motor skills: - Skillfully cuts with scissors, holds a brush and pen with a thumb and forefinger, paints. - The choice of the main hand is strengthened. - Write your name and a few letters in block letters. (Centrs Dardedze, 2020)

	1 to 2 years	3 to 4 years	5 to 7 years
Physical development (small finger muscles)	Gross motor skills: - Jumps, walks on toes, stands on one leg, keeps balance, walks in a straight line. - Throws and catches the ball, climb the stairs, alternating legs. - Around the second year of life, children often begin potty training. Fine motor skills: - Is able to screw the lid on the jar. - Builds towers from more than 10 blocks, put puzzles together (according to age). - Cuts with scissors, paint with a brush, draw a person with three parts, draws a circle, can copy some shapes and letters. (Centrs Dardedze, 2020)		
Language skills	By the age of 1, the child will most likely have said their first word. Thus, rapid language development begins, where conversation with the child is crucial. Until the age of 2, the child learns not only to name the objects he/she sees, but also to talk about events and objects that are not present - can refer to a memory or a wish (Bartolotta, Shulman, 2013).	At this age, the child's vocabulary expands rapidly. The child's frequent use of newly acquired words also improves the clarity and grammatical correctness of the language (Elnebija). The child becomes interested in listening to stories that stimulate his / her imagination. The child is able to talk briefly about himself - name, age, likes and dislikes.	The child's vocabulary continues to grow (10 000-14 000 words). Able to create a narrative of past events, as well as successfully engage in dialogue. Sentences of more complex structure are beginning to appear in the language, and most often there is an active interest in reading and writing (Elnebija).

	1 to 2 years	3 to 4 years	5 to 7 years
Language skills	Begins to use words that compare objects (bigger, smaller, equal), knows their name, learns to sing short songs and count verses. Vocabulary is about 1000 words, start naming colors, forming small sentences. (Centrs Dardedze, 2020)	Child starts asking questions about events and things that are not yet fully understandable or known. At this age children like to talk but do not always answer questions. The child knows his name, surname. It is common for children to get involved in the game by using words and commenting on the progress of the game (Centrs Dardedze, 2020)	At this stage, it is important to identify and to address incorrect speech.

Screen-based and Screen-free Digital Technologies in Teacher Education

Digital literacy today is considered to be one of the basic skills needed by every member of society (Council of the European Union, 2018). Consequently, the education sector plays an important role in ensuring digital literacy's full and successful acquisition in society. Teacher's digital competence includes the idea of meaningful use of digital technologies to organize students' learning activities (Brevik, Gudmundsdottir, Lund, u.c., 2019). It reflects the educator's ability to engage, adapt and develop appropriate digital technologies for students' learning needs (Brevik, Gudmundsdottir, Lund, u.c., 2019; Instefjord, Munthe, 2017). The European Commission proposes a model of pedagogical digital competence, distinguishing teacher professional and pedagogical competences, as well as demonstrating their impact and developing students' digital literacy (Redecker, 2017). In this model teacher's professional activity is divided in 6 sections (see Figure 2), where sections 2 to 5 include the implementation of teacher's pedagogical digital competence, while section 1 is more related to wider interaction and professional development of teachers and other professionals, and section 6 deals with teacher competencies that contribute to the development of students' digital literacy (Redecker, 2017).

Each section includes specific pedagogical and digital competences:

1. Professional engagement – the use of digital technologies for cooperation and communication, as well as professional development.
2. Digital resources – planning of the teaching-learning process using various search tools, databases, digital resources and sharing them in the digital environment, as well as selecting digital tools according to the content of their subject, the needs of students and the teacher's teaching style.
3. Teaching and learning – the use of digital technologies in the organization of a diverse learning process, supporting students' learning activities individually and as a team.
4. Assessment – obtaining feedback on their professional activities, as well as providing their feedback to students through effective and diverse assessment strategies and evaluating students' performance both during and outside the use of digital technologies.
5. Empowering learners – use, development and creation of digital tools to promote a student-centered, inclusive, personalized and active learning.
6. Facilitating learners' digital competence – offer students the opportunity to use digital technologies creatively and responsibly to search for information, solve problems, communicate and create new digital solutions.

Educators' professional competences
Educators' pedagogic competences
Learners' competences
1. Professional engagement
- Organizational communication
- Professional collaboration
- Reflective practice
- Digital CPD
2. Digital resources
- Selecting
- Creating and modifying
- Managing, protecting, sharing
3. Teaching and learning
- Teaching
- Guidance
- Collaborative learning
- Self-regulated learning
4. Assessment
- Assessment strategies
- Analyzing evidence
- Feedback and planning
5. Empowering learners
- Accessibility and inclusion
- Differentiation and personalization
- Actively engaging learners
6. Facilitating learners' digital competence
- Information and media literacy
- Communication
- Content creation
- Responsible use
- Problem solving
Digital competences
Subject specific competences
Transversal competences
Subject specific competences

The use of digital technologies in teacher education is based on the lecturer's digital skills (Krumsvik, 2011). Therefore, in order for a lecturer to be able to successfully use digital technologies in organizing students' work, he or she first needs skills to use digital technologies to fulfill his / her learning needs, organize his / her daily work, search for information, etc. As the lecturer gains a positive attitude towards the use of digital technologies, as well as the skills to use them, the lecturer can begin to identify the opportunities offered by digital technologies for organizing students' learning. The lecturer reflects on how the changes introduced promote or obstruct their own professional activity, students' learning, cooperation with colleagues and parents, and explore other opportunities offered by digital technologies to find new pedagogical solutions. (Krumsvik, 2011).

Digital Technologies in Preschool Education

Digital technologies are seen not only as part of the daily lives of adults, but also as part of children's lives. The European Commission has acknowledged in an extensive study that a user of modern technology is 0 years old (Chaudron, Di Gioia, Gemo, 2018), highlighting unprecedented educational challenges, trying to develop balance between learning to use digital technology and a child's active and experiential learning. In a family environment, children are most often active users of digital technologies, being aware of what digital technologies are available at home and which ones the child likes or dislikes, the child feels the content offered by digital technologies and media as part of their daily routine. (Chaudron, Di Gioia, Gemo, 2018).

Children's use of digital technology is closely linked to the wide range of digital technologies that exist in today's environment, offering touch screens, interactive textbooks and toys that can be intuitively perceived and learned by children without much adult help. This creates a misconception that the child learns the use of digital technologies himself and that the presence of an adult is not necessary in this process. By offering a child age-appropriate digital content, the child does not gain a learning experience, but, for example, enters and exits different applications, swipes between pages, watching the movement rather than learning (Chiong, Shuler, 2010).

There are several ways an adult can position and offer digital technology to children (Chaudron, Di Gioia, Gemo, 2018). Each of these strategies can have a suitable place and time, depending on the learning tasks, needs and general rules of the group's work organization.

- Co-use – an adult using digital technology together with a child. As the child acquires new knowledge, the presence of an adult is important, so that the child can ask questions, understand for what purposes and how a particular digital tool can be used.
- Active mediation – helping a child with technical or substantive difficulties encounte-

red in using the device or application. The adult serves as a helper in a situation where the child already has experience in using a certain digital device or application, but skills have not been stabilized.

• Restrictive mediation – setting limits on the child's use of digital technology in the time, space or content that the child may use. These are the rules that exist in a group regarding the daily use of digital technology – when children are allowed to use digital technology, for how long, where in the room it is allowed, and so on.

