

EXPLORATIONS ON THE NATURE OF CHILDREN'S
CONCEPTUAL CHANGE IN COMPUTATIONAL THINKING

During Hello Ruby Summer School 2016

Milla Kruskopf

Master's Thesis

Cognitive Science

December 7, 2016

University of Helsinki

HELSINGIN YLIOPISTO – HELSINGFORS UNIVERSITET – UNIVERSITY OF HELSINKI

| | |
|---|--|
| Faculty of Behavioural Sciences – Käyttäytymistieteellinen tiedekunta | Institute of Behavioural Sciences – Käyttäytymistieteiden laitos |
| Milla Kruskopf | |
| Explorations on the Nature of Children’s Conceptual Change in Computational Thinking During Hello Ruby Summer School 2016 Lasten ohjelmoinnillisen ajattelun käsitteellinen muutos Hello Ruby -kesäkoulun 2016 aikana | |
| Kognitiotiede – Cognitive Science | |
| Kirsti Lonka (HY, Department of Teacher Education), Mirjamaija Mikkilä-Erdmann (UTU, Department of Teacher Education) | 2016 |
| <p>The purpose of this study is to increase scientific understanding of children’s conceptual change in computational thinking during a summer school intervention. With a mixed method approach of self-report questionnaire and interview, the investigation highlights modern children’s knowledge, beliefs and understanding of as well as attitudes, emotions and motivations towards computers, programming and artificial intelligence. Think-aloud –tasks are also used to investigate children’s computational thought processes. The SRQ data with an intervention group (n = 28) and a comparison group (n = 21) was analysed with repeated measures and independent samples t-tests, MANOVA and ANCOVA, with the pretest condition as covariate. The data revealed a change in the intervention group’s conceptions about embedded cyber-physical systems and the application of computers in different industrial and artistic fields. A slight shift towards a strong AI –mindset was discovered in the intervention group through the SRQ. The interviews for the intervention group (n = 6) and two comparison groups (n = 4, material comparison n = 4) reinforced this conclusion and showed a strong enhancement of computational thinking attitudes and perspectives in the intervention group in contrast to the comparison groups. The computational skills were found to be tightly knit to level of mathematical understanding, and didn’t change notably during the summer school intervention.</p> <p>Tutkimuksen tarkoituksena on parantaa tieteellistä ymmärrystä kesäkouluintervention aikana lasten ohjelmoinnillisessa ajattelussa tapahtuvasta käsitteellisestä muutoksesta. Kyselytutkimuksesta ja haastatteluista koostuvalla mixed methods –lähestymistavalla tutkimus valaisee nykylasten tietokoneisiin, ohjelmointiin ja tekoölyyn liittyviä asenteita, tunteita ja motivaatioita sekä uskomuksia ja ymmärrystä. Lisäksi ääneenajattelutehtävien avulla tutkittiin lasten ohjelmoinnillisia ajatteluprosesseja. Kyselydatan (interventio n = 28, vertailu n = 21) varianssianalyysit ja t-testaus paljastivat muutoksen lasten käsityksissä sulautetuista järjestelmistä ja tietokoneiden sovelluksista eri taideteollisilla aloilla. Vahvoissa tekoölynäkemyksissä havaittiin myös lievä nousu interventioryhmän kohdalla. Haastattelut (interventioryhmä n = 6, vertailuryhmä n = 4, materiaali-vertailuryhmä n = 4) vahvistivat tätä johtopäätöstä ja osoittivat myös vahvaa parannusta ohjelmoinnillisissa asenteissa ja näkymissä interventioryhmän kohdalla vertailuryhmiin nähden. Ohjelmoinnillisten taitojen havaittiin olevan tiukasti sidoksissa matemaattisen ymmärryksen tasoon, eikä niissä havaittu huomattavaa muutosta kesäkouluintervention aikana.</p> | |
| Computational thinking, conceptual change, children, programming, technology, artificial intelligence, education Ohjelmoinnillinen ajattelu, käsitteellinen muutos, lapset, ohjelmointi, teknologia, tekoöly, kasvatus | |
| Helsingin yliopiston kirjasto – Helda / E-thesis (opinnäytteet) | <i>ethesis.helsinki.fi</i> |

TABLE OF CONTENTS

| | |
|--|-----------|
| 1 INTRODUCTION..... | 4 |
| 1.1 Computational thinking | 5 |
| 1.2 Conceptual change | 8 |
| 1.3 Learning theories | 10 |
| 1.4 Context of the study: Hello Ruby Summer School | 10 |
| 1.5 Research aims | 13 |
| 2 QUANTITATIVE STUDY | 14 |
| 2.1 Participants..... | 14 |
| 2.2 Materials | 16 |
| 2.3 Procedures..... | 19 |
| 2.4 Results..... | 22 |
| 2.5 Discussion of the results | 24 |
| 3 QUALITATIVE STUDY | 25 |
| 3.1 Participants..... | 25 |
| 3.2 Materials | 25 |
| 3.3 Procedures..... | 26 |
| 3.4 Results..... | 27 |
| 4 DISCUSSION | 36 |
| 4.1 Summary and triangulation of the main results | 36 |
| 4.2 Methodological reflections | 38 |
| 4.3 Conclusion | 39 |
| 5 CONCLUDING REMARKS | 40 |
| 6 APPENDICES..... | 40 |
| 6.1 Appendice 1: The research form..... | 40 |
| 6.2 Appendice 2: The interview frame | 45 |
| 6.3 Appendice 3: The qualitative change t-test results | 48 |
| 6.4 Appendice 4: The ANCOVA tables | 49 |
| 7 REFERENCES..... | 50 |

1 INTRODUCTION

This study intends to increase the scientific understanding of children's ability to grasp fundamental ideas on computational thinking, programming and artificial intelligence while attending a summer course designed by Linda Liukas (2015). The context of the study was a Finnish Hello Ruby summer school for 6-10 year old children in June 2016. In this rapidly digitalizing society, more and more of our day-to-day lives and social interaction is happening in virtual environments that are fundamentally based on some kind of program code and are ever increasingly run by artificially intelligent machine learning algorithms (Williams et.al., 2006). Nevertheless, most of us continue to live our lives happily ignorant of the languages that make up most of our daily working, studying and communication environments, like Facebook, Twitter, different learning platforms and intranets etc.. This because their high level of usability doesn't normally require it of us to dig deeper into the principles behind the intriguing façade (Hakkarainen, Hietajärvi, Alho, Lonka & Salmela-Aro, 2015; Merikivi, 2013; Kupiainen, 2013). Most of even the so-called "digital natives" (Prensky, 2001) born in the virtual era, even though almost constantly connected to these technologies, find them these mysterious magic boxes that just naturally invite them to action. The dividing lines of human-to-computer and human-to-human interaction are also becoming more blurred, minute by minute, as neural network algorithms enable companies to utilize the so-called "chatbots" in their digital communications channels and make virtual assistants more popular and natural to interact with (Bretton, 2016; Oord et.al., 2016). Many find worry in this fact that even the most basic principles of computing are hidden to most of today's technology users, even though they make up such a huge part of our lives today (Williams et.al., 2006).

One of these people is Linda Liukas, who realized these were the kind of things that 21st century children should learn in kindergarten. She had the idea of a children's book designed to help kids understand concepts such as an algorithm, programming language and artificial intelligence through the playful, classic medium of a children's storybook. The idea managed to raise 380 000 \$ on the online crowdfunding service Kickstarter and thus, Hello Ruby became a widely popular phenomenon. (Dredge, 2014; Burn-Callander, 2015; Liukas, 2015.) This in itself is proof of people's concern about this kind of information not being taught as fundamental 21st century citizen skills.

The purpose of this study is to produce scientific knowledge on the important subject of a child's conceptual development in the process of learning about the concepts and paradigms of computer programming at an early age. What kinds of understandings, concepts and mental models do modern children intuitively have about the nature of technology, artificial intelligence and programming? Can technology and programming education (radically) change those intuitive concepts and mental models? The quantity and quality of this conceptual change will be experimentally measured for the first time in this particular study. To Linda Liukas, it is important to include cognitive and educational science in the development of her teaching material, so the results of this study will be intensely utilized in her future business developments, hence, in changing the minds of our future digital natives interacting in the digital environments of our everyday world.

1.1 COMPUTATIONAL THINKING

Computational thinking is currently a hot topic in national curriculum design around the world (Barr & Stephenson, 2011; Mykkänen & Liukas, 2014). It is considered an important 21st century skill by many important players nationally and internationally. It is a common misconception that computational thinking is somehow limited to computers and devices, but this is, in fact, not the case. Many other things include computing like, for example, human behaviour processes, cooking a meal, finding your way back home from work or school, preparing a lesson etc.. As Jeanette Wing, the one who first defined the concept phrased it:

“Computational thinking involves solving problems, designing systems, and understanding human behavior, by drawing on the concepts fundamental to computer science. Computational thinking includes a range of mental tools that reflect the breadth of the field of computer science.” (Guzdial, 2015)

An important concept in the heart of computational thinking is that of an algorithm (Newell, Perlis & Simon, 1967). Algorithm is essentially a sequence of steps or rules, that precisely defines a problem as a sequence of operations (Stone, 1973). Intelligent algorithms and algorithmic thinking have basically already enabled the human species' technological mastery of our natural environment to a remarkable, ever growing extent (e.g. Lukka, Tossavainen, Kujala & Raiko, 2014).

To bring the concept a little closer to our everyday environment, a recipe can be presented as a common real-life example of an algorithm: it is a given set of instructions to get to a certain outcome. Of course, there is a lot more creativity and situational application involved in cooking, lots of confounding variables (e.g. tastes and spices) that can change the outcome in dramatic ways, and algorithms as far as they somehow relate to computers follow a specific set of logical operations, a more precise sort of language, that we will discuss further.

