Available online at ijci.wcci-international.org

IJCI
International Journal of Curriculum and Instruction

International Journal of Curriculum and Instruction 14(1) (2022) 1052–1070

The coding skills of pre-school teacher candidates

Ayşegül Zeynep Ergin[a] *, Zülfiye Gül Ercan[b]

[a] *Tekirdağ University, Graduate School of Educational Sciences, Edirne, Turkey*

[b]*Trakya Üniversity, Faculty of Education, Edirne, Turkey*

Abstract

Computer science works as hardware and software increasingly takes place in each of our daily lives with effective products and services. As in all areas, it is an inevitable part of learning and teaching computer-based environment-tools and equipment in the fate and type of education, including early childhood. Computer science products may require extensive hardware, hardware, and software, or it can be very simple in terms of level and purpose. It is possible to create products and services with coding to be done using advanced programming languages, as well as coding techniques that are simple to use for teaching children in education in early childhood education institutions. In this way, it is possible for children to realize what they want as a producer not only as a consumer, but with a more permanent learning. At the same time, learning of coding techniques for preschool teachers to use in the education of children in these age slices will make the education more participatory, collaborative, effective and successful. The "coding skills" that preschool teacher candidates will have will provide important benefits in their professional development. Therefore, it is important to investigate the knowledge, skills, and attitudes of preschool teacher students about coding education. The aim of this study is to determine the coding knowledge, skills, experience, and opinions of preschool teacher candidates at Trakya University Faculty of Education about this subject. A questionnaire consisting of 30 questions was applied to 213 students studying in the fall semester of 2019-2020 academic year at Trakya University Faculty of Education, Department of Basic Education Preschool teaching program. As a result of the research, it has been determined that preschool teacher candidates do not have sufficient knowledge, skills and experience in coding and do not show positive attitudes about coding education.

Keywords: Coding Education, Preschool Teacher Candidates

* Corresponding author Ayşegül Zeynep Ergin.
E-mail address: zeynepergin@gmail.com

1. Introduction

Programming has been one of the main and compulsory in computer science education. It has deep links with electronic and mechanical systems that relies on algorithms to create code. The computer programming offers insights into natural and artificial systems.

The introduction of concepts of Industry 4.0, which emerged with the rapid change in technology, has change the curriculum framework that countries expect from individuals who have flexibility and adaptability to manufacturing systems.

In this context, in today's World where development communities are experienced, code education has an important role in having necessary skills for survival of individuals. Therefore, many countries have adapted to the code education into their curriculum.

It is believed that code education is gaining importance, particularly in relation to coding environment for young learners. Code education can be embedded into code learning. curricula, with a primary goal being that students move through being active consumers of technology that they come to see themselves as active inventors of technology. Indeed, the code education is now accepted as an important skill for students to learn before they enter global labor market.

Coding is not only the fundamental skill for computer science, but it is also developing computational thinking skills, including problem solving, communication, collaborative working, planning, decision making, analyze, critical and creative thinking skills of students, while also supporting their lifelong success in workplace. Therefore, coding skills, which are one of the fundamental components of meta cognition and regarded among 21st century skills, are important for the lifelong success.

Early childhood education (Pre-K to 1) has long recognized and agreed upon the benefits of using code education methodologies to help young children learn by technology enhanced effective curriculum, by manipulating technology-based instruction. In the last decade, there has been an increase in demand for, and the controversies over, introducing computers in early childhood education settings. However, very few teachers have the experience and skill to conduct this kind of activity. In the best cases, they are facilitators how to use some computer applications, but haven't developed true technological fluency thus, professional development requirements of teachers became very important for code education. Many, teachers when faced with the challenge of using computers in the classroom, revert to instructions ways of teaching and learning. They lack the needed training and expertise. Mainly due to the code education, Professional developments prepare teachers in the area of technology, and they offer them a vision in which teachers see themselves as designer of technology enhanced curricula.

The aim of this study is to determine the coding knowledge, skills, experience, and opinions of preschool teacher candidates at Trakya University Faculty of Education about code education. This study will explore the following three research questions:

1. What are pre-school teacher candidates' knowledge and skill level about Coding Education?

2. What are pre-school teacher candidates' perception about Coding Education?

3. Is there a relationship between the perceptions of pre-school teacher candidates in terms of their grades?

1.1. Coding Education

1.1.1. The Role of Teachers

A professional code of ethics outlines teachers' primary responsibilities to their students to be actively guiding them with open-ended questions that stimulate students' inquiry and reflection skills as well as increasing their motivational resources. The activities in code education foster students to evaluate their thinking process. These metacognitive activities facilitate the process of applying a newly acquired checking, planning, selecting, monitoring, inquiring, reflection, collaboration and responsible decision making. evaluating and revising. These strategies are parts of important components for communication about learning models while discussing metacognitive activities in the classroom (Galen& Buter,2000).