• Monitoring – the child may use digital technology in the presence of an adult, or the adult reviews the activities performed by the child after using the device. This approach works as a successful condition in preschool, not leaving the child alone in the use of digital technologies and also not constantly analyzing what and how the child uses.

• Active distraction – the adult tries to divert the child's attention from the use of digital technologies to other real-life activities. The use of digital technologies should be followed by the use of newly acquired knowledge or skills in a real learning environment, inviting children to engage and participate in practical activities – movement games, conversations and other activities, ensuring the child's comprehensive development.

Remember that young children's learning needs teacher-directed activities, instead of free or passive use. The teacher has to act as an active role model, and to stimulate the use of technological tools to stimulate children's technological learning (Gimbert & Cristol, 2004). Please, also remember that a digital device does not only mean computers or tablets. As research shows, early childhood exposure to screen-based devices can lead to several problems (Stewart et al, 2019). A related topic is that it is important that students learn how to focus on specific tasks for a specific amount of time that is appropriate for their age. This means that for preschoolers and early childhood students, ICT tools, which naturally draw children's attention, should be introduced or available only when the actual task is to use them (Hačatrjana, 2019). Thus, it is important to correctly blend both screen-free and screen-based activities, and also to keep in mind to actively use screen-based system, instead of making children passive users of it.

How to Improve Computational Thinking and Coding Skills

For a very long time, coding was thought to be something that only people who were very skilled and even gifted at advanced mathematics could do and it was usually associated as a career path for adult men. However, recent attempts at simplifying coding and making it more accessible to people who are not mathematically inclined and especially children have opened up a wide array of possibilities for anyone to start learning coding in a simplified manner and then to move on to something more complex, if needed. There are numerous platforms that aim to teach preschool children how to code at a basic level such as Kodable, Blockly, Tynker, Code.org or Scratch.

In the following chapter we will be looking at Experiential Learning Theory and exploring 9 learning styles, along with deep learning concepts. Furthermore, we will be discussing learning principles for teaching-learning processes in preschool, followed by a few tips and tricks that will provide a guideline for preschool teachers who wish to integrate CT learning into their curriculum. Lastly, we will provide a framework on how preschool education assessment should be organized and what to keep in mind when assessing students work, with emphasis on assessment as an independent learning skill.

Learning Styles

Experiential Learning Theory (henceforth: ELT) reflects the way a person transforms their experience into knowledge (Kolb, 2015). The ELT model distinguishes between two ways of gaining experience - concrete experience and abstract conceptualization - and two ways in which experience can be transformed – reflective observation and active experimentation (see Figure 3). Thus, 9 learning styles are distinguished:

- Initiating – manifests itself in actively experimenting with concrete experiences. The student uses the knowledge from his / her experience, using it in new situations. Accordingly, the student is ready to integrate digital technologies, which he / she has used before (for his / her personal needs or study work) in a new setting – planning lessons for children, in order to promote the development of computational thinking.
- Experiencing – learning takes place through concrete and in-depth experience. The student sees the importance of experiencing and feeling on his / her own skin, situations in which he / she participates and reflects on his / her actions and learning. Thus, opportunities can be provided to allow students to get acquainted with the use of new digital technologies in the learning process and to experience the lesson from the their perspective.

• Imagining – imagining different actions or outcomes using your own experience. Allows you to manipulate various existing situations and imagine how they would affect the current situation. This is a successful way to introduce students to various problem questions before researching a real problem, for example, actualizing questions like "How in your experience did teachers deal with discipline problems?", "What strategies are used in your school to prevent bullying?", "What has been the best motivators for you to collaborate with your colleagues?".

• Reflecting – due to in-depth reflection on combining experiences and ideas. Demonstrating that both the student's existing experience and knowledge are combined to show conclusions. It is a process in which students summarize their knowledge of a given issue by analyzing it from different perspectives. During the course of study, students can form written or oral reflections, analyzing how dealing with the content of learning influenced or changed his / her perceptions of the use of digital technologies in preschool.

• Analyzing – reflection, which results in the integration of ideas into concentrated systems. The student seeks connections while processing abstract concepts and enriching his / her knowledge with new concepts. Once the student has developed a broader knowledge of the use of various robots, digital teaching aids and teaching materials, he / she can model their use in the acquisition of specific learning content in preschool.

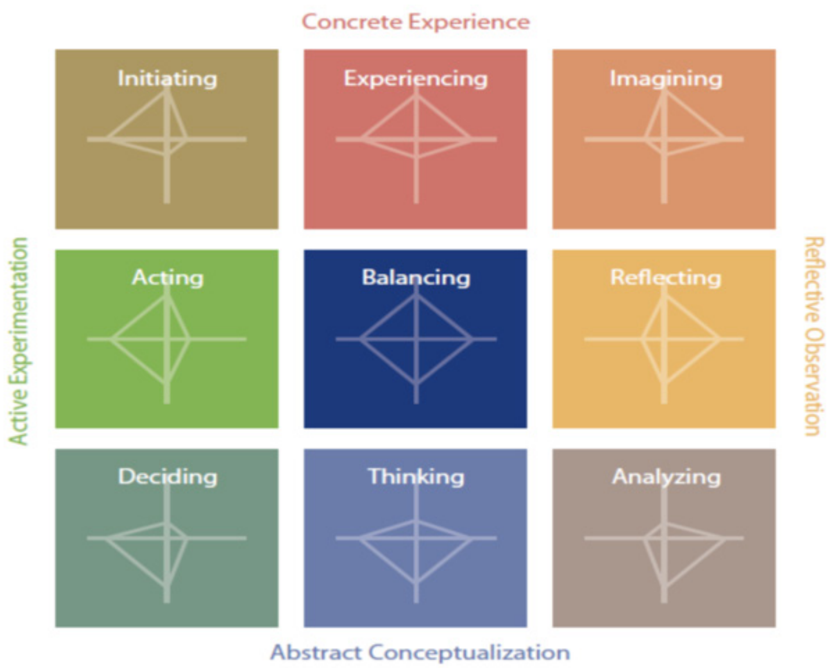

Figure 3. **9 learning styles** (Kolb, Kolb, 2013)

• Thinking – prudent involvement in abstract and logical reasoning. It is the use of one's own knowledge to discuss concepts that are familiar to oneself, thus "trying them out" in new situations and getting to know the opinions of colleagues. By organizing conversations, discussions and general exchanges of opinions, students have the opportunity both to formulate their understanding of the role of digital technologies in preschool and to come up with new ideas on how digital technologies can help support children's learning, engage children in learning content, strengthen children's motivation, etc.

•Deciding – using theories and models to decide on the most successful course of action. Students solve various situations by analyzing the theoretical literature on the process of improving programmatic thinking in preschool, the effective use of digital technologies for students' learning, etc., drawing conclusions and making abstract decisions about the solutions to these situations.

• Acting – strong motivation to achieve a specific goal by collaborating on tasks. Active experimentation with one's own experience and knowledge is essential for acting, so students should be given the opportunity to try out the acquired knowledge in real practice. This can be done, both by students going to practice and working with children to promote the development of their programmatic thinking, or by modeling situations between students, where one student is the teacher and other students are the pre-schoolers.

• Balancing – adapting and weighing up the strengths and weaknesses of use and reflection, experimentation and thinking in each situation. To implement this learning style, the student must have an understanding of how and at what moments to use each of the learning styles more successfully. Such a skill can be acquired by a student only through active reflection on his / her learning activities. When implementing complex problem situations in learning activities (for example, adding some specific limitations for the task (resources, time or opportunities), incorporating elements that increases difficulty (modeling of activities for children with special needs, outdoor activities), etc.), students can use balancing learning styles to reach the most effective solution.