The following is an example of a simple algorithm for making hot chocolate:

1. *Boil water*
2. *Put cocoa powder into a cup*
3. *Pour water into the cup*

These are the basic components of the simple task of making hot chocolate, that are in the world of computation or programming called *modules*. Each module can also consist of other modules, specified to perform a smaller subtask, which can also consist of other subtasks, as in this case, the subtask 2. could consist of subtasks like:

- 2.1 *Find the cocoa powder jar from the cupboard*
- 2.2 *Insert the spoon into the cocoa powder jar*
- 2.3 *Fill the spoon with cocoa powder*
- 2.3 *Insert the spoon into the cup*
- 2.4 *Put away the cocoa powder jar*

One can easily see how also each of these subtasks can be further divided into other subtasks and so on and so on. Since all people have their own preferences as to, for example, how many scoops of cocoa powder they prefer, the number of repeat loops, and also whether they add the water or the cocoa powder first, the organization of the modules, there's many different ways of implementing the specific algorithm for adding cocoa powder. Therefore, there's a lot of place for variation and the coder's creativity involved in computation as well, and this well demonstrates how the way computers "think" and the way people think differ from each other; both are good at some things, and a little worse at others.

One popular existing framework for the programming education of children is the MIT developed Scratch, a visual programming environment for young people to create their own interactive multimedia and share it with people from around the world. ScratchEd, an online community for Scratch educators of the Harvard Graduate School of Education, define computational thinking through the dimensions it involves that they have discovered through studying the activities of the Scratch online community and Scratch workshops: *concepts*, *practices* and *perspectives*. The seven *concepts* they have found to be useful in Scratch projects, are:

| | |
|---------------------|--|
| sequence | series of steps for a task |
| loops | the same sequence running a multiple times |
| parallelism | simultaneous events |
| events | causation between one thing and another |
| conditionals | decisions based on conditions |
| operators | mathematic and logical expressions |
| data | storage, retrieving and updating of values |

Programming is, of course, not only concepts – alone they are empty, just like grammar without words and meaning, out of touch with the natural world. The concepts are expressed through a set of *practices* involved in the act of designing code:

| | |
|-------------------------------------|---|
| experimenting and iterating | developing, trying out, developing further |
| testing and debugging | finding and solving problems if things don't work out |
| reusing and remixing | building on existing projects and ideas created previously or by others |
| abstracting and modularising | exploring the connections between the whole and the parts |

As constantly growing, developing human beings, we naturally learn something new through each action we perform. ScratchEd discovered the shift in thinking or *perspectives* brought about by practicing the art of programming in fact involves three fairly distinctive elements that emerge as we go along our computing practice:

| | |
|--------------------|---|
| expressing | computing as a medium of creative expression |
| connecting | creating in connection with others |
| questioning | asking questions to make sense of the computational things in the world |

Although there are many current debates going on concerning, for example, ethical aspects of robotics and automation, machine learning based artificial intelligence can be seen as the triumph of computational thinking, and the philosophical aspects related to it have been quite actively discussed since the beginning of cognitive science. “Turing test” is a name for an imitation game situation (Turing, 1950), measuring the humanity or strength of artificial intelligence, and to succeed in the test, a human has to be unaware that they’re interacting with a computer. Some philosophers, on the other hand, think it is as sensible to talk about computers understanding language as it is to talk about a lonely translator manually translating Chinese text in a room full of prewritten rulebooks (Searle, 1980). This position has later been dubbed the “Chinese room argument” and it represents a notion of so-called “weak AI”.

The problem of consciousness has in related literature been referred to as the “hard problem of qualia”; We can never truly find out how another person or another animal, such as a bat with an echolocation system, first-hand experiences phenomena such as colour or pain, for example (Jackson, 1986; Nagel 1974). This is why it is just as sensible to consider a computer, which passes the Turing test as conscious, as it is to consider that of anything external to our own consciousness. This view represents a notion of “strong AI”. However, from the perspective of this study, it is an interesting question whether children growing up today, with the current level of technological development, would rather support an idea of strong or weak AI. It is a fact, for example, that computers already have vision and hearing due to deep neural networks and affective computing is making them more and more like sentient beings (Bretton, C., 2016; Picard, R., 1995; Kleine-Cosack, C., 2008).

1.2 CONCEPTUAL CHANGE

Children intuitively have mental models or naïve theories about different phenomena in their everyday environment (Vosniadou 1994; Vosniadou 2013; DiSessa 1998). These are conceptually organized so, that they contain both framework and specific theories. Specific

theories constitute a narrower range of phenomena and can be changed through conceptual enrichment or learning through accretion (Rumelhart & Norman 1978), without changing the framework theory. A computational thinking –related example would be seeing computers as tools to search information or watch cartoons, etc..

Framework theories, however, are much wider, sort of hypotheses of the state of things in the world that are not so easily changed, but require a larger shift in perspective or revision of previously acquired but ill structured knowledge. This shift is also called *conceptual change* and it most often requires more systematic instruction, because the learner may not feel a need for a change in perspective since their framework theory may appear to work in everyday life settings, despite minor contradictions. It has been discovered, for example, that before achieving the conceptual change required to understand photosynthesis, children have naïve theories that plants eat with their roots, just like animals do, and to adopt this new phenomenon into their framework theory, they have to move across ontological categories or paradigms, which can be seen as a more radical conceptual change. (Rumelhart & Norman 1978; Mikkilä-Erdmann 2001; Mikkilä-Erdmann 2002; Penttinen, Anto & Mikkilä-Erdmann 2013; Södervik, Mikkilä-Erdmann & Vilppu 2014; Södervik, Virtanen & Mikkilä-

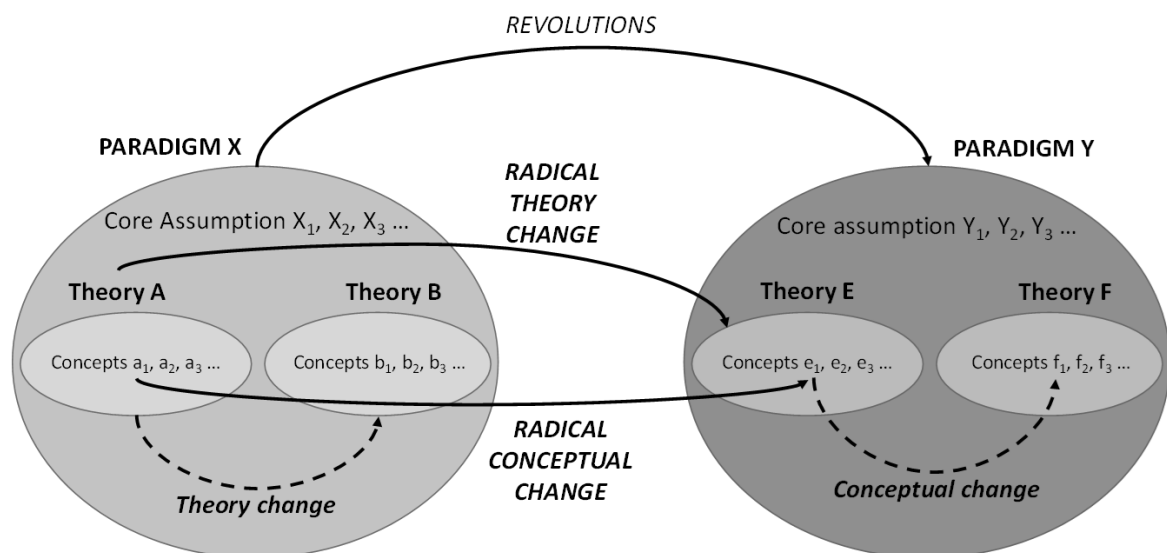


Figure 1. An illustration on the theoretical structure of conceptual change, from Chi 1992. Replacing concepts of a specific theory with concepts from another theory is referred to as conceptual change. When the conceptual change means a leap into a whole other paradigm, it can be seen as radical conceptual change. When entire theories replace each other, this is referred to as (radical) theory change.

Erdmann 2015; Chi, 1992.) A computational thinking –related example would be the shift from seeing computers as tools towards seeing them as assistants or companions with agency and ability to interact with oneself and affect one’s behaviour in a similar way another person would – from a view of weak to strong AI.

1.3 LEARNING THEORIES

Learning theories that emphasize the role of the teacher as an instructor in the conceptual change of the learner can be described as “instructionistic” theories. The legacy of Jean Piaget and Seymour Papert, however, puts the learner at the centre of the process of their own knowledge construction, and also the construction of technology. The *constructivistic* thinking, laid ground to by Piaget, emphasizes the learner’s personal role in the construction of their knowledge and personal view of the world (Lonka 1997). The *constructionistic* thinking of Papert (1972 & 1980) continues this tradition by proposing something similar John Dewey did in the beginning of the 20th century with his experimental learning (Dewey, 2004), that the construction process of technology, actively applying the mathematical ideas into something concrete, helps the learner gain:

“--a greater and more articulate mastery of the world, a sense of the power of applied knowledge and self-confidently realistic image of himself as an intellectual agent” (Papert, 1972)

Seymour Papert left an irreplaceable legacy of educational technologies: LOGO, further developed to Scratch, is an irreplaceable asset in the present situation of an ever-increasing need for computational thinking education that has spurred countless more inventions to help enhance the computational understanding of humanity. His legacy and thinking transforms the view of humans interacting with computers and technology from merely passive users into active constructors of the information (and) technologies around them.

1.4 CONTEXT OF THE STUDY: HELLO RUBY SUMMER SCHOOL

Hello Ruby can be seen as a literary continuation of the legacy of Seymour Papert. Ruby is a curious little girl with a big imagination embarking on a journey to crack the code of a mysterious card her father had left her one day before going off to work. She is a children’s book character created by Linda Liukas as a role model for children to get immersed in the

world of technology, computing and coding in a fun and playful, more inquiry-based way (Hakkarainen, Lonka & Lipponen 2004; Gordon 2014). Her story is intended to introduce small children to the programming logic and culture behind the world they grow up in even before introducing them to a single screen. Ruby's world also includes online environments with exercises that enhance the computational thinking skills of the child. The goal is not only to educate the children on the background logic of computer programming, but also to get them grapple and cultivate the idea of the endless possibilities of expression, creativity and collaboration code allows as a language, in a similar way as crayons and paper or wood and tools. (Hello Ruby, 2016; Resnick 2006; Jacobs & Buechley 2013.)