Students who possess metacognitive management skills with new technology and strategies perform well in programming compared to those in whom these skills are lacking. Moreover, the more difficult the problem in programming, the higher the demand for scaffolding by teachers, positive feedback, and purposeful reflection within collaborative learning (Clemenets,2002:172).

1.1.2. Computational Thinking (CT)

Computational Thinking (CT) is seen as an important competence that is required in order to adapt to the future. It was found that the promotion of CT in education has made great progress in the last decade. It was also found that CT has mainly been applied to the activities of program design and computer science, while some studies are related to other subjects. Although Paper (1996) had already used the term "Computational Thinking" (CT) ten years earlier, the current discussion of CT can be traced back to Wing (2006) (Kafai & Burke 2014a: 2). Wing characterized CT by stating that it "involves solving problems, designing systems, and understanding human behavior, by drawing on the concepts fundamental to computer science" (Sayın & Seferoğlu, 2016).

It has created the following working definition of CT as a 21 century skill: The conceptual foundation required to solve problems effectively and efficiently (Tağci, 2019). Wing (2006:33) refined her first statement pointing out that the essence and key of CT is primarily a way of thinking and acting, which can be exhibited through the use particular skills, which then can become the basis for performance-based assessments of CT skills. Based on the literature, we categorized CT into six main facets: decomposition, abstraction, algorithm design, debugging, iteration, and generalization.

The consecutive create phase is essential for the appropriation of CT skills, giving learners the opportunity to be creative and express themselves through collaborative learning in while the students solve problems creating new computational artifacts in line with algorithmic thinking ((Korkmaz, Çakır & Özden, 2017: 15).

Computational Thinking is a problem-solving process that includes (but is not limited to) the following characteristics (Barr, Harrison & Conery (2011).

Formulating problems in a way that enables us to use a computer and other tools to help solve them;

- Logically organizing and analyzing data
- Representing data through abstractions, such as models and simulations
- Automating solutions through algorithmic thinking (a series of ordered steps)
- Identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources
- Generalizing and transferring this problem-solving process to a wide variety of problems.

The first three bullet points highlight the importance of problem solving, organizing and analyzing data and computational modeling as part of CT process. General approaches to CT are typically facilitated by increasing interest in programming and code education (Kafai & Burke 2014b).

The Concepts of Programming and Code Programming is basically a process of translating from the language convenient to aid learning including analyzing, problem solving with algorithmic thinking (Michael & Omolove 2014).

"Programming is the process of drawing up the schedule of the sequence of individual operations required to language convenient to human beings to the language convenient to the computer. Coding is the sequence of orders is known as the program, and the machine performs it automatically without intervention from the individual (Sayın & Seferoğlu, 2016).

1.1.3. The Importance of Coding

Based on programming which is an incredibly important skill that all individual today will need to know as well as coding. In many countries, there are software projects in this field constituted by support from companies, governments, and universities such as Scratch, code.org. Coding is a system that complex problems can be fixed in sequence with responsible decision making. Given the extent to which technology generates our lives, learning to code helps develop a better understanding of the world around us and prepare children an almost innate ability to adapt to future life thus, it is important teach ii within early ages (Aytekin, vd., 2018).

Learning coding is like learning a language of creativity which lays an innovation and ability to connect well established ideas with all new solutions and new concepts. Many tasks we can perform easily when children spent time with computer, so why not let our children to learn coding. Programming is not only learning aid of coding but also information technologies also important to tech to children (Özcan, 2017).

It is suggested that fun activities are those which can support fun ways to teach kids to code that don't only involve typing code. Early coding is generated as fun and exciting to stimulate children motivation to coding. The earlier children explore the basics of coding, the more easily they will be able to learn, understand, and apply coding are just the right age and level to begin learning. It is suggested that some aspects of precoding that would be meaningful to preschoolers are those who are lack of responsible decision making and algorithmic thinking skills than other children (Aytekin vd., 2018).

1.1.4. The Advantages of Code Education

Coding can be described as a computational thinking process of developing algorithmic thinking skills to solve a complex mathematical of problems (Taylor, Harlow & Forret, 2010) Coding can enhance individuals' computational thinking for solving problems as well as critical thinking.

Code.org and Scratch are the 2 most popular platforms for young coders learning to code for the first time. They both can support children problem solving, computational thinking and creative skill. platforms enabling children build what them imagine instead of learning complex programming while stimulate children's' problem solving, computational thinking and creative skill (Aytekin vd., 2018).

Coding can enhance individuals' cognitive skills. Coding can stimulate students' critical thinking, responsible decision making, reflective and problem-solving skills (Wachenchauzer, 2004).