A learning style is not fixed for a person's whole life but is a habit that a person applies in a learning situation (Kolb, Kolb, 2013). A learning style is formed from a person's experience and choice to change it (both conscious and unconscious). Accordingly, the learning style is strengthened in a person's personality by repeating similar learning situations. The way we learn in a new situation determines the choices and decisions we can see – the decisions we make in a situation determine the next situation we experience. Therefore, flexibility is part of successful learning.

Deep learning is based on 4 parts of the experiential learning cycle – experiencing, reflecting, thinking, acting. In order for a person to be able to experience deep learning, 3 action steps are distinguished: (1) acquisition and performance-based learning, which uses two stages of the learning cycle, (2) specialization and learning-oriented learning, which already includes 3 stages of learning, (3) integration and developmental learning, including 4 stages of learning (see Table 1). Learning flexibility is shifting from specialization to integration during one's life – a shift from the use of a particular learning style to the use of all learning styles, according to contextual needs (Kolb, 2015).

Table 2. Levels of deep learning (Kolb, Kolb, 2013)

Level of deep learning Stages of the learning cycle	Stages of the learning cycle
1. Acquisition and performance-oriented learning	Experiencing, reflecting – operates with provided information.
2. Specialization and learning-oriented learning	Experiencing, reflecting, thinking – increases the intensity of the learning experience by exploring new models using the information provided.
3. Integration and development-oriented learning	Experiencing, reflecting, thinking, acting – uses a wide variety of learning experiences, such as group projects to explore broader or more in-depth topics.

Assessment

Assessment is the process of gathering and interpreting evidence to evaluate the quality of student achievement (Atjonen, 2014). Self-directed assessment of one's own learning is a complex process that requires strong independent work and reflection skills. Today, the importance of self-assessment is especially emphasized in the context of self-directed learning. Self-directed learning is defined as the ability to decide, with or without the help of other people, identifying and articulating learning needs, as well as selecting and implementing appropriate learning strategies, and assessing learning outcomes. (Oladoke, 2006). When a student receives constructive feedback from the lecturer and the lecturer encourages the student to reflect on what has been done, the student's self-directed assessment skills are developed. (Cathcart, Greer, Neale, 2014). There are a number of factors that determine the provision of effective feedback in terms of self-directed assessment of learning outcomes.

Assessment and fairness

When the lecturer is ensuring the improvement of the student's self-directed assessment skills by choosing the appropriate type, methods and regularity of assessment, the following question becomes pertinent: how fair / appropriate is the assessment? The fairness of the assessment touches on a number of issues (Atjonen, 2014):

- result or process – focus on the correctness of the answer or the course of thoughts in the solution,
- equality – what opportunities to provide to students, whether or not they support their individual learning needs,
- criteria – what criteria are set for evaluation,
- personality – how and whether to evaluate the student's attitude and behavior.

Giving students more freedom of self-determination requires defining much clearer boundaries and conditions, showing a field of action in which students can feel free. A. Rasoli and his colleagues conclude (Rasooli, Zandi, DeLuca, 2019) that fairness in education is a topical issue that is widely mentioned but is lacking a clear definition and meaning. This conclusion is drawn from the analysis of 50 scientific articles, of which only 8 mentioned the definition of fairness in the context of classroom management. As a result of the research, fairness in the group is viewed in 3 aspects: fair opportunities, fair process and fair interaction (Table 3).

Table 3. Understanding of the fairness in the classroom (Rasooli, Zandi, DeLuca, 2019)

Type of fairness	**Content**	**How is it related to evaluation?**
Classroom distributive justice	Equity	Equal assessment, consequences and rules for all
	Equality	Providing opportunities for student support
	Need	Meeting the needs of each student
Clasroom procedural justice	Consistency	Compliance with conditions
	Accuracy	Accurate and timely documentation
	Bias suppression	Judging by achievements and not by prejudices or sympathies
	Correctability	The lecturer acknowledges mistakes and corrects them
	Ethicality	Respect for moral principles – prohibition of plagiarism
	Voice	The student expresses his / her opinion, ideas, interests, needs, wishes in the learning process
	Transparency	Establishing and following clear criteria
	Reasonableness	The criteria and conditions are meaningful and appropriate to the situation

Classroom interactional justice	Respect	Interest in students' work, achievements and needs, supporting both students' successes and mistakes
	Adequate, Truthful, Justified information	Lecturer-driven explanation of rules and criteria, as well as involvement of students in the development of criteria
	Timeliness	Providing information to students in a timely manner, allowing them to prepare

Summarizing the ideas presented in Table 2, fairness in the assessment process is understood as setting the same rules and requirements for all, taking into account the needs of students to achieve the results, as well as the adequacy and moderation of assessment criteria based on honest and timely exchange of information about time, place, purpose and criteria. It is also important to have open discussions and receive respectful feedback, during which a possible error would be acknowledged and corrected.

Teaching-learning process in preschool

As mentioned previously, efficient CT can be achieved progressively, as the child evolves, and age should not be and is not in any way restrictive when it comes to learning basic coding. The advantages of using CT in preschool education are related to an increased awareness of the usefulness of learned notions, developing logical and mathematical thinking, familiarization with a cognitive and independent work style, strengthening scientific investigation skills and an increased level of awareness of the usefulness of learned notions.

In the case of preschool children three to six years old at most, Bers (2008) advises a constructivist teaching approach to CT learning that offers children the possibility of exploring their own interests while also observing specific CT content provided by the use of computers and robotics, and also understanding the software and hardware correlations that make them function.

There are resources online for lesson plans that are more appropriate for five to six years old's that are mildly literate and can to some extent either individually or with the help of a teacher perform the tasks. However, in the case of younger children of three or four years old it has been proven by recent studies (Bers, 2014; 2018) that they were able to make use of code blocks marked with symbols in order to program a robot to perform a local dance. The robot read the blocks' instructions via scanning them.

Here are some tips and tricks on how to begin the process of familiarizing preschool children with computational thinking and coding basics.

1. Firstly, it is important to understand that breaking down CT concepts into their most basic form makes them comprehensible to young children. In this sense teachers' materials found online fall into the following categories: loops, decomposition, branching, algorithms, sequencing and to some degree, debugging.

2. Algorithm materials refer to precise instructions given to perform a certain task, while also verifying that it produced the desired result. Such material can include a simple maze at the end of which there is a treasure, and a pirate must reach it by moving along the maze. Children would have to mark via arrows which way the pirate must go in order to get to the treasure. It can also be more simply explored through examples from real life, such as putting clothes on in the right order or preparing the schoolbag in the right order.

3. Loops materials refer to how many times an action must be repeated in order to achieve the desired effect. For example, a sheet showing a bee that has to fly over 5 identical flowers in order to get the right amount of pollen back to the hive, can be used.

4. Sequencing materials refer to arranging a number of images in the right order by use of either narrative knowledge or logical evolution. To be more exact, children can be provided with Little Red Riding picture cards and be encouraged to arrange them according to their prior knowledge of the story, or to have them look at pictures of the seed-growth-flower bloom process at different stages and ask them to arrange the pictures in accordance with the natural evolution of the flower.

5. Decomposition materials allow for understanding how elements of the whole can be divided into smaller pieces that are easy to work with and invite children to attentively examine an image comprised of several parts and to focus on the attributes of just one. This can be achieved through any type of object arrangement such as a castle made of blocks of different colors and shapes where the individual building blocks have to be singled out by children.