The fundamental value behind the Hello Ruby universe could be described as “digital equality”, that every child, despite socioeconomic background, gender or race, deserves to discover the world of computing, and to be able to read the language behind its 21st century world. There are massive amounts of data collected of us every second when we interact with technology that is ever increasingly all around us. We think we are the ones using computers and technology for our own benefit, but at the same time there are companies and AI that are constantly learning from us and selling that information forward to marketers, a sort of “invisible hand” that guides our behaviour without us noticing (e.g. Schmitz, 2014). People from diverse backgrounds should be able to understand the possibilities of technology that surrounds us and also contribute to the building of it. That is why Hello Ruby decided in the year 2016 to offer a summer day camp in Helsinki for 6—10 -year-old children to get in touch with the world of code and enhance their computational thinking. Who knows where we will find the next Mark Zuckerberg, Steve Jobs, Elon Musk or Linus Torvalds that will, yet again, change the world as we know it?

The learning goals of Hello Ruby Summer School included two bigger modules of computational thinking and systems thinking (Table 1). Each of these includes specific knowledge, skills and attitudes, values and ethics components that are essential to them.

The 10-day Hello Ruby Summer School 2016 was held on the 6th to 17th of June. The camp included some plugged and many unplugged activities with specific themes related to computing and its various applications. One specific theme was selected per day, and the themes included were computers, home, society, recycling, space, music, robots, data and sensors, communication and the internet. The visitors related to the different themes included

Table 1. The learning goals of Hello Ruby Summer School, consisting of the two modules of computational thinking and systems thinking.

| COMPUTATIONAL THINKING | | | SYSTEMS THINKING | | |
|------------------------|---------------|-----------------------------|--|--|--|
| Knowledge | Skills | Attitudes / Values / Ethics | Knowledge | Skills | Attitudes / Values / Ethics |
| Understanding data | Sequence | Open Source Culture | Understanding and seeing the whole picture of interdependencies | Analyze and interpret information in everyday life through a systems lense | Understanding that the system generates its behaviour and there is no room for blame |
| Cryptography | Loops | Collaboration | Understand systems and strategies for tackling unfamiliar problems | Analyze how parts of a whole interconnect with each other to produce overall outcomes | Understanding myself as a part of the system |
| Data structures | Algorithm | Modularity | | Examining a systems behaviour through different perspectives, stocks, feedback loops (reinforcing, balancing) and delays | |
| | Booleans | | | Drawing causal loops | |
| | Debugging | | | Observing a system's behaviour over time | |
| | Variable | | | Discovering leverage points | |
| | Events | | | | |
| | Modelling | | | | |
| | Abstraction | | | | |
| | Decomposition | | | | |

representatives of Finnish companies from some of the particular fields (ZenRobotics, F-Secure, Elisa and Studio Puisto Architects).

In addition to an included breakfast and lunch, similar to what the children have in Finnish schools, each day consisted of the day's theme first being introduced by a visitor representing the field, indoor and outdoor projects, games and activities related to the theme, and a story time to reflect on what was learned during the day. An example activity for the theme music would be algorithmic painting that included first composing a painting algorithm and then "running" it with the "robot" child to create a beautiful painting. The camp also included a

background story of Blob the alien, acted out by Google translate, that sent the kids messages and to whom they built up a spaceship, part by part, during the camp. At the end of the camp there was an open exhibition to demonstrate the artistic products of the different, more creative type of computing education. The key idea of the entire summer school was to introduce the background concepts essential to computational thinking integrated to issues most probably familiar to the children from their everyday lives through art and play in a safe and engaging environment, and also to encourage friendship, creative confidence and hands-on building and making.

Example day schedule of the Summer School:

8:45-9:15 Flexible arrival to camp.

9:15 Good morning! Breakfast available, getting started for the day ahead.

9:45 Learning about the theme of the day. Exciting visitors and group play.

10:45 Individual or group projects and activities around the theme of the day.

12:00 Lunch time.

12:30 Outdoor play and activities on the theme of the day.

13:30 Story time and reflecting on learning.

14:00 Free play or continue working on individual projects.

15:30-16:00 Flexible pick-up from camp.

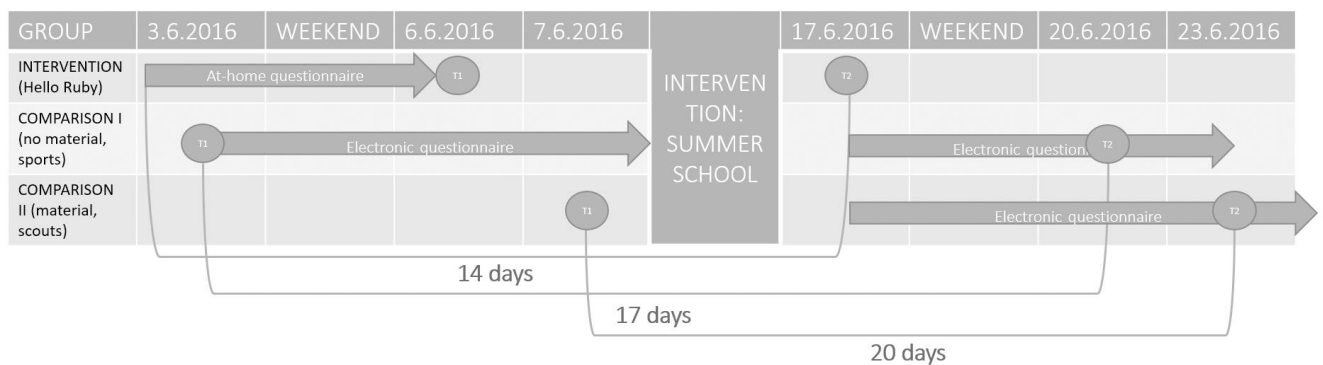
1.5 RESEARCH AIMS

This research was commissioned by Hello Ruby to objectively investigate the efficiency of their first ever computational thinking Summer School. The aim of the research is to provide the answer to the following questions:

- 1. Did any conceptual change take place during a wireless, play-oriented summer school concerning the children's (participants) ideas of technology and their computational thinking. If yes, how did the conceptual change manifest itself on the level of...**
 - a. attitudes, emotions and motivations?
 - b. knowledge and understanding?
 - c. computational skills?
- 2. Did any parallel conceptual change take place in two other contexts, sports camp and scout camp?**

The methods of this research are mixed, including two sorts of data, qualitative and quantitative collected from the participant children in two occasions (Table 2), before and after an “intervention” (Hello Ruby Summer School). This helped us get a more thorough image of the phenomenon of conceptual change regarding computation and AI. (Elmes, Kantowitz & Roediger, 2012.) Next chapters will present the results of these studies separately, with first the quantitative study and second the qualitative study.

Table 2. The course of the study.



2 QUANTITATIVE STUDY

2.1 PARTICIPANTS

The intervention group of the study were 28 6–10 -year-old children (Table 3) who attended the two-week Hello Ruby Summer School that took place in the first two weeks of June 2016 in Helsinki. The whole comparison group, $n = 38$, consisted of children of the same age but attending a different sort of summer camp during the intervention period. Out of the comparison group, part attended a shorter weekend sports camp and didn’t have access to any Hello Ruby material ($n = 21$). Another group of children attended a scout day camp but got Hello Ruby –material to browse for the duration of the intervention period ($n = 17$). In the end, it was decided to use only the sports camp comparison group in the statistical

Table 3. Participant groups of the study and their descriptive parameters. 3 out of Raven’s Progressive Matrices (numbers 1., 30. and 60.) were used to control the non-verbal aspects of intelligent functioning of the children.

| | INTERVENTION GROUP (Hello Ruby Summer School) | COMPARISON GROUP I (Sports camp) | COMPARISON GROUP II: Scout camp | All |
|---|---|--|---|---|
| N | pretest: 28 posttest: 22 (78 %) | pretest: 21 posttest: 20 (95 %) | pretest: 17 posttest: 9 (53 %) | pretest: 65 posttest: 51 (77 %) |
| AGE M | 8.00 (6—10 Std. Dev 1.2225) | 7.81 (6—10, Std. Dev .981) | 8.06 (7—10, Std. Dev .827) | 7.95 (6— 10, Std. Dev 1.038) |
| SEX (N) | | | | |
| F | 20 (71.4 %) | 12 (60 %) | 8 (47.1 %) | 40 |
| M | 8 | 8 | 9 | 25 |
| SOCIOECONOMIC STATUS MEDIAN (SCALE 1—8, stat.fi) | 3 3 – 6 | 4 3 – 8 | 3 3 – 5 | 4 3 – 8 |
| PARENTS IN TECHNOLOGY | 25 % | 10 % | 35 % | 23 % |
| FAMILIAR WITH HELLO RUBY (%) | 73 % | 5 % (Has played coding games 15 %) | 31 % | 38 % |
| RAVEN’S MATRICES | M: 1.7083 Std. Dev .69025 Matrix 1: 100 % (Missing 5) Matrix 30: 52 % (Missing 5) Matrix 60: 29 % (Missing 11) | M: 1.5500 Std. Dev .82558 Matrix 1: 95 % (Missing 1) Matrix 30: 65 % (Missing 1) Matrix 60: 6 % (Missing 3) | M: 1.4667 Std. Dev .91548 Matrix 1: 93 % (Missing 2) Matrix 30: 40 % (Missing 2) Matrix 60: 36 % (Missing 6) | M: 1.6207 Std. Dev .76840 Matrix 1: 97 % (Missing 8) Matrix 30: 53 % Matrix 60: 22 % |

analyses to be able to better detect the conceptual change of the children in the Summer School intervention group, which ended up difficult because of the homogeneity and acquaintance with the Hello Ruby universe of the intervention group. (Elmes, Kantowitz & Roediger, 2012.) The data from the interviews of four children in the material comparison group is, however, used for qualitative comparison in the study (see Chapter 3).

The parents' permission for both the study's data collection and interviews was requested when signing up for the Hello Ruby Summer School. The sports camp participants' parents signed them up for the study itself, so they naturally agreed to the data collection, and were rewarded for the study with movie tickets and the Hello Ruby –book after the study was finished. The scout camp participants' parents were asked for permission in written form, and each of the child participants of this study have written permission from their parents for the data collection and interviews. The scout camp participants got to keep the Hello Ruby –material they got to browse for the intervention period, and also got movie tickets after finishing the second SRQ and interviews.