Coding can foster productivity of students' while students create a new image then they can be desired to create new one so, they develop self-awareness and self-regulation skill while they criticize their weakness and strengths while coding (Howland & Good 2015). Logo and image programming can foster students' productive skill that's' why students develop positive attitudes and interest towards to coding (Minuto, Pittarello & Nijholt 2015).

1.1.5. Coding Skills

Computers can enhance meta cognitive skills of students. Computer users are those who are in preschool get higher point according to metacognitive criterions, holding strength memories and creating cognitive theories. than non-computer users at the same age. Some researchers suggested that logo development can strength conceptual understanding and problem-solving process of students are those who are at pre and primary school (Clements, 1990; Fletcher-Flinn & Suddendorf, 1996).

It suggested that coding can stimulate " learning how to learn". In addition, it fosters literacy and mathematical communication as well as metacognitive skills if children starts coding at early ages (Clements, 2002: 166-167).

Some researchers suggested that programming course is not explained as rote memorization of coding however, design and create a new while solving problems in appropriate sequences (Wang, Huang & Hwang, 2016: 33-34).

1.1.6. Code Education Approaches

Coding education mainly has divided in four approaches such as visual programming, robotics, text-based coding, and coding without computer (Bower & Folkner 2015).

1.1.7. Text-Based Programming

It defies classical programming language (C++, Fortran, Cobol, VBasic, Java vb.) using command line.

1.1.8. Robotics

Robotics allows students to see their thinking in a real way as they go through trial and error until the task is accomplished and the robot's motions are performed as originally intended instead of classical programming course therefore, students get motivation and interests towards to programming when perform their image (İnce, 2018).

1.1.9. Coding without Computer

Coding without computers refer to lessons in which students are not working on a computer as "unplugged." In unplugged lessons, aid of learning is not focus on algorithmics, computer-individual connection or databased instead students often work with pencil and paper or physical manipulatives. These are intentionally placed kinesthetic opportunities that help students digest complicated concepts in ways of cards, games, and puzzle these are relate to their own lives and foster to learn to main information technologies concepts (Kalelioğlu & Keskinkılıç 2017).

1.1.10. Visual Programming

Visual programming has been stared as logo programming then it has been continued within Code.org, Scratch platforms. Visual programming fosters early learners with easily build an image except from complex coding activities (Resnick vd., 2009). Programming languages is best taught according to students' developmental needs for visual programming. Children develop gaming, gaming- style coding, creating animation and story (Taylor, Harlow & Forret, 2010). The most useful resources about visual programs are Alice, Code org and Scratch. In addition, Tynker, Codable and Touch develop are also useful resources for visual programming. These platforms can enhance coding without using any complex programming language. However, they foster productivity of children while they create animation and new things in virtual environment (Sayın & Seferoğlu, 2016).

Virtual platforms such as Toon Talk, Squeak Etoys, Stage cast Creator, Microworlds JR, Scratch and Code.org can support children to create interactive games, animations, simulations, and stories in line with easily programming language. Visual programming aims to develop students' thinking skills and stimulate their interest towards to coding get higher their motivation to learn (Özcan, 2017).

1.1.11. Coding for Kids

There are many platforms for children to coding from five years of age upwards (Baz,2016)

Table 1. Coding platforms for kids

Coding platform	Web site	Age
code.org	https://code.org/	5
Scratch	www.scratchjr.org/	5-11
Codable	www.kodable.com	5-7
The Foo's	http://thefoos.com/	5-7
Tynker	https://www.tynker.com/	5-7
Box Island	https://boxisland.io/	5-7
Cargo Bot	https://itunes.apple.com/tr/app/cargo-bot	5-7

Daisy Dinosaur	https://itunes.apple.com/us/app/daisy-the- dinosaur/	5-7
Blocky	https://developers.google.com/blockly/	5-7
Move the Turtle	http://movetheturtle.com/	5-7

(Baz, 2016)

1.1.12. Scratch

Scratch is the world's largest coding community for children and a coding language with a simple visual interface that allows young people to create digital stories, games, and animations (Wang, Huang & Hwang, 2016: 35).

Scratch has been designed, developed, and moderated by Massachusetts Technology Institute (MIT) Media Lab (http://scratch.mit.edu/).Scratch has been developed by idea of Seymour Paper who is effected to logo programming language(Taylor, Harlow & Forret, 2010: 562).

Children snap together graphical programming blocks to make characters move, jump, dance, and sing. Children can modify characters in the paint editor, add their own voices and sounds, even insert photos of themselves -- then use the programming blocks to make their characters come to life. Scratch was inspired by the popular Scratch programming language as Lifelong Kindergarten and MIT Media Lab (scratch.mit.edu) (Shin, Park & Bae: 2013: 246).