6. Branching materials are designed as a means of understanding conditional or if statements. As such, the use of logical thinking is pivotal in this situation. Children can be asked to say what happens if it is raining outside and the logical answer would be to use an umbrella, or stay at home and so on, or be expected to choose the option of using an umbrella out of a set of other clearly unrelated options to the situation.

7. Finally, debugging materials simply invite children to look at a wrongly arranged sequence or algorithm and to point out the errors that keep the instructions from achieving the desired result and how to solve them.

In the end, we stress three main aspects that must be mentioned in respect to creating the premises of teaching CT in preschool education:

1. the need for a broader approach to the artificial world, created by man (science-technology-engineering);

2. the need to train teachers in these fields in terms of scientific, technological and engineering knowledge.

3. the need for pedagogical approaches and practices adapted to the level of children's development.

Assessment in Preschool Education

CT involves students solving problems, breaking them down into solvable parts, and developing algorithms to solve them. Therefore, it focuses not on the production of outcomes or evidence, but on students performing a thought process. Because it is difficult to measure actual thinking processes, these features of the skill can be challenging for evaluation. For teachers to know accurately what their students think and how they think, they need to show students' thought processes in some way (Mueller, Beckett, Hennessey, & Shodiev, 2017). Any approach to assessment should seek to explain where a student has been, where they are now, and where they might be going (Brennan & Resnick, 2012).

Unlike other skills such as language, literacy or mathematical thinking, there are no effective and reliable assessments to measure the CT skills of young learners. However, assessing CT skills can provide learning evidence and useful feedback for students, educators, and researchers evaluating the effectiveness of educational programs, curricula, or interventions (Relkin, de Ruiter, & Bers, 2020).

In the last two decades, a variety of tools have been developed to measure CT skills, but only a small group of studies have focused on CT in children as young as four to nine years of age in primary schools. Most previous work in early age groups used project-based coding assessments or interview protocols.

In the interview-based approach, the researchers analyzed the children's responses during individual interviews while observing the performance of programming tasks. Mioduser and Levy (2010) presented the results of LEGO robot building tasks for preschoolers. The children's CT levels were assessed qualitatively by analyzing the terms used to describe the robot's movements while navigating the built environment. For example, children who assigned robot movements to magic received lower CT skill scores, and children who made mechanical explanations were considered more advanced.

Wang et al. (2014) used a similar approach with children aged 5-9 who were asked open-ended questions about a concrete programming task they created called "Tmaze". "Tmaze" uses TopCode to convert physical programs to digital code. To determine whether the children grasped these concepts, the researchers identified CT elements in the children's responses (e.g., abstraction, decomposition).

Bers et al. (2014) analyzed programs created by preschool children (4.9-6.5 years old) using a tangible graphical programming language called CHERP. For example, the children were instructed to program their robot to dance Hokey Pokey. The researchers then assessed the four CT concepts by rating the children's projects on a Likert scale.

Moore et al. (2020) used a task-based interview technique to assess CT. Three participants were filmed as they were asked questions and completed tasks using the Code and Go Robot Mouse Coding Activity Set developed by Learning Resources. Researchers have qualitatively explored how children use representations and translations to invent problem-solving strategies.

Although project-based interviews and assessments open an opportunity for children to think, the form of these assessments and the time they require make them unsuitable for management outside of specific research scene. In particular, the interview-based approach is both time-consuming and subjective, and may be further constrained by children's ability to verbalize their thought processes (Relkin vd., 2020).

Recent efforts have been made to create CT assessments for young children. Marinus et al. (2018) created the 3-6 Coding Development Test (CODE) (for ages 3-6), using the Cubetto robot. CODE asks children to program the robot to go to a specified location on a mat by inserting wooden blocks into a "remote control". The task is to create a program from scratch or debug an existing program. Children are given up to three attempts to complete each of the 13 items, with more points awarded for fewer attempts. Although the authors state that CODE is intended to measure CT, their assessment requires knowledge of coding, which increases the likelihood that their assessors will confuse coding with CT skills.

In addition to these, unplugged activities have been used in assessment for educational purposes and more recently applied to the assessment of CT skills, primarily in children in elementary school and above. Code.org (www.code.org) provides a widely used online resource for teaching computer programming to elementary school children from Kindergarten to Grade 5 (ages four to thirteen). Code.org uses unplugged activities as assessments in its end-of-lesson quizzes. However, code.org does not provide a scoring or base system for interpreting quiz results, and there is no way to aggregate results from multiple lessons for summative assessment purposes (Relkin vd., 2020).

Preparing Instructional Materials and Games on Computational Thinking and Coding

Nowadays there is an extensive range of computer hardware and software available on the educational and toy market which makes it even more difficult to make informed choices in selecting these materials. Instructional materials should be appropriate for the age, emotional and social development, and ability level of the children for whom the materials are selected. Instructional materials should differentiate with respect to levels of difficulty and be of a wide variety.

Principles of preparing instructional materials in preschool

The Developmentally Appropriate Technology in Early Childhood (DATEC) project (Siraj-Blatchford, J. & I. 2002, 2006) identified seven general principles for determining the effectiveness of ICT applications – or uses of ICT – in the early years, to help practitioners provide the best possible experiences. These are still relevant today and can be used as a useful tool to evaluate software programmes or other ICT applications:

1. Ensure an educational purpose
2. Encourage collaboration
3. Integrate with other aspects of the curriculum
4. Ensure the child is in control
5. Choose applications that are transparent (ie their functions should be clearly defined and intuitive so the application can complete each task in a single operation)
6. Avoid applications containing violence or stereotyping
7. Be aware of health and safety issues

Teachers can decide which instructional materials are most likely to help children achieve the learning goals outlined in the curriculum and contribute their knowledge of how children learn, how to manage a classroom learning environment, and the particular challenges of the preschool curriculum when preparing instructional materials in preschool.
Instructional materials make learning more interesting, practical, realistic and appealing. They also enable both the teachers and students to participate actively and effectively in lesson sessions. They give room for acquisition of skills and knowledge and development of self- confidence and self- actualization. (IJHSSI, 2017)

IMs are an important element within the curriculum and are often the most tangible and visible aspect of it (Nunan, 1991)

They can provide a detailed specification of content, even in the absence of the syllabus (Richards and Rogers, 1986)

They can define the goals of the syllabus and the roles of the teachers and the learners within the instructional process (Wright, 1987)

By offering children an imaginative, engaging, introduction to computational thinking and coding you help them make solid steps towards understanding the world.

The best practice is where activities and resources:
- are imaginative and fun,
- challenge,
- involve being creative,
- require collaboration and sharing,
- involve listening, understanding, following and giving instructions,
- encourage describing, explaining and elaborating,
- encourage investigation,
- involve problem solving,
- include lots of ‘unplugged’ activities: computing without computers.

Assessment tools for preschool materials

To assess the instructional materials you are implementing in the teaching-learning process you can used our 8-step guide.

1. Consider the age of the children – each age group is different in the sense of what children are capable of and what kind of developmental stage they are in – sensitive periods etc.
2. Consider the interests of the children – maybe you have noticed that children are interested in some specific type of animated movie or they like some book or particular type of toy.
3. Compliance with the curriculum – consider what learning goals are planned to be reached and what knowledge you have planned to actualize at the time.
4. Active learning
    i. Activity goal – how many learning goals can be reached using this material? Can it be used for just one purpose only?
    ii. Activity type – what cognitive activities are enabled using this material?
5. Feedback
    i. Proportion of feedback – how will you as a teacher give feedback to a child when he/she will be using the material?
    ii. Feedback delivery method – is there any automatic way to provide feedback (right answers to the other side of the material / “answer sheet” / only one way the task can be completed)?
6. Complexity of the learning environment – how many elements does the material consist of or how many elements are displayed on the material? Are they distracting from or

helping to use the material?