In the beginning, it was unclear whether Hello Ruby Summer School participants would represent a reliable sample considering socioeconomic status of parents; the children whose parents work in the field of technology could give more informed answers than those whose parents work in other fields. This issue was controlled by asking the children of their guardians' profession in the SRQ. Socioeconomically the children were quite similarly positioned in the intervention and comparison groups, although the Summer School children's parents were slightly more often in leading positions in companies and society. A little surprisingly, scout participants' parents worked most often in technology, sports campers' the least. The SRQ also included three Raven's matrices (1., 30. and 60.) to control the non-verbal aspects of intelligent function of the participants. Scout camp participants got the hardest one correct most often, and sports campers the least often. Although this was not an entire IQ test, it was the best available solution to control IQ considering the resources of the study.

2.2 MATERIALS

The pre- and posttest conditions of this study were exactly the same, apart from one skills task that ran into a ceiling effect in the pretest condition. There was a self-report questionnaire conducted, examining both the participants' knowledge, beliefs &



Figure 2. The visual format of the Likert-questions in the questionnaire, derived from the official Hello Ruby assets.

understanding of, and attitudes, emotions & motivations (Eccles & Wigfield, 2002; Wigfield & Eccles, 2000) towards computers and programming, that the participants filled with the help of their parents and/or the researcher and assistants present to clarify the questions in cases needed. The attitudes, emotions & motivations and the knowledge, beliefs & understanding parts of the survey are Likert –form matrices of questions examining the children’s concepts and cognitions about programming, executed in such a form, that the children could better understand what they were being asked, with five different smiley faces derived from the official Hello Ruby illustrations (Figure 2). There was also an open-ended question about programming included, and a skills-section, with tasks similar, but not identical to the ones found in the Hello Ruby-book (Liukas, 2015). The skills-section varied slightly between the pre- and post-conditions to prevent recollection of the tasks. The tasks included two sorts of tasks, explained below.

Crack the code

The crack the code task was a simple alphabet-decoding task that was similar to the secret language –task in the Hello Ruby -book (Figure 3), and a similar task was found on the interview as well. The pretest version of the task was very simple, and it ran into a ceiling effect in both the questionnaire and the interviews. The posttest version contained a debugging aspect, a missing alphabet or messed up words, to try to block the ceiling effect and to measure the debugging capabilities and context comprehension of the children: whether they just decoded the letters or tried to use context cues to aid the process. The task measured the concepts of operators and data, and the practices of testing and debugging, reusing and remixing and abstracting and modularising. The tasks were scored based on two

Anteeksi mitä?

Nyt pingviinillä on sinulle jotain sanottavaa. Pystytkö selvittämään viereistä taulukkoa apuna käyttäen, mitä hän haluaa sanoa?

| | | | |
|---|--|---|--|
| A | | P | |
| B | | Q | |
| C | | R | |
| D | | S | |
| E | | T | |
| F | | U | |
| G | | V | |
| H | | W | |
| I | | X | |
| J | | Y | |
| K | | Z | |
| L | | Ä | |
| M | | Ä | |
| N | | Ö | |
| O | | | |

Figure 3. Penguin’s message task from the first questionnaire. The idea of the task is simply to solve the penguin’s message by using the given alphabet of visuals.

levels, the decoding level (“What did the penguin say?”) and the debugging level (“What would the penguin have wanted to say?”), and each level brought one point, the pretest task therefore having maximum score of 1 point, and the posttest task 2 points. The case sensitivity and spacing were not introduced in the assignment, so they were not taken into account.

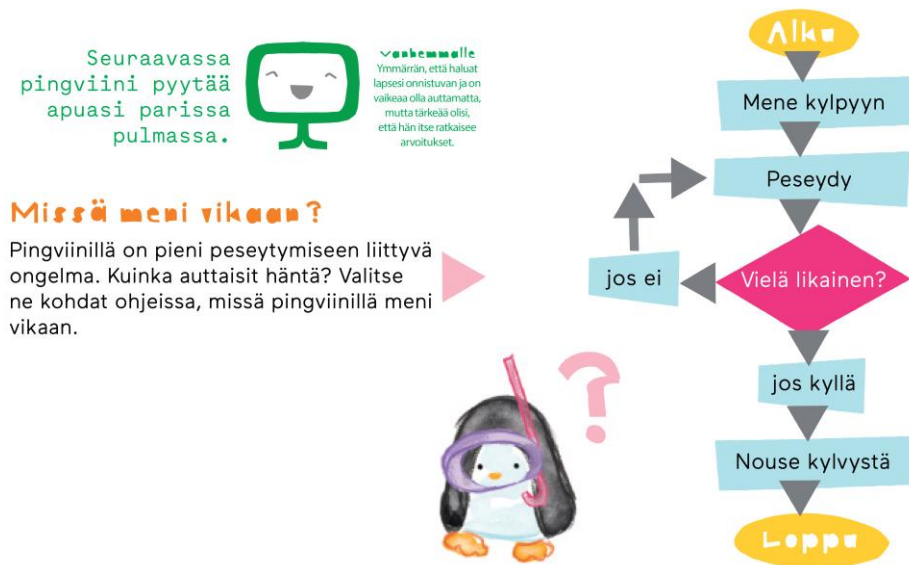


Figure 4. Penguin’s message task, that was in the first questionnaire.

What went wrong

The What went wrong -task (Figure 4) was given on the SRQ exclusively. It was a process chart of the penguin’s mission, in the pretest getting washed and in the posttest safe diving, that contained a debugging aspect of trying to find the part of the chart, that went wrong. In the chart, there were multiple correct items, each of which were worth 1 point, one of the items was worth 0 points (the tile before the diamond shaped selection tile), and the incorrect items brought -1 point.

2.3 PROCEDURES

The descriptive analysis of the questions revealed, that many questions formulated in the manner “computers can” and “robots can” yielded different answers, which were much more skew concerning computers. Yet the statement “Robots are computers” was very much agreed upon (Mean: 3.58, Median: 4, Std. Dev: 1,303). This perhaps reflects a conceptual confusion about the nature of robots and computers, which was even more strongly visible in the posttest with the variable having within-intervention group kurtosis of -1.445. Nevertheless, the questions left out of the analysis were the ones having to do with computers rather than robots to achieve a cohesion of the sum composite variables. The other variables

that were removed due non-normal distribution were: “Tietokoneiden sisällä on ihmisiä”, “Tietokone voi olla kuin ihminen”, “Tietokoneet ajattelevat”.

The sum composite variables were formed based on the 2012 ACM Computing Classification. The SRQ was not initially formed based on this system, but classifying the questions based on these categories seemed to provide the best alphas and there were not enough participants for a factor analysis, so it was decided to use this classification. The reliability of the measures was first looked at in the intervention group posttest situation, then the pre-post change was analysed in this group, and after the scale reliability measures were checked also in this group exclusively. This was done because it was assumed, that the intervention group would have the most accurate conceptions on different areas of computing after the summer school intervention. The sum composite variables formed for the pretest and posttest questions based on the ACM classification were: *Hardware*, *Mathematics of Computing*, *Computing Methodologies*, *Computing Methodologies – Artificial Intelligence Philosophical*, *Computing Methodologies – Artificial Intelligence Epistemic*, *Human-Centred Computing* and *Computer Systems Organizations – Embedded Cyber-Physical Systems*. The contents, reliability values and descriptions of each variable are presented below in Table 4.

Paired samples t-tests were performed between the pre- and posttest sum composite variables and the What went wrong –skills task in the intervention and comparison groups to detect the variables with possibly the most significant differences for further analysis, and independent samples t-tests were performed on all the sum composite variables between groups. After this, a repeated measures MANOVA was performed for each sum composite

Table 4. Sum component variables of technology concepts. The correlations and alphas are based on Hello Ruby Summer School participants. Note that the variables were formed based on T2 values.

| SUM COMPOSITE VARIABLE | COMPONENTS & CORRELATION (T1, T2) | CA T1 | CA T2 |
|---|---|--------------|--------------|
| HARDWARE <i>The hardware aspects of computing.</i> | Tietokone on sama asia kuin internet .090, .747 Tietokone on sama asia kuin kone 488, .625 Tietokone on sama asia kuin sähkö .468, .658 | .516 | .821 |
| MATHEMATICS OF COMPUTING <i>The mathematical and data handling aspects of computing.</i> | Tietokone on matematiikkaa .682, .750 Tietokone on numeroita .767, .731 Tietokone on dataa .042, .358 | .634 | .752 |
| COMPUTING METHODOLOGIES <i>The general computing methodologies aspects related to e.g. machine learning and under the subclass artificial intelligence computer vision and natural language processing.</i> | Tietokoneet tuntevat .632, .669 Tietokoneet aistivat .524, .424 Tietokoneet näkevät .781, .216 Tietokoneet kuulevat .738, .417 | .830 | .640 |
| COMPUTING METHODOLOGIES -- ARTIFICIAL INTELLIGENCE Philosophical <i>The more philosophical aspects of artificial intelligence.</i> | Robotit ajattelevat .114, .350 Robotti voi olla kuin ihminen .486, .400 Tietokoneet ymmärtävät minua .170, .432 | .407 | .582 |
| COMPUTING METHODOLOGIES -- ARTIFICIAL INTELLIGENCE Epistemic <i>The epistemic aspects of artificial intelligence.</i> | Tietokoneilla on tietoa minusta .299, .848 Tietokoneet tietävät minusta .299, .848 | .455 | .915 |
| HUMAN-CENTRED COMPUTING <i>The questions on autonomy of computing systems.</i> | Ihmiset ohjaavat aina tietokoneita .398, .665 Ihmiset ohjaavat robotteja .398, .665 | .555 | .780 |
| COMPUTER SYSTEMS ORGANIZATION - Embedded and cyber-physical systems <i>The questions on utilization of embedded robotics and sensor networks in different industrial and artistic fields.</i> | Tietokoneet voivat luoda musiikkia .518, .678 Tietokoneet voivat luoda taidetta .458, .816 Tietokoneet voivat suunnitella koteja .623, .671 Robotit voivat kierrättää .394, .481 | .707 | .825 |

variable nearing statistical difference in the repeated measures t-test for Hello Ruby participants. Since a significantly large portion of the Summer School participants were already fairly familiar with Hello Ruby technology education material (a factor discussed as a methodological limitation below), this could, of course, affect their pretest answers. The effect of the pretest value on the posttest was controlled using it as a covariate in an ANCOVA on the variables approaching statistical significance in the MANOVA or statistically significant on the t-test.