In creating Scratch, programming language has been combined with education that is amazing opportunity for code education therefore, students are get more interest and motivation towards to visual programming (Neumajer, 2012). Developing literacy of Scratch can be solution of many code education problems because students can enhance creativity and collaborative writing skills while programming ((Williams & Cernochova, 2013: 18).

Scratch is limited for collaborative functioning and learning thus, Scratch team develop 2.0 version to reduce these limitations using Clutter. In addition, children who are get more ability towards to IT may be isolate themselves from society thus Scratch combined itself with Clutter to develop individuals' creativity and characteristic features (Shin, Park & Bae, 2013: 249).

1.1.13. Scratch Jr

ScratchJr is a collaboration between MIT Media Lab, Human Development at Tufts University, and the Playful Invention Company. Children snap together graphical programming blocks to make characters move, jump, dance, and sing. ScratchJr is inspired from Lego and building them with playing (Kaplancalı & Demirkol, 2017: 34).

1.1.14. Clutter

Clutter was a website created by Scratch Team, that allowed users to put Scratch projects together in a sequence called a "clutter", much like a slideshow. Clutter allowed users to do the following three things include: Order projects sequentially, like galleries, enter a secret word to move to the next project in the clutter and link to another project in the clutter within a project (Shin, Park & Bae 2013: 247)

1.1.15. Codable

Codable is suitable for children who are more than five years. There is no need to literacy to play. Codable is version of Scracth which is designed from 5 to 7 years of children. It is free to use can play within iPad and Android software. In addition, LightbotJr is also fit from 4 to 8 years of children. Moreover, Robot Turtles is a board game which is focus on learning programming without using any technological functions (Rey, 2019).

1.1.16. Tynker

Tynker is a visual programming language and medium of multimedia writing. Children use visual code blocks to understand logic and programming concepts. Tynker includes " coding time" page to develop creativity of children offering games and story activities to children. In school settings, Tynker may be used as project work and objective orientation while enhance basic programming information to students (Kaplancalı & Demirkol, 2017: 34).

1.1.17. Google Blockly

Google Blockly is a Google library for the programming language JavaScript for creating block-based visual programming languages. Block programming allows users to create scripts and programs by using visual blocks, even if they do not know any programming language. Blockly includes everything you need for defining and rendering blocks in a drag-n-drop editor. Each block represents a chunk of code that can be easily stacked and translated into code. It can be used to let users independent within time and space (Çalışır, 2016).

1.1.18. Code.org

In 2013, Code.org was launched by twin brothers Hadi and Ali Partovi with a video promoting computer science. They have encouraged children, particularly school students to learn and love computer science. Code. org aims to learn coding to children with playing games. Code.org may enhance children's' algorithmics skills while teach them cycle theories and stimulate problem solving skill.

Code.org is a nonprofit organization. Children can snap together graphical programming blocks making command sequences. Partovi said that the command sequences are framework of learning programming. Blocky is visual code platform and open library and free for user (Kaplancalı & Demirkol, 2017: 34).

It has since become a worldwide effort to celebrate computer science, starting with 1-hour coding activities but expanding to all sorts of community efforts! Over the past years since its inception in 2013, the Hour of Code has reached tens of millions of students in 180+ countries. In hour of code, more than 20 million students make up 668.008.671 code line. Celebrities, tech visionaries and even Former President Obama support Hour of Code. In addition, Code studio is a web site which is managed by code.org storage the resources of code for children. Code studio is not limited for one hour thus is not like function with hours of code (Kaplancalı & Demirkol, 2017: 34).

2. Method

2.1. Participants of The Study

The participants of the study consisted of preschool teacher candidates at Trakya University Faculty of Education about this subject. A questionnaire consisting of 30 questions was applied to 213 students studying in the fall semester of 2019-2020 academic year at Trakya University Faculty of Education, Department of Basic Education Preschool teaching program.Include in these subsections the information essential to comprehend and replicate the study. Insufficient detail leaves the reader with questions; too much detail burdens the reader with irrelevant information. Consider using appendices and/or a supplemental website for more detailed information.

Table 2. Frequency Distribution

		f	%
Grade	a) 1.	75	35,2
	b) 2.	58	27,2
	c) 3.	50	23,5
	d) 4.	30	14,1
Age	a) 19 and smaller	73	34,3
	b) 20-21	78	36,6
	c) 22-23	38	17,8
	d) 24 and older	24	11,3
Gender	a) Female	189	88,7
	b) Male	24	11,3
	Total	213	100,0

2.2. Data Collection

Data were collected from a questionnaire developed by author conducted for determine the coding skills perceptions of students consisted of 30 questions was applied to 213 students studying in the fall semester of 2019-2020 academic year at Trakya University Faculty of Education. The questionnaire was divided into two parts include:

1. Personal Information Form of Candidacy Teachers: It consists of such questions to the candidacy teachers about their age, gender, professional development about the topic, academic average

2. Perceptions about Code Skills of Candidacy Teachers: It consists of such questions to candidacy of teachers to determine whether their perceptions about code skills, their experiences within code skills, they get professional development about code skills, they have interest towards to code skills

Quantitative data was analyzed in the study.