7. Design

i. Visual – how complex visual design is used on the material – colors, items, movements?

ii. Touch – are there different textures used in the material? Are they helpful in reaching the learning goal? If you are evaluating a digital app – how many different types of gestures have to be used to use an app – tapping, swiping, dragging, tracing and is it intuitive?

iii. Sound – is there any sound used in the material? Is it helpful in reaching the learning goal? If it is an digital app are there instructions described audially?

8. Design sophistication – What is the overall feel of the material? If this is a digital App can everything be easily found in it?

Learning Environment in Preschool

A learning environment is the context in which learning happens. Learning occurs in environments that promote collaboration, exploration and discovery. An environment respects the agency of "rich and powerful learners" (Edwards, Gandini & Forman, 2012), inspires creativity and innovation, and recognizes experimentation and failure as an integral part of the learning process. The child impacts the environment (Bronfenbrenner, 1979) and is, in turn, impacted by the environment.

Learning environments consist of multiple learning spaces. They can be natural and man-made; physical and virtual; indoors and outdoors; formal and informal. Environments also include the relationships among and between the children and adults and the materials and resources.

Learning spaces have to be flexible, to allow for planned and spontaneous opportunities, for quiet, independent and interactive small group learning, and spaces for children to be. Flexible learning spaces take into account children as individuals and are responsive to diverse needs, backgrounds, abilities and interests. If children can be involved and consulted in the design of their learning spaces, they will feel greater ownership over their learning. This supports well-being, a sense of belonging, and mutual engagement for adults and children alike.

Preschool teachers should adapt the environment and resources to support children's emerging needs and developing interests.

Children may not use an area because it is overcrowded and confusing for them. An unattractive, chaotic, and noisy environment is likely to hype up children's behaviour so they become disruptive and disrespectful of the environment, and the materials and equipment within it. On the other hand, environments that are too pristine and immaculately tidy do not provide enough challenges for children. Children who are bored, who have their creativity stifled by too many controls in the environment, and who are not challenged enough will also manifest disruptive and disrespectful behaviour.

Digital and non-digital technology literacies are integral to designing learning spaces as they extend when, where and how learning and teaching takes place. Effective design provides space for children to safely learn about technology and through technology. Adequate access to digital technologies and the internet is increasingly important to support children's inquiries as they develop in the classroom or other learning spaces.

How to Bring Changes in Preschool Education?

With the rapid development and spread of technology, the definition of literacy, digital literacy skills, and sharing information have altered significantly. Computational and digital skills are accepted as the fundamental elements of twenty-first century literacy skills (Papadakis, Kalogiannakis, & Zaranis, 2016). The National Curriculum in England addressed computational skills, stating that young children (aged 5–7) need to know basic program concepts and commands and to “understand algorithms, and use technology purposefully to create, organise, store, manipulate and retrieve digital content” (Department of Education, 2013). The digital era requires the use of technical skills not only in work and education settings, but also in daily life. Today, children have opened their eyes to the digitized world, one in which technology is evolving rapidly, creating new forms of knowledge, and requiring new skills and abilities (Yang, Hwang, Yang, & Hwang, 2015). The National Association for the Education of Young Children (NAEYC, 2017) shared information about why introducing coding to preschoolers could be beneficial. In this context, there has been some recent attention on developmentally integrating appropriate technology practices into early childhood education programs (Barr et al., 2011; Bers, Flannery, Kazakoff, & Sullivan, 2014; NAEYC & Fred Rogers Center for Early Learning and Children’s Media, 2012).

Young children use new devices, from smartphones and tablets to digital toys, and they are exposed to digital environments from an early age (Altun, 2017; Berns et al. 2014; Parette, Quesenberry, & Blum, 2010), but most of the research and applications regarding computational skills deal with school-age children (Barr & Stephenson, 2011; Durak & Saritepeci, 2018; Grover & Pea, 2013; Lye & Koh, 2014). However, previous studies have pointed out that children as young as four to six years old can understand basic programing concepts, learn to code, engage with simple robotic projects, and build programs (Berns et al., 2014; Cejka, Rogers, & Portsmore, 2006; Kazakoff, Sullivan, & Bers, 2012; Papadakis et al., 2016; Wyeth, 2008).

Therefore it will be inevitable to consider some necessary updates in teacher education programs. In other words, countries should integrate computational thinking and coding subjects into preschool education undergraduate programs. In this way, it will be possible to improve preschool teacher undergraduates’ computational thinking and coding skills. Because we need trained preschool teachers so that they can improve computational thinking and coding skills of preschool children. This change in teacher education curricula should be addressed holistically and systematically. All aspects such as providing basic knowledge, improving skills and attitude, instructional material development, methods to be used in classroom activities, assessment and resources etc. should be taken into consideration to ensure an effective change in teacher education programs in this man-

ner. The results/outputs of the EarlyCode Project covers an innovative curriculum on computational thinking and coding, a handbook of teaching materials and a manual for lecturers. For this reason, all outputs should be utilized for a beneficial change.
The following recommendations can be given to bring about change in preschool education:

To improve computational thinking and coding skills of children, we should start improving the knowledge and skills of teachers, educators and preschool teacher candidates.

Teacher educators, preschool teachers and preschool teacher candidates should be provided with all necessary resources, sample activity plans and teaching materials.
All stakeholders should be informed about importance of computational thinking and coding education at preschool level.

Preschool teacher candidates should be encouraged to be creative in developing innovative and original teaching materials and activities for children.
Best practices on computational thinking and coding education in preschool level should be shared with stakeholders.

Policy makers should keep in mind that improving computational thinking and coding skills of children will result in a shift in their future abilities.

Conclusion

This Manual has been written by the partners of the Erasmus+ European Project "EARLYCODE – Developing Teaching Material for Preschool Teaching Undergraduates on Computational Thinking and Introduction to Coding" (Project Number 2018-1-TR01-KA203-058832) which has the main aim of fostering and developing computational and algorithmic thinking in the early years.

The Training Manual on Computational Thinking and Introduction to Coding for Preschool Education has been developed to guide trainers and lecturers to implement the curriculum, which is also an intellectual output of the project, in preschool education programs.

Send any comments by an email to
earlycoderseu@gmail.com
You can find more information about the project at
www.earlycoders.org

Project Coordinator
Kırşehir Ahi Evran University (Turkey)
IO-3 Responsible Partner
University of Latvia (Latvia)
Other Partners
Scuola di Robotica (Italy)
EarlyYears (Northern Ireland)
University of Bucharest (Romania)
Gazi University (Turkey)
A.P.E.C. (Turkey)

Resources for existing preschool computational thinking materials

There are a multitude of ICT resources available and many free downloadable programmes and free apps for tablets and smart phones. Teachers should ensure that they adhere to the principles outlined above when selecting resources.