2.4 RESULTS

In single question analysis it was, first of all, interesting to note, that in the question “I know what programming is”, the intervention group children had become more hesitant in the posttest situation (Figure 5). The t-tests revealed, that in the pretest there was a statistically significant difference between the intervention and the comparison group only in the epistemic artificial intelligence variable ($t(60) = 3.403, .001$), but the posttest showed no statistically significant difference in this variable. In the posttest statistically significant differences could be found in the variables Embedded Cyber-Physical Systems ($t(46.810) = 2.287, .027$) and Artificial Intelligence Philosophical .050 ($t(50.261) = 2.008, .050$). The individual variable belonging to the latter sum composite variable showing most significant difference in the t-tests was “Tietokoneet ymmärtävät minua”, “Computers can understand

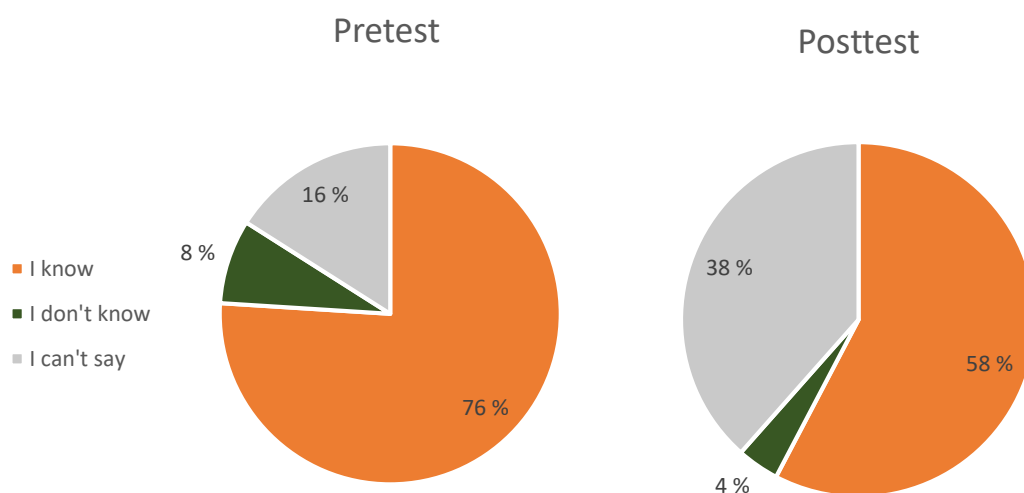


Figure 5. The reclassified distribution of answers to the question “I know what programming is” in the intervention group.

me” ($t(51) = 3.305, .002$), and the change of the intervention group’s variable mean was from 2.88 to 3.52.

Concerning the skills-tasks, it was already mentioned, that the decoding task ran into a ceiling effect. The What went wrong -task, however, was analysed, and showed no statistically significant difference between the groups in the pre- or posttest, but an independent samples t-test showed a statistically significant difference between the comparison group’s pre- to posttest performance ($t(19) = 2.517, .021$). When counting in the scout comparison group, it also seemed like the intervention group was the only group whose performance had actually declined.

In repeated measures ANOVA, statistically significant differences were not found in any of the variables. The ones approaching statistical significance were Embedded Cyber-Physical Systems (Wilk’s Lambda = .904, $p = .078$) and Technology Attitudes (Wilk’s Lambda = .926, $p = .075$). However, the results from the ANCOVA with the pretest condition as covariate

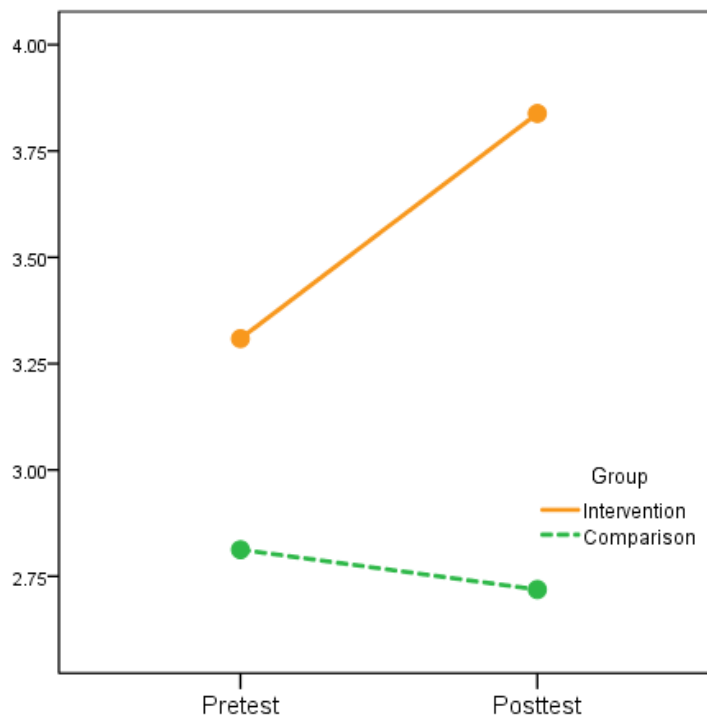


Figure 6. The statistically significant change of sum composite variable Embedded Cyber-Physical Systems. Shows that the intervention group’s conceptions about the utilization of embedded robotics and sensor networks in different industrial and artistic fields changed in a whole different direction in contrast to the comparison group.

show a significant difference after controlling the pretest performance on the variable Embedded Cyber-Physical Systems ($F(1, 30) = 7.605, p = 0.010$, see Figure 6). Technology attitudes, being high in the beginning and the end in both groups (intervention M: 4.35 & 4.59, comparison M 4.51 & 4.50), the variable didn't show a statistically significant change in this investigation ($F(1, 49) = 2.884, 0.096$).

2.5 DISCUSSION OF THE RESULTS

In conclusion, it can be stated based on the quantitative data that the summer school curriculum themes played a large part in changing the computational thinking beliefs of the children. The children's view on the subject of embedded systems with physical attributes and their creative possibilities seems to have changed during the summer school based on this analysis. Also with some reservations, it can be concluded from this study that the summer school affected the participants' philosophical thoughts on artificial intelligence, and that they now more often feel, that computers understand them, and that robots can have human features. The comparison group's awareness of computers' possessing information about them seems to have increased after the intervention period. This can be speculated to have happened as some kind of a "side effect" of the research, and the difference in the pretest can be concluded to have to do with the intervention group children having been more familiar with Hello Ruby prior to the pretest and their parents being more often in technology related occupations.

3 QUALITATIVE STUDY

3.1 PARTICIPANTS

The qualitative data collected were videotaped interviews conducted before and after the intervention period, in the intervention group (n = 6) and each of the comparison groups (n = 4 + 4). The participants for the interviews were randomized in the intervention group and the scout comparison group, and selected in order of signing up in the sports camp group. The groups were, however, age-coherent, consisting of only 8-year-old children, apart from one 10-year-old in the scout comparison group, to ensure the comparability of the answers (more on the reliability and validity of the participant selection in the discussion).

3.2 MATERIALS

The interview included a concept explanation section, with concepts drawn from the SRQ themes and questionnaires that were previously conducted at a lower level comprehensive school by Hello Ruby (see Appendice 2), and a think-aloud skills section with tasks similar but not identical to the ones found in Hello Ruby –book, scored and analysed qualitatively (Liukas, 2015). The other one of the tasks, task number 2., the Crack the code –task was explained in the quantitative part of the report, and the primary think-aloud task, the Computational thinking mine, is explained below.

Computational thinking mine

The computational thinking mine task (Figure 7) was similar to an assignment in the Hello Ruby -book called “Plant and weed” which resembles visual programming languages, like Scratch. It included 5 “mines” to be “mined” by the miner creature and connected to the appropriate instructions to solve the particular mine, and in some cases, there were two instructions to solve one mine. The computational thinking mine task measures all concepts

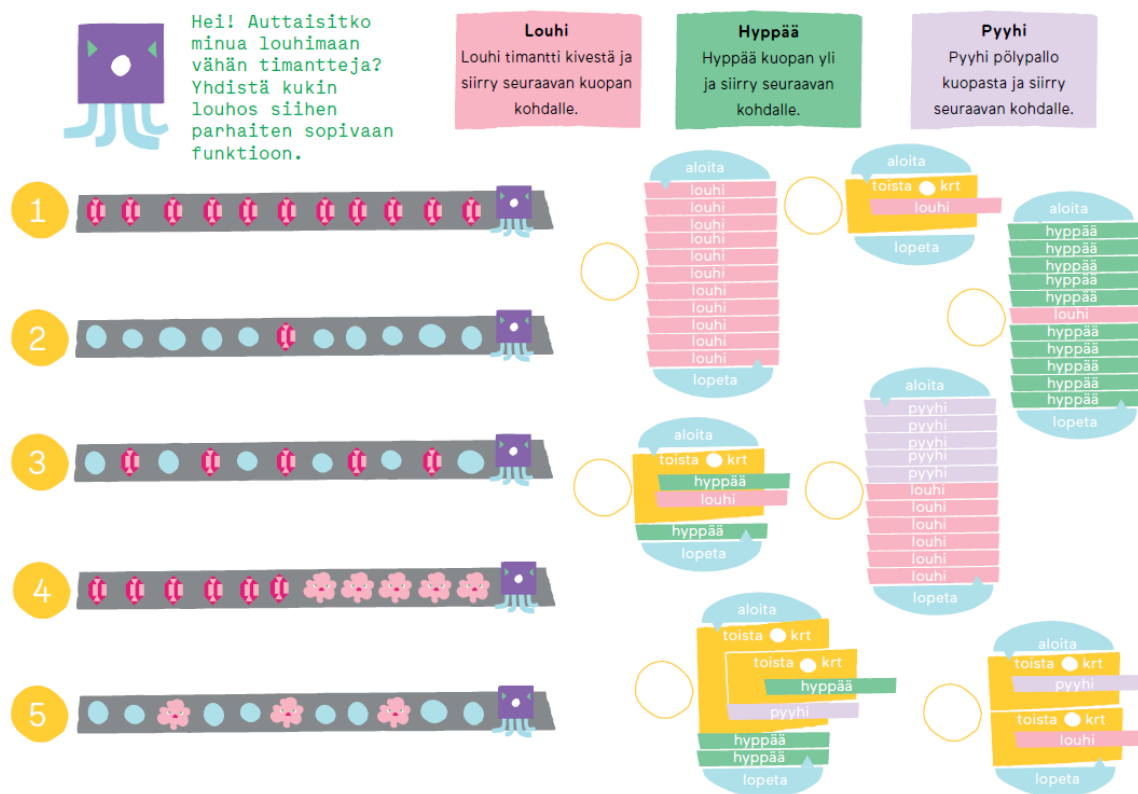


Figure 7. Computational thinking mine -task from the first interview. Each of the “mines” on the left (1, 2, 3, 4 & 5) corresponded to one or two of the “mining instructions” on the right. The mission is to choose the correct instruction for the “miner creature” to solve the correct mine. Each of the modules in the instructions (“louhi” = mine, “hyppää” = jump & “pyyhi” = wipe) on the instructions were explained to the children and the written instructions were given above. Each mine contained one “sticker” for the child to put on the correct instruction, and they were also given a pen to fill in the blank fields with the correct number of repetitions (“toista [X] krt”).

related to computational thinking mentioned above, and in addition the practices of experimenting and iterating.