3. Results

3.1. Results According to Participants' Grade level

There was significant relationship between professional development about code skills and grade level (x^2=9,272 sd=3 p=0,026). Accordingly, respondents (%13,3) who are at 4th grade of preschool teaching students had professional development about code skills however, respondents who are at 1st,2nd,3rd grade of preschool teaching students had no any professional development about code skills. See table 3.

Table 3. Professional Development About Code Skills According to Grade Level

		a) 1.	b) 2.	c) 3.	d) 4.	Total
a) Yes	f	1	2	1	4	8
	%	1,3%	3,4%	2,0%	13,3%	3,8%
b) No	f	74	56	49	26	205
	%	98,7%	96,6%	98,0%	86,7%	96,2%
Total	f	75	58	50	30	213
	%	100,0%	100,0%	100,0%	100,0%	100,0%
		chi square	sd	p		
		9,272[a]	3	0,026		

There was no significant relationship between context of professional development of preschool teaching student and their grade level (x^2=3,000 sd=3 p=0,392). Respondents (%50,0) who had professional development about coding skills (f=8) said that they had the professional development in computer course from Educational Faculty. However, respondents (%50,0) defined as other. See table 4.

Table 4. The Context of The Professional Development About Code Skills According to Grade Level

		a) 1.	b) 2.	c) 3.	d) 4.	Total
b) Computer course in faculty	f	0	1	0	3	4
	%	0,0%	50,0%	0,0%	75,0%	50,0%
c) Other	f	1	1	1	1	4
	%	100,0%	50,0%	100,0%	25,0%	50,0%
Total	f	1	2	1	4	8
	%	100,0%	100,0%	100,0%	100,0%	100,0%
		chi square	sd	p		
		3,000[a]	3	0,392		

There was significant relationship between interest towards to professional development about code skills of preschool teaching students and grade level (x^2=11,872 sd=3 p=0,008).

Accordingly, respondents (%92,0) who were at 3rd grade and respondents (%86,7) who were at 4th grade answered the question of ‘’do you want to have professional development about code skills”? as Yes. In addition, respondents (%69,3) who were at 1st grade and respondents (%70,7) who were at 2nd grade gave an answer similar like others as Yes. See table 5.

Table 5. Interest of Getting Professional Development About Code Skills According to Grade Level

		a) 1.	b) 2.	c) 3.	d) 4.	Total
a) No	f	23	17	4	4	48
	%	30,7%	29,3%	8,0%	13,3%	22,5%
b) Yes	f	52	41	46	26	165
	%	69,3%	70,7%	92,0%	86,7%	77,5%
Total	f	75	58	50	30	213
	%	100,0%	100,0%	100,0%	100,0%	100,0%
		chi square	sd	p		
		11,872[a]	3	0,008		

There was relationship between interest getting higher level of professional development about code skills of preschool teaching students and grade level (x^2=8,000 sd=3 p=0,046). Accordingly, respondents (f=8 Table 7) who were expressed that they had professional development about code skills was very lower. In addition, respondents who were at 2nd, 3rd grade interested to had higher level of professional development about code skills however, respondents who were at 1st,4th grade no interest about getting higher level of professional development about code skills. See table 6.

Table 6. Interests of Preschool Teaching Students Getting Higher Level of Professional Development about Code skills According to Grade level

		a) 1.	b) 2.	c) 3.	d) 4.	Total
a) No	f	1	0	0	4	5
	%	100,0%	0,0%	0,0%	100,0%	62,5%
b) Yes	f	0	2	1	0	3
	%	0,0%	100,0%	100,0%	0,0%	37,5%
Total	f	1	2	1	4	8
	%	100,0%	100,0%	100,0%	100,0%	100,0%
		chi square	sd	p		
		8,000[a]	3	0,046		

Respondents (%79,81; f=170) who gave an no answer to the question of do you know any coding program? If said yes write it down. However, respondents (%11,74) who gave an answer and wrote some code program name. Moreover, respondents (%6,57) did not interest to the question. See table 7.

Table 7. Frequency Distribution According to Interested Coding Program

	f	%
robotic	9	4,23
scratch	2	0,94
Coding without computer	3	1,41
code.org, tree house	13	6,10
total	27	12,68
Programming languages c, c++,c#, php, pascal, python,R,java	25	11,74
other (edx, udemy, google developer,spss, autocad, atsima, codex	14	6,57
boş	170	79,81
	225	110,80

Respondents (%94,84) who gave no answer to question of do you know using any code program? However, respondents (%2,82) who gave answer as code.org and respondents (%0,47;f=1) who gave answer as stem for similar question. See Table 8.