1. preschoolsteam.com
2. teachyourkidscode.com
3. teachoutsidethebox.com
4. code.org
5. scratch.mit.edu
6. kodable.com
7. codemonkey.com
8. https://www.commonsense.org/education/search?contentType=reviews
9. https://www.nagc.org/resources-supporting-computational-thinking-primary-grades
10. https://www.nsta.org/connected-science-learning/connected-science-learning-april-june-2020/creating-preschool
11. http://www.icompute-uk.com/news/tag/computational-thinking/

References

1. Ackermann, E. (2001). Piaget's constructivism, Papert's constructionism: What's the difference. Future of Learning Group Publication, Vol 5 p 438.
2. Alimisis, D. (2013). Educational robotics: Open questions and new challenges. Themes in Science & Technology Education, Vol 6, pp 63-71.
3. Altun, D. (2017). Young children's literacy habits in digital world regarding digital equality perspective. 69th OMEP World Assembly and International Conference, Opatija, Croatia.
4. Altun, D., & Ulusoy, M. (2017). Using printed and electronic children's picture books in education process in classroom practices. 26th International Congress on Educational Sciences, Antalya, Turkey.
5. Angel-Fernandez, J., M., Vincze, M. (2018). Philipp Zech, Justus Piater (Eds.) Proceedings of the Austrian Robotics Workshop 2018, pp 37-42. Innsbruck university press, ISBN 978-3-903187-22-1, DOI 10.15203/3187-22-1
6. Arfé, B., Vardanega, T., Ronconi, R. (2020). The effects of coding on children's planning and inhibition skills. Computers & Education, Volume 148. ISSN 0360-1315. https://doi.org/10.1016/j.compedu.2020.103807.
7. Atjonen, P. (2014). Teachers' views of their assessment practice. The Curriculum Journal, 25:2, 238-259, DOI: 10.1080/09585176.2013.874952
8. Atkins, D. E., Bennett, J., Brown, J. S., Chopra, A., Dede, C., Fishman, B., Williams, B. (2010). Transforming American education: Learning powered by technology. Learning. 114
9. Backus-Naur form. (n.d.). In Appendix 1 of a Springer publication. Retrieved from https://link.springer.com/content/pdf/bbm%3A978-1-4612-5983-1%2F1.pdf
10. Baird, Catherine, & Henninger, Maureen. (2011). Serious play, serious problems: Issues with ebook applications. Cosmopolitan Civil Societies Journal, 3(2), 1–17.
11. Bartolotta, T., Shulman, B. (2013). Child development. Language Development: Foundations, Processes, and Clinical Applications. Jones & Bartlett Publishers, 35-53. http://samples.jbpub.com/9780763747237/47238_CH02_Shulman.pdf
12. Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: what is Involved and what is the role of the computer science education community?. Acm Inroads, 2(1), 48-54.
13. Ben-Ari M., Mondada F. (2018). Robots and Their Applications. Elements of Robotics. Springer, Cham. https://doi.org/10.1007/978-3-319-62533-1_1
14. Bers, M. U. (2018a). Coding as a Playground. London and New York: Routledge Press.
15. Bers, M., U. (2018b). Coding and Computational Thinking in Early Childhood: The Impact of ScratchJr in Europe. European Journal of STEM Education, vol 3. ISSN: 2468-4368.
16. Bers, M. U., Flannery, L., Kazakoff, E. R., Sullivan, A. (2014). Computational thinking and

tinkering: Exploration of an early childhood robotics curriculum. Computers & Education, Volume 72, Pages 145-157. ISSN 0360-1315. https://doi.org/10.1016/j.compedu.2013.10.020
17. Bers, M.U., González-González, C. & Armas–Torres, M.B. (2019). Coding as a playground: Promoting positive learning experiences in childhood classrooms. Computers & Education, 138(1), 130-145. Elsevier Ltd. Retrieved March 18, 2021 from https://www.learntechlib.org/p/209930/
18. Bers, M., Seddighin, S. & Sullivan, A. (2013). Ready for Robotics:Bringing together the T and E of STEM in early childhood teacher education. Journal of Technology and Teacher Education, 21(3), 355-377. Waynesville, NC USA: Society for Information Technology & Teacher Education. Retrieved March 18, 2021 from https://www.learntechlib.org/primary/p/41987/.
19. Borawska-Kalbarczyk K., Tołwińska B., Korzeniecka-Bondar A. (2019). From Smart Teaching to Smart Learning in the Fast-Changing Digital World. In: Daniela L. (eds) Didactics of Smart Pedagogy. Springer, Cham
20. Bognar B., Sablić M., Škugor A. (2019). Flipped Learning and Online Discussion in Higher Education Teaching. In: Daniela L. (eds) Didactics of Smart Pedagogy. Springer, Cham
21. Bourn, D., Hunt, F., Ahmed, H. (2017). Childhood development stages and learning on global issues. https://assets.publishing.service.gov.uk/media/5a7447eeed915d0e8e398742/253_global_learning.pdf
22. Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. Içinde AERA (C. 2012).
23. Brevik, L.M., Gudmundsdottir, G.B, Lund, A., Strømme, T.A. (2019). Transformative agency in teacher education: Fostering professional digital competence, Teaching and Teacher Education, Volume 86, 102875, ISSN 0742-051X, https://doi.org/10.1016/j.tate.2019.07.005.
24. Britannica, T. Editors of Encyclopaedia (2021, January 20). Grace Hopper. Encyclopedia Britannica. https://www.britannica.com/biography/Grace-Hopper
25. Bronfenbrenner, U. 1979. The ecology of human development: Experiments by nature and design. Cambridge, MA, USA. Harvard University Press.
26. Buckleitner, W. (2009). What should a preschooler know about technology? Early Childhood Today.
27. Cambridge University. (n.d.). Code. In Cambridge English Dictionary. Cambridge University Press.
28. Cathcart, A., Greer, D., Neale, L. (2014). Learner-focused evaluation cycles: facilitating learning using feedforward, concurrent and feedback evaluation. Assessment & Evaluation in Higher Education, 39:7, 790-802, DOI: 10.1080/02602938.2013.870969
29. Cejka, E., Rogers, C., & Portsmore, M. (2006). Kindergarten robotics: Using robotics to motivate math, science, and engineering literacy in elementary school. International Journal of Engineering Education, 22(4), 711.
30. Centrs Dardedze (2020). Kā notiek bērna attīstība. http://drosaberniba.lv/emacibas/kursi/modulis-1
31. Chang, C. W., Lee, J. H., Chao, P. Y., Wang, C. Y., & Chen, G. D. (2010). Exploring the pos-