3.3 PROCEDURES

The qualitative data was analysed with the method of qualitative content analysis (Krippendorff, 2004; Kvale, 2007), and the answers’ breadth and conceptual content was analysed based on the conceptual change theories specified above. The concept explanations

were also graded on a 1—4 Likert-scale by the researcher on different dimensions listed below, and t-tests were performed on this data.

1. Were the explanations related to Hello Ruby –material?
2. Had the explanation of the concept of program changed to computer (rather than TV) related?
3. If the child could explain the concept “programming”, could they also explain “algorithm” and “data”?
4. Could the child explain more concepts than in the pretest?
5. Were the explanations more detailed and accurate?
6. Were the concepts clearly easier to explain?
7. Had the child’s primary interests changed during the study?

With the think aloud –tasks of the computational thinking patterns, misunderstandings and incoherencies of the children were analysed qualitatively.

3.4 RESULTS

In the interviews, most Ruby’s Summer School participants had more extensive and elaborative correct answers than the other groups and that they had at the pretest, some also giving implication of conceptual change in addition to conceptual accretion (results presented on Table 5). The comparison group participants didn’t really seem to understand why they should answer the same questions twice, but some of their answers, especially one scouts camp participant’s (PP3) who was really knowledgeable beforehand, were as correct as in the pretest, but simply more compact. The sports camp participant who had the most knowledgeable pretest answers (LP3) even presented some level of computational diminishment in the posttest with the concept of program (see Table 5). The family environment and parents being in technology also affected the children’s’ answers to some extent, in both the intervention and comparison groups. Both of the more knowledgeable comparison group participants (LP3 and PP3) had parents in technology. One summer school participant (HP6) mentioned he first thought he was coming to his mother’s workplace when they said he was going to the Hello Ruby coding summer school.

HP2 T1: *”Mut mä oon joskus nähny ku isi... Meiän isillä on semmonen joku tieto-
-... aika iso tietokone, niin mä oon nähny kyl ku se on tosi paljon sillä tehny ja
tuijotellu sitä.-- Pienenä joskus autoin sitä semmoses, ku öö... se suunnitteli jotain*

unileluu jotain, siit on aika kauan, jossain pelis oli joku mukana ollu, ni sitte, öö mä autoin sitä ja tein sille yhen osan, niinku päähän törröttävän.

But I've sometimes seen when daddy... Our daddy has some kind of a comp--... pretty big computer, so I've seen when he's done lots of stuff on it, and stared at it. --When I was small, sometimes I helped him with, umm... He was designing some toy or something, it's been a long time, it was for some game, and then, uh, I helped him, and I made him a part of it, like a part sticking from the head."

The concept of program was in general more often correct with the Ruby's Summer School participants, and they could also more often explain the additional concepts of data and algorithm. Considering how little time the children spent on a computer during the summer school (the smaller children spent only approximately three days on a computer during the entire camp), this was quite an accomplishment. The little amount of time spent on a computer compared to the large amount of time spent on games and artistic activities was also reflected in the nature of the concept explanations, for example this explanation of an algorithm: *"--eiks se ollu sitä niinku jos sä sanot vaik et 'piirrä ympyrä monta kertaa', niin sit se niinku... pieni ympyrä, niin sit sä piirrät näin, niin pitkään, kun sä saat hyvän lopputuloksen. // Isn't it like if you say 'draw a circle many times', so it's like ... a small circle, and then you draw like this, for a long time, as long as you get a good outcome"*.

Concerning motivation and attitudes, the summer school participants clearly outplayed the comparison group participants. One participant (HP6) drew a parallel between games and coding in the posttest, and when asked about whether he meant that coding is fun, he said *"Yes, more fun."* Only one comparison group participant (LP1) brought up new kind of motivation to start learning coding to know how to make games. The summer school participants also seemed to feel quite empowered by all the things they had created during the summer school.

Table 5. Analysis of the conceptual part of the interview, with change indicating quotes from pre- and posttests.

| | Hello Ruby Summer School | Comparison group |
|---|---|---|
| <p>Computational thinking:</p> <p>Computational expansion / preservation vs. computational diminishment or compression</p> | <p>HP2T1: "[<i>Ohjelmointi</i>] –No ne jotenkin niinku... laitetaan sille käskyt, esim. Jos hissin haluu, et se menee eka siihen mikä ensimmäisenä painetaan, esim. Jos painetaan ensimmäisenä nelonen, sitten painetaan kolmonen, se menee eka sinne neloseen, sit menee vielä pikkusen alemmas kolmoseen."</p> <p>HP2T2: "Se on yleensä sitä, millä <i>ohjelmoidaan</i>, esim. robotin voi ohjelmoida siivoamaan ton lattian vaik tossa, tai vaik maalaamaan, ja sit... tietokoneit ohjelmoidaan, telkkareit ohjelmoidaan, kaikenlaisii muitaki ohjelmoidaan. <i>Algoritmi</i>, eiks se ollu sitä niinku jos sä sanot vaik et piirrä ympyrä monta kertaa, niin sit se niinku... pieni ympyrä, niin sit sä piirrät näin, niin pitkään, kun sä saat hyvän lopputuloksen."</p> | <p>LP3T1: "[<i>Ohjelma</i>] No se on niinku... voi olla niinku... se on juttu joka on niinku... öö, tota koodattu tai ohjelmoitu, ja sitten niinku sen voi toistaa monta kertaa."</p> <p>PP3T1: "No <i>sähkö</i> on periaattees sellanen niinkun... sitä syntyy niinkun tietynlaisista fyysisistä tapauksista ja sellasta, ja sillä voidaan tehdä virtaa, niinkun... no saada esimerkiksi lamppuja päälle."</p> <p>LP3T2: "No niinku semmonen sovellus, tai sitte semmonen <i>ohjelma</i> jota voi niinku kattoa."</p> <p>PP3T2: "no minusta <i>sähkö</i> on periaatteessa vaan sellasta virtaa.—no laitteiden pyörittämiseen."</p> <p>→ Computational diminishment with sports camp participants</p> <p>→ Computational compression in scouts and sports camp participants</p> |
| <p>Conceptual change</p> | <p>HP1T1: "<i>Tietokone</i> on sellanen, että se on niinku kännykkä mut vaan isompi" "[<i>Ohjelmointi</i>] En tiedä."</p> <p>HP1T2: "<i>Tietokone</i> on vähän niinku ihminen, mut se on vaan robootti" "se on ohjelmoitu ite, ja tietokoneessahan ohjelmoidaan, mutta se on myös ohjelmoitu ja silleen niinku päästä" "<i>Ohjelmointi</i> on sellasta, et se on niinku tietokoneen käyttöä, joka... joka on niinkun sellasta tietokoneen harjoittamista. – no silleen, että siinä sä voit tehdä sen ite, tai sit se on jo valmiiks tehty."</p> | <p>LP1T1: "No <i>teknologia</i> o sellane niinku tulevaisuuden sellane... sellane mis on kaikkii sähköä uutta, ja sellasii... ku mä kuulen sanan teknologia, nii mun päähän tulee sellasii lentävii autoi ja leijulauttoi ja sitte kaikkee sellast, nii sen saa mus mieleen teknologia."</p> <p>LP1T2: "Vähän samanlaist ku viimeks. --No <i>teknologia</i> on kyl sellane mis o kaikkii sellasii sähkölaitteit ja sellasii."</p> <p>→ No signs of conceptual change.</p> |
| <p>Conceptual accretion</p> | <p>HP3T1: "<i>Tietokone</i> on elektroninen laite ja se antaa tietoja."</p> <p>HP3T2: "<i>Tietokone</i> on semmonen mis on näppäimiä, ja sen sisällä on ram, rom, prosessori, näyttönohjain, öö, ja sit joku... mä opin ne täällä. No siit löytää tietoa, esim. Wikipediasta, sit sil voi, öö... sil voi, sil voi... kirjottaa nettiin asioita, sil on hyvä muisti. Sit sil voi pelaa pelejä, ja semmosta."</p> | <p>PP4T1: "Olisko se jotain että voi kattoo... laittaa sähköpostia?"</p> <p>PP4T2: "En tiä. Kun me sanotaan aina kotona tietokoneeks sitä läppäriä." "mitä varten se on olemassa?" "että voi suunnitella taloja."</p> <p>→ No signs of conceptual accretion.</p> |
| <p>Motivational change</p> | <p>HP6T2: "<i>Peli</i> on sellanen vaikka niinkun... niinku se koodaaminen on vähän sama asia kuin peli, mutta siin ei oo mitään taistelua. --Ai voiko pelin tehdä? Voi." M: "—tarkottaaks se et se on sun mielest hauskaa? Et sä tykkäät koodaamisesta?" "On. Joo." M: "Tuntuuks sust et se on yhtä hauskaa ku pelaaminen? Vai hausempaa?" "Hauskempaa." HP1T2: "Sen verran tullu, että, tietokoneella voi tehdä ihan mitä vaan ja jaata sen nettiin--"</p> | <p>LP1T1: "No oommä suunnitellu, et mä haluaisin tehdä jotai pelei, mut ku mä oon kattonu, että ne o vitsin monimutkasii, nii ehkä mä viel harkitsen sitä, ku... -- On sellanen [peli] mis se koko näyttö on sellanen juttu ja siel liikutaan, ja sit siel on vitsin paljon noit yksityiskohtia... Siel on vitsin paljon noit yksityiskohtia, kuten vaikka... jos vaikka... Mä en osaa oikeen selittää, mut se näyttää niinku ois oikeesti jossai tietys paikas, ku siel on vitsin pal yksityiskohtii..."</p> <p>LP1T2: "[<i>Ohjelmointi</i>] no sil voi tehdä kaikkii pelei, ja sitä mä oo aatellu äitin kaa just alkaa harjottelee."</p> |

HP1: *"Sen verran tullu, että, tietokoneella voi tehdä ihan mitä vaan ja jaata sen nettiin, kuten... kuten... ku me tehtiin musiikkeja ite, niin sitten Olli lupas et se menee tonne nettiin, ja sitte se sano että siitä voi tehdä soittoaänen"*

"So much I've learned that with the computer you can do whatever you want and share it online... like ... when we made the music ourselves, then Olli promised we can put it online and then make it into a ringtone."