Table 8. Knowing to Use Any Code Program

	f	%
dull	202	94,84
code.org	6	2,82
stem	1	0,47
ewievs	1	0,47
linux	1	0,47
spss	1	0,47
autocad	1	0,47
Total	213	100,00

There was relationship between experiences of any coding activities of preschool teaching students at internship and grade level (x^2=9,577 sd=3 p=0,023).

Accordingly, respondents (%10,0) who were at 3rd grade and respondents (%6,7) who were at 4th grade gave an answer is Yes. However, respondents (%1,3) who were at 1st and respondents (%0,0) who were at 2nd grade gave answer is Yes. See table 9.

Table 9. Experiences of Any Coding Activities of Preschool Teaching Students at Internship According to Grade Level

		a) 1.	b) 2.	c) 3.	d) 4.	Total
a) No	f	74	58	45	28	205
	%	98,7%	100,0%	90,0%	93,3%	96,2 %
b) Yes	f	1	0	5	2	8
	%	1,3%	0,0%	10,0%	6,7%	3,8%
Total	f	75	58	50	30	213
	%	100,0%	100,0%	100,0%	100,0%	100,0 %
		chi square	sd	p		
		9,577[a]	3	0,023		

There was relationship between an awareness of preschool teaching students' about teaching and learning material about code education related to preschool education and grade level (x^2=19,932 sd=3 p=0,000).Accordingly, respondents(%20,0) who were at 3rd grade, they were aware of teaching and learning materials about code education related to preschool education and respondents (%13,3) who were at 4th grade they were aware of teaching and learning materials about code education related to preschool education. However, respondents (%2,7) who were at 1st grade and 2nd grade (%0,0), they were a few or no aware of teaching and learning materials about code education related to preschool education. See table 10.

Table 10. Preschool Teaching Students' Awareness of Teaching and Learning Material about Code Education Related to Preschool Education according to Grade Level

		a) 1.	b) 2.	c) 3.	d) 4.	Total
a) No	f	73	58	40	26	197
	%	97,3%	100,0%	80,0%	86,7%	92,5%
b) Yes	f	2	0	10	4	16
	%	2,7%	0,0%	20,0%	13,3%	7,5%
Total	f	75	58	50	30	213
	%	100,0%	100,0%	100,0%	100,0%	100,0%
		chi square	sd	p		
		19,932[a]	3	0,000		

There was relationship between preschool teaching students' interest towards to social media or internet resource about code education and grade level (x^2=35,633 sd=3 p=0,000). Respondents (%38,0) who were at 3rd grade interested social media or internet resources about code education. However, respondents who were at 1st grade (%4,02),2nd grade (%6,9) and 4th grade (%6,7), they were weakly interested towards to social media and internet resources about code education. See table 11.

Table 11. Preschool Teaching Students' Interest Towards to Social Media or Internet Resource about Code Education According to Grade Level

		a) 1.	b) 2.	c) 3.	d) 4.	Total
a) No	f	72	54	31	28	185
	%	96,0%	93,1%	62,0%	93,3%	86,9%
b) Yes	f	3	4	19	2	28
	%	4,0%	6,9%	38,0%	6,7%	13,1%
Total	f	75	58	50	30	213
	%	100,0%	100,0%	100,0%	100,0%	100,0%
		chi square	sd	p		
		35,633[a]	3	0,000		

There was relationship between preschool teaching students' awareness of code activities without being computer users and grade level (x^2=33,894 sd=3 p=0,000). Respondents (%34,0) who were at 3rd grade and 4th grade (%23,3), they were aware of code activities without being computer users. However, Respondents (%4,0) who were at 1st grade and 2th grade (%1,7), they were weakly aware of code activities without being computer users. See table 12.

Table 12. Preschool Teaching Students' Awareness of Code Activities Without Being Computer Users According to Grade Level

		a) 1.	b) 2.	c) 3.	d) 4.	Total
a) No	f	72	57	33	23	185
	%	96,0%	98,3%	66,0%	76,7%	86,9%
b) Yes	f	3	1	17	7	28
	%	4,0%	1,7%	34,0%	23,3%	13,1%
Total	f	75	58	50	30	213
	%	100,0%	100,0%	100,0%	100,0%	100,0%
		chi square	sd	p		
		33,894[a]	3	0,000		

4. Discussion and Conclusions

Respondents who were at 3rd and 4th grade they more aware of code education and coding material than respondents who were 1st and 2nd grade. In the study, respondents (%63,4) were no more awareness of code education. It could be weak of professional education about information technologies during bachelor's educational settings. During the bachelor's degree, students had no sufficient information about Microsoft Office Program, therefore, they can only enhance individual capacity instead of career development. In addition, the students had not been studied about programming and code education during their bachelor's degree. Thus, it is urgent need to add teaching modules to related curriculum.