sibility of using humanoid robots as instructional tools for teaching a second language in primary school. Educational Technology & Society, 13(2), 13–24.
32. Chaudron S., Di Gioia R., Gemo M. (2018). Young children (0-8) and digital technology, a qualitative study across Europe; EUR 29070; doi:10.2760/294383
33. Chen, N. S., Quadir, B., & Teng, D. C. (2011). A Novel approach of learning English with robot for elementary school students. In M. Chang et al. (Eds.), Edutainment 2011, LNCS 6872 (pp. 309–316). Heidelberg, Germany: Springer-Verlag Berlin Heidelberg.
34. Chiong, C., & Shuler, C. (2010). Learning: Is there an app for that? Investigations of young children's usage and learning with mobile devices and apps. New York: The Joan Ganz Cooney Center at Sesame Workshop.
35 Chontelle Bonfiglio. (2021, January 02). 5 coding concepts That 5-Year-Olds can understand. Retrieved March 19, 2021, from https://teachyourkidscode.com/coding-for-kindergarten-5-basic-coding-concepts-5-year-olds-can-understand/
36. Çiftci, S., Bildiren, A. (2020). The effect of coding courses on the cognitive abilities and problem-solving skills of preschool children, Computer Science Education, 30:1, 3-21, DOI: 10.1080/08993408.2019.1696169
37. Council of the European Union. (2018). Council Recommendation of 22 May 2018 on key competences for lifelong learning. Official Journal of the European Union.
38. Crick, R.D. (2007). Learning how to learn: the dynamic assessment of learning power. The Curriculum Journal, 18:2, 135-153, DOI: 10.1080/09585170701445947
39. Dagdilelis, V., Satratzemi, M., & Evangelidis, G. (2004). introducing secondary education students to algorithms and programming. Education and Information Technologies 9, p 159–173. Kluwer Academic Publishers. DOI https://doi.org/10.1023/B:EAIT.0000027928.94039.7b
40. Daniela. L. (2019). Smart Pedagogy for Technology-Enhanced Learning. Didactics of Smart Pedagogy, Springer Nature Switzerland AG doi:10.10007/978-3-030-01551-0
41. Department of Education (2013) . National Curriculum in England: Computing Programmes of Study. Retrived from https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study
42. Dijkstra, E.W. (1973). A simple axiomatic basis for programming languages constructs. EWD; Vol. 372. International Summer School.
43. Domingues-Montanari, S. (2017). Clinical and psychological effects of excessive screen time on children. Journal of Paediatrics and Child Health, Vol. 53, Issue 4, pp 333-338. DOI https://doi.org/10.1111/jpc.13462
44. Durak, H. Y., & Saritepeci, M. (2018). Analysis of the relation between computational thinking skills and various variables with the structural equation model. Computers & Education, 116, 191-202.
45. EARLY - Education Advancements through Robotics Labs for Youth. (n.d.). Robotics database. Retrieved April 8, 2021 from https://edurobots.eu/robotics-database/
46. Edwards, C, Gandini, L and Forman, G. 2012. The hundred languages of children: The

Reggio Emilia experience in transformation (Third edition). Santa Barbara, CA, USA. Praeger.
47. Elnebija, I. Pakāpieni bērna attīstībā. Rīgā: Pētergailis.
48. Engle, R.A. (2006). Framing Interactions to Foster Generative Learning: A Situative Explanation of Transfer in a Community of Learners Classroom, Journal of the Learning Sciences, 15:4, 451-498, DOI: 10.1207/s15327809jls1504_2
49. Erikson, E., H. Childhood and Society. New York: Norton, 1950.
50. e-Media project Consortium. (2019). Educational Robotics. Retrieved April 8, 2021 from https://all-digital.org/resources/educational-robotics-handbook/
51. Gimbert, B. Cristol, D. (2004). Teaching Curriculum with Technology: Enhancing Children's Technological Competence During Early Childhood. Early Childhood Education Journal, Vol. 31, No. 3.
52. Gladwin, L., A. (1997). Alan Turing, Enigma, and the breaking of German machine chiphers in World War II. Prologue, vol 29, n 3 pp 203-217.
53. Green, H., Facer, K., Rudd, T., Dillon, P., Humphreys, P. (2005). Futurelab: Personalisation and digital technologies.
54. Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. Educational Researcher, 42(1), 38-43.
55. Guizzo, E. (2018, August 01). What Is a Robot? Top roboticists explain their definition of robot. IEEE Robots, your guide to the world of robots. https://robots.ieee.org/learn/what-is-a-robot/
56. Hačatrjana, L. (2019). Using Technologies to Teach Different Age Groups Meaningfully. In L. Daniela (ed.), Didactics of Smart Pedagogy. DOI https://doi.org/10.1007/978-3-030-01551-0_5
57. HarperCollins. (n.d.). Robot. In HarperCollins COBUILD Advanced English Dictionary, online. Retrieved April 8, 2021 from https://www.collinsdictionary.com/dictionary/english/robot
58. Hemmendinger, D. (2021, January 29). Computer programming language. Encyclopedia Britannica. Retrieved on 7th of April, 2021 from https://www.britannica.com/technology/computer-programming-language
59. Heudin, J. (2007). Les créatures artificielles: Des automates aux mondes virtuels [Artificial creatures: From automata to virtual worlds]. Odile Jacob. p. 73
60. Hoare, C.A.R. (October 1969). An axiomatic basis for computer programming. Communication of the ACM, Vol 12, n 10, pp 576-583. Doi https://doi.org/10.1145/363235.363259
61. Hooper, L. (1910). Hand-loom weaving. Pitman Publishing Limited.
62. Instefjord, E.J., Munthe, E. (2017). Educating digitally competent teachers: A study of integration of professional digital competence in teacher education, Teaching and Teacher Education, Volume 67, Pages 37-45, ISSN 0742-051X, https://doi.org/10.1016/j.tate.2017.05.016.
63. Kazakoff, E., Sullivan, A. and Bers, M.U. (2013) 'The effect of a classroom-based intensive robotics and programming workshop on sequencing ability in early childhood', Early Childhood Education Journal, 41(4), 245–255

64. Knuth, D., E., Pardo, L., T. Early development of programming languages. Encyclopedia of Computer Science and Technology. Marcel Dekker vol 7 pp 419–493.
65. Kolb, D.A. (2015). Experiential Learning: Experience as the Source of Learning and Development. Pearson Education, Inc, New Jersey https://books.google.lv/books?id=jp-beBQAAQBAJ&dq
66. Kolb, D., Kolb, A. (2013). The Kolb Learning Style Inventory 4.0: Guide to Theory, Psychometrics, Research & Applications.
67. Komis, V., Misirli, A. (2016). The environments of educational robotics in Early Childhood Education: towards a didactical analysis. Educational Journal of the University of Patras UNESCO Chair Vol 3 pp 238-246. ISSN: 2241-9152.
68. Krumsvik, R. (2011). Digital competence in the Norwegian teacher education and school. Högre Utbildning. 1. 39-51.
69. Lamrani, R., Abdelwahed, E., H. (2020). Game-based learning and gamification to improve skills in early years education. Computer Science and Information Systems, Vol 27, Issue 1, pp 339-356. DOI: https://doi.org/10.2298/CSIS190511043L
70. Lee, J. (2020). Coding in early childhood. Contemporary Issues in Early Childhood, 21(3), 266–269. https://doi.org/10.1177/1463949119846541
71. Lee, J., Junoh, J. (2019). Implementing Unplugged Coding Activities in Early Childhood Classrooms. Early Childhood Educ J Vol 47, pp 709–716. https://doi.org/10.1007/s10643-019-00967-z
72. Levine, L.E., Munsch, J. (2017). Child Development: An Active Learning Approach. SAGE Publications, Inc. https://edge.sagepub.com/levine3e
73. Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12?. Computers in Human Behavior, 41, 51-61.
74. Marinus, E., Powell, Z., Thornton, R., McArthur, G., & Crain, S. (2018). Unravelling the cognition of coding in 3-to-6-year olds: The development of an assessment tool and the relation between coding ability and cognitive compiling of syntax in natural language. ICER 2018 - Proceedings of the 2018 ACM Conference on International Computing Education Research, 18, 133–141. https://doi.org/10.1145/3230977.3230984
75. Merriam-Webster. (n.d.). Robot. In Merriam-Webster.com dictionary. Retrieved April 8, 2021 from https://www.merriam-webster.com/dictionary/robot
76. Mioduser, D., & Levy, S. T. (2010). Making Sense by Building Sense: Kindergarten Children's Construction and Understanding of Adaptive Robot Behaviors. International Journal of Computers for Mathematical Learning 2010 15:2, 15(2), 99–127. https://doi.org/10.1007/S10758-010-9163-9
77. Moore, T. J., Brophy, S. P., Tank, K. M., Lopez, R. D., Johnston, A. C., Hynes, M. M., & Gajdzik, E. (2020). Multiple Representations in Computational Thinking Tasks: A Clinical Study of Second-Grade Students. Journal of Science Education and Technology 2020 29:1, 29(1), 19–34. https://doi.org/10.1007/S10956-020-09812-0