When asked to reflect on what they had learned during the summer school, some girl participants of the summer school brought up quite impressive metacognitive analysis. It seems with at least one of them they had noticed an improvement even in their skills playing different leisure time coding games (HP3), and one just said they didn't really know what computers were before and now know better what is inside them (HP2). So the confusion about the nature of computing that showed up in the questionnaire was visible in the interviews as well. One girl (HP2) also mentioned when she was younger she even thought computers weren't made by humans.

HP3: *"Mä oon periaattees samanlainen ku ennen, mut mä vaan tiään mitä tietokoneen sisäl on. Ja mä osaan koodata. Ainaki vähän. Ennen mä en tienny mitään ohjelmoinnista, tai ainakaan paljoo. Nyt mä tiään enemmän. Ja nyt mä tiään mikä on systeemi." "Tuntuuks sust, et se liittyy tähän koodaamiseen jotenkin?" "No kyl se vähän, koska systeemi on koodattu."// "I'm basically same as before, but I just know what there is inside a computer. And I know how to code. At least a little bit. Before I didn't know anything about programming, or at least not much. Now I know more. And now I know what is a system." "Do you think it is related to coding somehow?" "Well, yes, it does a little bit, because a system is coded."*

HP2: *"Emmä käytä hirveesti tietokoneit, mut mä tiedän ainakin enemmän mitä siel on sisällä... Mä joskus pienenä luulin, et se joku... et se... et se ei oo mitenkään ihmisen tekemä..." // "I don't use computers much, but at least I know more about what is inside them... When I was very little I thought it was some kind of... that it ... that it's not man made or anything..."*

Concerning the variables indicating most significant change in the SRQ, the differences could clearly be seen in the interviews as well with the concepts of embedded cyber-physical systems and philosophical artificial intelligence (Table 6). There were many parallels drawn

between robots and humans, even between computers and humans, in three of the intervention group interviewees' posttest situation, as in the comparison group there was one incident with a scouts camp participant, who was already a Hello Ruby fan in the pretest. The summer school participants also felt they could be used for anything, such as cooking, dishing and housework in general, and they brought up on average more applications than the comparison group participants.

The between-group t-test results show, that the Hello Ruby Summer School participant's answers were more based on Hello Ruby -material and that they could explain more concepts more precisely and correct than the other groups. They also more often mentioned changed primary interests or hobbies than the other groups, and for example, when the sports camp participants, who had tried different forms of sports during their camp, more often brought up a new sports hobby, the summer school participants' new hobbies were more often related to games, coding, arts and even music.

Think-aloud material

The think-aloud data revealed some differences between the participants' thinking patterns, and indeed, some incoherencies were discovered as well. The data indicated similarities and differences on two levels: between all children and between groups. First of all, in the computational thinking mine assignment all of the children consistently displayed inability to calculate the last function's or instruction's numbers correct in both pretest and posttest situations, even though they could select the correct function for the right mine. Other children made the decision to sum up the entire mine's worth of wipe and jump steps ($6 + 3$), some just for one iteration ($2 + 1$). This task with two levels of loops was later realized to contain a conceptual challenge of multiplication tables, that the children had not yet been taught at school, and that a short two-week summer school was incapable of overcoming, similar to what Kurland and Pea found already in their 1985 study. The problem was, therefore, most probably with the selection of the 8-year-old participants, although the one 10-year-old scouts camp participant was also incapable of solving the task. This problem will be analysed in more detail in the methodological limitations section of the discussion below.

Table 6. Analysis of the statistically changed sum composite variables in the posttest interviews highlighting different nature of responses between groups.

| | Hello Ruby Summer School | Comparison group |
|--|---|---|
| Embedded Cyber-Physical Systems | <p>HP5 T2: <i>"No se on niinku ihmisen tyyppinen tai se on ohjelmoitu samanlaiseks ku ihminen, se voi auttaa esimerkiks äitii tiskaamisessa ja tehdä erilaisii kotitöitä."</i></p> <p>HP1 T2: <i>"Tietokone on vähän niinku ihminen, mut se on vaan robootti" "se on ohjelmoitu ite, ja tietokoneessahan ohjelmoidaan, mutta se on myös ohjelmoitu ja silleen niinku päästä."</i></p> <p>HP5 T2: <i>"Robotti on vähän samanlainen kun ihminen, mutta viruksilla ja... mitkä ne olikaan... sillä... niin robotti on sama asia kun tietokone, mutta ilman näyttöä."</i></p> | <p>PP1 T2: <i>"Se on semmonen, että ihmiset voi ohjelmoida sen vaikka kokkaamaan, tai jotain semmosta. --[Ohjelmointi] niin silleen että sen laittaa jollain tietokonejutuilla niinku toimimaan." (No change)</i></p> <p>PP3 T2: <i>"Tavallaan kone, joka niinku liikkuu. – No yleisesti ottaen niil on paremmat mahdollisuudet kun ihmisillä, eli niil on esimerkiks paremmat refleksit ja sellaset. –No autojen rakentamiseen tai muiden sellaisten koneiden rakentamiseen, koska siinä ne on vähän nopeempia." (No change)</i></p> |
| Artificial Intelligence - Philosophical | <p>HP1 T2: <i>"Kone voi olla mikä vaan, kuten sydän on kone. Sydän on iso kone, ja se auttaa ihmisiä ja se tekee niitä, ja sitten se... se voi auttaa" "Robotti on sellanen, niitten ne sähkö... ja sitte jos sä vaikka äänität jotain, niin sitten se vastaa"</i></p> <p>HP6 T2: <i>" [Robotit] No ne on olemassa siksi, että ne voi joko suojella ihmisiä tai pelastaa ne, tai olla ihan vaan kotihoitajina."</i></p> | <p>PP2 T2: <i>"Robotti on vähän ku ihminen." M: "Miten se eroaa? Vai erooks se?" "Riippuu siitä minkälainen robotti."</i></p> <p>LP1 T2: <i>"Robotti on sellanen vehje mitä pystyy käskyttämään jos se on ohjelmoitu oikein." (No change)</i></p> <p>PP4 T2: <i>"Ne on jotain leikkikaluja millasilla jotkut joskus leikkii." (No change)</i></p> <p>LP1 T2: <i>"Se o sellane ihmisen näköne joka liikkuu sellattis oudosti." (No change)</i></p> |

Another notable difference was, that only a few of the children could independently realized, that for some tasks there were several functions that could be applied, and that some were just more condense and efficient than others, namely the repeat functions. Concerning the mines, the third notable between-participant difference was, that only a few people could get the iterations correct in the mine number 3, and they represented all of the participatory groups, both in pretest (3) and in posttest (5) situation. Since the participants who had improved their performance with this mine belonged to the intervention and the material groups (scouts). This could indicate the effect of Hello Ruby material, since the one sports camp participant that got the 3rd mine correct both in pretest and posttest (LP3, who also mentioned having played coding games prior to the pretest), was also notably hesitant in the posttest.

LP3 T1: *Eli... Eli tota... Tää pitää tota toistaa, öö... Yks kaks kolme neljä viis kertaa. Tää. // Well... Well... well keep umm... repeat, uh ... One, two, three four five times. This.*

M: *Sä voit kirjottaa sinne sen... oliko tohon se viis? Tää pyyhkiytyy sit pois.* "You can write it there... was it there, five? It rubs off.

LP3 T1: *Viis. Jaaa... // Five. Ohhh...*

LP3 T2: *Hmm... Mut se on niinku toi, mut sit siin ei niinku hyppää kertaakaan... Noku ei täs oo muutakaan... Ei se voi... Ei se voi olla toi, koska ekaks tohon tulee toi louhi, ja sit vast pyyhi. Se on toi. // Hmm ... But it's like that, but then it's not like jumping at all... Well cuz there's not anything else... It cannot ... It cannot be that, because first there is that "mine", and after that "wipe". It's that one.*

M: *Osaaksä sanoo mikä luku siihen tulis? // Can you say what number it is?*

LP3 T2: *No tähän pyyhi tulee... Yy kaa koo nee vii... Siihen tulee kuutonen. Tohon louhi... Siihen louhimiseen tulee niinku vitonen. Ei siin oo niinku toista paikkaa sille... Eiku siis joo, koska lopussa se pyyhkii. Eli tohon tulee niinku vitonen. // Well, this wipe would be... one two three four five... It will be six. That mine will be... It will be like a five. Hey but there's not another place for it... No but yes, because in the end it wipes. So, it will be five.*

Lastly, in the penguin's message decoding assignment there were in total 4 different kinds of strategies for message decoding that were visible. The most common one, used by a vast majority of the participants in both pre- and posttest situations, was writing the message down checking each letter from the alphabet, one by one, and then reading the message out loud (and perhaps commenting it somehow). In the pretest situation, there were two Summer School participants (HP4 and HP5), who were incapable of thinking aloud during the assignment, since they didn't choose or, apparently, didn't need to write anything down, but just decode the message in their heads and pronounce it out loud. As the posttest assignment contained a debugging aspect, this strategy was computationally too heavy to solve all the messages, and these participants, in this case, also had to write them down to solve them. There was also one sports camp participant (LP2), who was too insecure or reluctant to solve any of the assignments in the pretest, but in the posttest was able to decode and pronounce the second message with this same strategy, without writing anything down.