According to results of the study, there was significant relationship between professional development about code skills and grade level of preschool teaching students. Accordingly, respondents who are at 4th grade of preschool teaching students had professional development about code skills however, respondents who are at 1st,2nd,3rd grade of preschool teaching students had no professional development about code skills. In addition, there was no significant relationship between context of professional development of preschool teaching student and their grade level. Respondents who had professional development about coding skills said that they had the professional development in computer course from Educational Faculty. However, respondents defined as other. Moreover, there was significant relationship between interest towards to professional development about code skills of preschool teaching students and grade level. Accordingly, respondents who were at 3rd grade and respondents who were at 4th grade answered the question of "do you want to have professional development about code skills"? as Yes. In addition, respondents who were at 1st grade and respondents who were at 2nd grade gave an answer similar like others as Yes.

In the study, there was relationship between interest getting higher level of professional development about code skills of preschool teaching students and grade level. Accordingly, respondents who were expressed that they had professional development about code skills was very lower. In addition, respondents who were at 2nd, 3rd grade interested to had higher level of professional development about code skills however, respondents who were at 1st,4th grade no interest about getting higher level of professional development about code skills. Respondents who gave an no answer to the question of do you know any coding program? If said yes write it down. However, respondents who gave an answer and wrote some code program name. Moreover, respondents did not interest to the question.

There was a relationship between the awareness of preschool teaching students about teaching and learning materials of coding education related to preschool education and grade level. Accordingly, respondents who were at 3rd grade, they were aware of teaching and learning materials about code education related to preschool education and respondents who were at 4th grade they were aware of teaching and learning materials about code education related to preschool education. However, respondents who were at 1st grade and 2nd grade, they were a few or no aware of teaching and learning materials about code education related to preschool education.

Considering to the results of the study, it may be suggested to enhance preschool teaching students' educational needs of code education listed below:

- It may be added as a course to related curriculum,
- It may be added as an optional course to required discipline,
- It may be added as an additional training about code education within related course,
- It may be added as an additional project work and homework about code education to related courses,
- It may be constructed coding laboratory to the related faculties,
- It may be stimulated preschool teaching students' interest towards to using internet resources about code education,
- It may be enhanced research that may be conducted on curriculum and instructional needs about code education,
- They may be suggested to enhance academic studies for researchers about code education,
- Research may be conducted on capacity of teachers about code education at other faculties,
- Research may be conducted on as a comparative study to compere efficacy of coding resources and teaching and learning materials,
- Research may be conducted on content and instructional needs of code education.

References

Aytekin, Alper. Sönmez Çakır, Fatma. Yücel, Yakup Bahadır. Kulaözü, İlknur (2018). Geleceğe yön veren kodlama bilimi ve kodlama öğrenmede kullanilabilecek bazi yöntemler. Avrasya Sosyal ve Ekonomi Araştırmaları Dergisi (ASEAD), 5 (5): 24-41

Barr, David. Harrison, John & Conery, Leslie (2011). Computational thinking: A dijital age skill for everyone, http://files.eric.ed.gov/fulltext/ EJ918910.pdf [Erişim tarihi: 21.10.2020.]

Baz, Fatih Çağatay (2016). Çocuklar için kodlama yazılımları üzerine karşılaştırmalı bir inceleme, Current Research in Education 4 (1), 36-47

Bower, Matt & Falkner, Katrina (2015). Computational thinking, the notional machine, pre-service teachers, and research opportunities. Proceedings of the 17th Australasian Computing Education Conference (ACE 2015), Australia, 27-30 January, 37-46.

Clements, Douglas (1990) Metacomponential development in a Logo programming environment. Journal of Educational Psychology, 82: 141-149.

Clements, Douglas (2002). Computers in early childhood mathematics. Contemporary Issues in Early Childhood, Volume 3, Number 2

Çalışır, Esma Çukurbaşı (2016). Google Blockly ile çocuklar için kodlama. https://esmacalisir.com/2016/09/08/google- blockly-ile-cocuklar-icin-kodlama/ [Erişim Tarihi: 21/10/2020].

Fletcher-Flinn, Claire & Suddendorf, Thomas (1996) Do computers affect 'the mind'? Journal of Educational Computing Research, 15: 97-112.

Galen, Frans van & Buter, Arlette (2000) Computer tasks and classroom discussions in mathematics, International Congress on Mathematics Education (ICME-9), Tokyo/Makuhari, Japan.

İnce, Ebru Yılmaz (2018). Önlisans öğrencilerinin kodlama eğitiminde robotik sistemlerinin kullanımına yönelik görüşleri.