78. Moravcik, M. Pekarova, J. Kalas, I. (2008). Digital Technologies at Preschool: Class Scenarios. Word paper for co-located conferences at the WCC 2008 congress.
79. Mueller, J., Beckett, D., Hennessey, E., & Shodiev, H. (2017). Assessing Computational Thinking Across the Curriculum. Emerging Research, Practice, and Policy on Computational Thinking, 251–267. https://doi.org/10.1007/978-3-319-52691-1_16
80. National Association for the Education of Young Children (NAEYC), (2017). Creating coding stories and games Retrieved from https://www.naeyc.org/resources/pubs/ tyc/feb2017/creating-coding-stories-and-games
81. National Research Council. (2015). Professional Learning for the Care and Education Workforce. Washington, DC: The National Academies Press. https://doi.org/10.17226/21786.
82. Oladoke, O. A. (2006). Measurement of self directed learning in online learners. (Doctoral dissertation). Capella University, Minnesota, MN.
83. Oxford University. (n.d.). Code. In Oxford English Dictionary. Oxford University Press.
84. Oxford University. (n.d.). Coding. In Oxford English Dictionary. Oxford University Press.
85. Papadakis, S., Kalogiannakis, M., & Zaranis, N. (2016). Developing fundamental programming concepts and computational thinking with ScratchJr in preschool education: a case study. International Journal of Mobile Learning and Organisation, 10(3), 187-202.
86. Papadakis, S. (2020). Robots and Robotics Kits for Early Childhood and First School Age. International Association of Online Engineering. Retrieved April 23, 2021 from https://www.learntechlib.org/p/218338/
87. Papert, S. (1980). Mindstorms: children, computers, and powerful ideas. Basic Books, Inc.
88. Parette, Howard P.; Quesenberry, Amanda C.; & Blum, Craig. (2010). Missing the boat with technology usage in early childhood settings: A 21st century view of developmentally appropriate practice. Early Childhood Education Journal, 37, 335–343.
89. Perret, P. (2015). Children’s Inductive Reasoning: Developmental and Educational Perspectives. Journal of Cognitive Education and Psychology, 14 (3), pp.389 - 408. https://hal-amu.archives-ouvertes.fr/hal-01772202/document
90. Piaget, J. (1974). To Understand is to Invent. Basic Books.
91. Poundstone, W. (2021, February 4). John von Neumann. Encyclopedia Britannica. https://www.britannica.com/biography/John-von-Neumann
92. Rasooli, A., Zandi, H., DeLuca, C. (2019). Conceptualising fairness in classroom assessment: exploring the value of organisational justice theory. Assessment in Education: Principles, Policy & Practice, DOI: 10.1080/0969594X.2019.1593105
93. Redecker, C. (2017). European Framework for the Digital Competence of Educators: DigCompEdu. Punie, Y. (ed). Publications Office of the European Union, Luxembourg, ISBN 978-92-79-73494-6, doi:10.2760/159770, JRC107466
94. Relkin, E., de Ruiter, L., & Bers, M. U. (2020). TechCheck: Development and Validation of an Unplugged Assessment of Computational Thinking in Early Childhood Education. Journal of Science Education and Technology 2020 29:4, 29(4), 482–498. https://doi.org/10.1007/

S10956-020-09831-X
95. Resnick, M. (2017). Lifelong Kindergarten: Cultivating Creativity through Projects, Passion, Peers, and Play. MIT Media Press.
96. Resnick, M. (2018). Lifelong Kindergarten: Cultivating Creativity Through Projects, Passion, Peers, and Play. Erickson.
97. Ricketts, R. (2018). Computational thinking for kindergarteners.
98. Rochkind, M., J. (2004). Advanced Unix Programming, Second Edition. Addison-Wesley. p. 1.1.2.
99. Røkenes, F.M., Krumsvik, R. (2014). Development of Student Teachers' Digital Competence in Teacher Education - A Literature Review. Nordic Journal of Digital Literacy. 9. 250-280.
100. Sant'Anna School of Advanced Studies – Pisa. (n.d.). Educational robotics. Retrieved April 8, 2021 from https://www.santannapisa.it/en/institute/biorobotics/educational-robotics
101. Schugar, H. R., Smith, C. A., & Schugar J. T. (2013). Teaching with interactive picture e-books in grades K–6. The Reading Teacher, 66(8), 615-624. DO - 10.1002/trtr.1168
102. Stewart, T. Walker, C. Berry, S. (2019). Effects of screen time on preschool health and development. Ministry of Social Development. ISBN Online 978-1-98-854155-6.
103. Sugimoto, M. (2011). A Mobile mixed-reality environment for children's storytelling using a handheld projector and a robot. IEEE Trans Learning Technologies, 4(3), 249-260
104. Sullivan, A., Bers, M., U. (2016) Robotics in the early childhood classroom: Learning outcomes from an 8-week robotics curriculum in pre-kindergarten through second grade. International Journal of Technology and Design Education 26, pp 3–20.
105. Tedre, M, (2014). The Science of Computing: Shaping a Discipline. Chapman and Hall/CRC.
106. Toh, L. P. E., Causo, A., Tzuo, P. W., Chen, I., & Yeo, S. H. (2016). A review on the use of robots in education and young children. Journal of Educational Technology & Society, 19(2), 148-163.
107. Vasquez, V. and C. Felderman (2013). Technology and critical literacy in early childhood. New York: Routledge.
108. Wang, D., Wang, T., & Liu, Z. (2014). A tangible programming tool for children to cultivate computational thinking. The Scientific World Journal, 2014. https://doi.org/10.1155/2014/428080
109. Wanner, T., Palmer, E. (2018). Formative self-and peer assessment for improved student learning: the crucial factors of design, teacher participation and feedback. Assessment & Evaluation in Higher Education, 43:7,1032-1047, DOI: 10.1080/02602938.2018.1427698
110. Wartella, E., Blackwell, C.K., Lauricella, A.R. & Robb, M.B. (2013). Technology in the lives of educators and early childhood programs. Latrobe, PA: Fred Rogers Center for Early Learning and Children's Media at Saint Vincent College
111. Wyeth, P. (2008). How young children learn to program with sensor, action, and logic

blocks. The Journal of the learning sciences, 17(4), 517-550.
112. Yang, T.C., Hwang, G.J., Yang, S.J. and Hwang, G.H. (2015) 'A two-tier test-based approach to Improving students' computer-programming skills in a web-based learning environment', Education Technology & Society,18 (1), 198–210.
113. Zaranis, N., Kalogiannakis, M. and Papadakis, S. (2013) 'Using mobile devices for teaching realistic mathematics in kindergarten education', Creative Education. 4(7A), 1–10.
114. Zhu, Z., Yu, M. & Riezebos, P. (2016). A research framework of smart education. Smart Learn. Environ. 3, 4 https://datubazes.lanet.lv:4876/10.1186/s40561-016-0026-2
115. Zigler, E. F., & Bishop-Josef, S. J. (2006). The cognitive child vs. the whole child: lessons form 40 years of Head Start. In D. G. Singer, R. M. Golinkoff, & K. Hirsh-Pasek (Eds.), Play= learning: How play motivates and enhances children's cognitive and social-emotional growth (pp. 15–35). New York, NY: Oxford University Press