Akdeniz Eğitim Araştırmaları Dergisi, 12(25), 326-341. doi: 10.29329/mjer.2018. 153.17

Kafai, Yasmin B. & Burke, Quinn (2014a). Mindstorms 2.0 children, programming and computational participation.

Constructionism Conference 2014

Kafai, Yasmin B. & Burke, Quinn (2014b). Connected Code: Why Children Need to Learn Programming. The MIT Press.

Kalelioğlu, Filiz & Keskinkılıç, Fatma (2017). Bilgisayar bilimi eğitimi için öğretim yöntemleri. Yasemin Gülbahar (Ed.), Bilgi

İşlemsel Düşünmeden Programlamaya, Pegem Akademi, Ankara, 155-178.

Kaplancalı, Uğur Tevfik & Demirkol, Zafer. (2017). Teaching coding to children: a methodology for kids 5+ International Journal of Elementary Education. 6(4): 32-37

Korkmaz, Özgen; Çakır, Recep & Özden, M. Yaşar; (2017). A validity and reliability study of the Computational Thinking

Scales (CTS), Computers in Human Behavior. DOI: 10.1016/j.chb.2017.01.005

Michael, Adu & Omoloye, A b e (2014). Improving structural designs with computer programming in building construction. IOSR Journal of Computer engineering (IOSR- JCE), 16(3): 10-16.

Minuto, Andrea. Pittarello, Fabio & Nijholt, Anton (2015). Smart material ınterfaces for education. Journal of visual languages and computing, 31, 267-274. https://doi.org/10.1016/j.jvlc.2015.10.006

https://www.sciencedirect.com/science/article/pii/S1045926X15000634?via%3Dihub [Erişim Tarihi: 22.10. 2020]

Neumajer, Ondřej. (2012) Further teacher education in ICT (Další vzdělávání učitelů v oblasti ICT). https://clanky.rvp.cz/clanek/s/Z/ 16139/DALSI-VZDELAVANI-UCITELU-V-OBLASTI-ICT.html/ [Erişim Tarihi: 22.10. 2020]

Özcan, Betül (2017) Çocuklar için kodlama eğitimi. http://kodlamaegitim. blogspot.com/2017/01/cocuklar-icin-kodlama- egitimi.html [Erişim Tarihi: 22.10.2020]

Resnick, Mitchel. Maloney, John. Hernández, Andrés Monroy. Rusk, Natalie. Eastmond, Evelyn. Brennan, Karen. Millner, Amon. Rosenbaum, Eric. Silver, Jay. Silverman, Brian. Kafai, Yasmin (2009). Scratch: programming for everyone. Communications of the ACM. (52): 60-67

Rey, Dustine (2019). The importance of coding in early childhood. http://www.drdustinerey.com/the-importance-of-coding- in-early-childhood/ [Erişim Tarihi: 15.11.2019]

Sayın, Zehra & Seferoğlu, S. Sadi. (2016). Yeni bir 21. yüzyıl becerisi olarak kodlama eğitimi ve kodlamanın eğitim politikalarına etkisi. XVIII. Akademik Bilişim Konferansı, 3-5 Şubat 2016, Adnan Menderes Üniversitesi, Aydın.

Shin, Seungki. Park, Phanwoo & Bae, Youngkwon (2013). The effects of an ınformation-technology gifted program on friendship using scratch programming language and clutter. International Journal of Computer and Communication Engineering. 2 (3)

Tağci, Çiğdem (2019). Kodlama eğitiminin ilkokul öğrencileri üzerindeki etkisinin incelenmesi. Afyon Kocatepe Üniversitesi Fen Bilimleri Enstitüsü Yüksek Lisans Tezi

Taylor, Merilyn. Harlow, Ann & Forret, Michael (2010). Using a computer programming environment and an ınteractive whiteboard to ınvestigate some mathematical thinking. ınternational conference on mathematics education research. Procedia Social and Behavioral Sciences. 8 (2010):561–570

Wachenchauzer, Rosita (2004). Work in progress promoting critical thinking while learning programming language concepts and paradigms. 34th ASEE/IEEE Frontiers in Education Conference. Savannah, GA, 20-23 October.

Wang, Hsiu-Ying. Huang, Iwen & Hwang, Gwo-Jen (2016). Comparison of the effects of project-based computer programming activities between mathematics-gifted students and average students. J. Comput. Educ. (2016) 3(1):33–45 DOI 10.1007/s40692-015-0047-9 [Erişim Tarihi: 15.11.2019]

Williams, Lawrence & Cernochova, Miroslava (2013). Literacy from Scratch. X World Conference on Computers in Education July 2-5, 2013; Toruń, Poland

Wing, Jeannette M. (2006). Computational thinking. Communications of the ACM. 49(3): 33-35.