The third decoding strategy was spontaneously displayed by only one sports camp participant (LP4), in both the pretest and the posttest situations, and it provided her with a computationally notably efficient “decoding detour”, so to speak. This participant used the strategy of “cheating” the already solved alphabets from the previous words and messages, therefore expanding the idea that some children used with two successive similar icons or letters. The participant spent a lot of time explaining about the different games she had been playing, so perhaps there was some very successful, subconscious transfer between the computational skills learned in those games and this decoding assignment. This might also be the case with the participants, who did not choose to write anything down, since all of them also mentioned playing some mobile and computer games.

The last strategy, which was the only unsuccessful one, was displayed by one scouts camp participant (PP4). She started solving the assignment by circling the letters from the alphabet to be able to “remember them”, but later realized this was not very successful in aiding memorization.

PP4: Tää kohta on ainaki A. Sitte tää, mä katon täältä mikä se on, sit se on helpompaa ja nopeempaa. Tätä ei ikinä tiedä mikä se o, paitsi jos sen on ratkonu. Täs on et... Tähän tarvitaan aika paljon A:ta. // This is A, at least. Then I look here what it is, it's easier and faster. You never know what it is unless it's solved. This is like... This requires quite a lot of A's

The first one of the between-group differences was mentioned before with the sports camp participant LP3, who was much more hesitant with his posttest answers. Taking into account the fact, that all other children were able to achieve at least the same level performance in the posttest as in the pretest, it could be, that either the lack of playing coding games and the increase of varying forms of physical exercise during the sports camp could have led to this development. Another interesting between-group difference was the nature of the debugging mistakes the different groups made during the posttest in the penguin's message debugging tasks. The summer school participants made very different and interpretive mistakes than the other groups, for example:

HP1 T2: Se ois halunnu sanoo, "Mei... Meäkö... kö... yhtä" "Pidetään yhtä". ("Let's stick together.")

HP4 T2: *"Mennäänkö". "Mennäänkö yyy"... "yhdessä"? ("Shall we go together?")*

LP4 T2: *"Mavuka"... En kyl tiää mitä se vois olla... "Naa"... Sitte toi on O... "Noa"... "No"... "Nodahda"... Ihan päläpäläkieltä. "Tas"... "Taas" on oikee, niinku. // "Mavuka"... I dunno what it could be ..."Naa"... Then that one is O... "Noa"... "No"... "Nodahda"... This is gibberish. "Taas"... "Again" is correct, like.*

J: *Se on varmaan ihan oikee sana. Oisko siel joku virhe? Mitä pingviini on yrittäny sanoa? // It's probably a real word. Could there be a mistake? What was the penguin trying to say?*

LP4 T2: *No tää vois olla "mennäänkö". Sitte tää ois... Tää vois ehkä vaihtaa paikkaa... Sit se "taas" "uimaa"... Se voi olla mahollista, mut jos se... Mun mielest tää p oli outo... // Well, this could be "mennäänkö" ("shall we go"). Then this would be... This could change places... Then it would be "again" "swim"... It could be possible, but if it... I think this p was strange...*

The different nature of the debugging mistakes between the different groups could indicate a mere misunderstanding of the question “What would the penguin have wanted to say?”, or they could have been due to the intervention group’s change in computational thinking perspectives, attitudes and collaborative values, since many of the mistakes imply a spirit of solidarity and community. They could also have been made due to the group’s different level acquaintance with the interviewer. Although not being present the entire time at the summer school, I had to drop by during some days, and although not interacting with or instructing the participants in any way, small children perceive relationships with adults in a different way, and for example, I heard myself being called “teacher” a few times despite my role as a researcher and lacking class teacher education.

4 DISCUSSION

4.1 SUMMARY AND TRIANGULATION OF THE MAIN RESULTS

In conclusion, based on this study it can be stated, that there were some notable conceptual changes that took place during the Hello Ruby Summer School 2016 in the participants' ideas of technology and their computational thinking. The conceptual change manifested on the level of attitudes, emotions and motivations and knowledge and understanding.

The most notable discovery concerning the participants' technology attitudes, emotions and motivations was, that based on the self-report questionnaire they are extremely high in all the participant groups, regardless of acquaintance with Hello Ruby. The interviews, however, demonstrated the intervention group's much stronger spirit of collaboration, expressing, connecting and questioning, the perspective dimension of computational thinking.

Concerning knowledge, beliefs and understanding of the children, it was an interesting discovery, that the intervention group were much more hesitant with the question "I know what programming is" after the summer school. This conceptual confusion about the nature of programming could have been brought about by the much more creative context for programming education than what the children might have been used to, which makes it only a good thing concerning the Hello Ruby company objectives. It might implicate an increased metacognitive understanding through a cognitive conflict (Anto, Penttinen & Mikkilä-Erdmann 2010), since the children might have become aware that there is actually a lot more to programming that they currently do not know. The confusion can also be only situational, since the posttest was conducted more or less in the middle of its appearance, and it might dissolve with time and, in fact, mature into more creative and gifted coder minds.

The actual concepts demonstrating notable (conceptual) change in this study had to do with embedded cyber-physical systems and artificial intelligence. After the summer school, the intervention group had somewhat more optimistic views about the potential applications of computing and robotics in different industrial and artistic fields, which were thematically present in the curriculum. Whether this can be seen as conceptual change or knowledge accretion (Rumelhart & Norman 1978) is, however, controversial, since it can only imply that the themes stuck to their minds, which changed their answers in the posttest SRQ. In the interviews, however, the children were also able to bring up much more applications for

robots, one even speaking about the human heart as a “machine”, which is in fact a concept very much in line with today’s medical technology developments with 3D printed organs slowly gaining ground (Mironov, V., Boland, T., Trusk, T., Forgacs, G., & Markwald, R., 2003). Based on the study, it also seems like the intervention group became at least some level believers in strong AI (Jackson, 1986; Nagel 1974) in consequence of the summer school. Perhaps the Google deep learning based speech synthesis algorithms and all the talk about robotics were powerful enough to make them feel computers can actually understand them, which lead to a theoretical shift through paradigms or ontological categories (Chi 1992; Vosniadou 1994; Vosniadou 2013; DiSessa 1998), which is not uncommon for children at this age, who are open to adapt to dramatically different kinds of thoughts and situations.

The computational skills didn’t seem to have changed significantly as an effect of participation in the Hello Ruby Summer School based on the comparison group analysis of this study. As a result of think-aloud –task analysis it would seem, that the computational skills go very tightly hand in hand with general mathematical understanding, which is gained in primary schooling in the extensive and organized settings of the official system. In the recent Finnish national curriculum reform, this issue is recognized, since computational thinking skills are included in the subject studies of mathematics (The Finnish Board of Education, 2014), which, based on this study as well, seems a well-grounded choice. Games, however, seem to provide transfer and speed up the performance of at least the more low-level decoding kind of tasks, and the children who had played games during leisure time showed more efficient instinctive strategies in solving these kinds of tasks, regardless of the summer activity they had participated in.

Concerning the other participatory contexts explored in the study, for some reason it would seem that the sports camp participants’ feelings of computers knowing and holding information about them had increased significantly as an effect of the study. Only one participant in this group brought up a new interest in learning to code to make games in the interview. The participants’ slightly increased hesitation in the think-aloud –tasks could imply a slight decrease in their cognitive performance as an effect of the sports camp participated, or just simply confusion about the pre-to-posttest design of the study.

4.2 METHODOLOGICAL REFLECTIONS

Regardless of the mixed methods approach there were multiple factors affecting the reliability and validity of the study at hand. First of all, the participants being children of course brings about some problems related to the different methods used in the study. The research form with the scale 1—5 emoji-Likerts was piloted beforehand with primary school- and 5-year-old kindergarten children, and it was deemed a valid measure, as children younger than the study participants understood its' functionality. Also, although there were some cases who did not seem to make it to the end with concentration, especially in the scout participant pretest situation, where there were multiple distractions, most of the children seemed to find the form quite enjoyable to fill, with all its' nice, Hello Ruby –related pictures and colours. The posttest situation, however, seemed to cause confusion with such a short time period between the test situations; The children didn't quite seem to get the idea of a scientifically valid pre-to-posttest. The presence of parents in the form fill-out situation could also have affected the participants' answers, as the Hello Ruby participants and the sports camp participants got to fill out the pre-SRQ at home, and the scout participants' pretest situation was at a room filled with restless fellow campers, with only three adults present to give them help and advice. This is why a part of them missed the first skills-task altogether, as many had left the task untouched. The form had instructions for the parents on not to intervene in the fill-out process, but there was no one present to monitor whether or not they were followed.

Concerning the interview, the interviewer's inexperience with children brought about some minor challenges, and of course, the interview situation can be quite confusing for children happening twice in the exact same format, same as the SRQ. Some children, even the summer school participants who had been immersed in the interview themes, seemed to repeat the answers they gave the first time without further consideration. However, they were still able to improve performance and explain multiple new concepts in the posttest, when the comparison group participants were in a much deeper state of confusion. Perhaps if the posttest would have been conducted after a longer period of time, the children would have had more time to let all the new information and experience soak in and they could have been able to produce more extensive answers with the conceptual change even more visible. A follow-up study on the summer school participants would also be able to answer to

questions on the lasting nature of the realizations and new knowledge constructions formed during summer school.

Second, there were some problems concerning the comparison groups. The sport camps participants' parents recruited their children, so they could have been select in regards of the parents' personal interests and occupations. They could also have spoken with their parents about the subjects prior to and sparked by the pretest, which was visible in, for example one participant mentioning starting to learn programming with their mother to be able to make games. However, there was no notable effect in the posttest interview, only the pretest answers might have been affected by parents' occupational input in two of the comparison group participants, and their answers were also diminished in word length in the posttest, so there were clear differences between the two situations.

Last, there were problems concerning the skills tasks, both in the questionnaire and the think-aloud section of the interview. The SRQ tasks were left out of the analysis due to a strong ceiling effect, the pretest tasks were too easy to show any difference between groups. The think-aloud tasks were somewhat problematic because of the participant selection process; The participants within the Hello Ruby group were selected based on the age they informed in connection to signing up to the camp, and not based on class in primary school. This meant all of them, except one 10-year-old, had not learnt multiplication tables at school, which made it conceptually challenging for them to solve the correct number of iterations in the last one of the subtask mines, that contained a two-level loop structure. This can also be seen as a result of the study, since they were, however, able to connect the mines to the correct one-level loop structures and count the right amount of iterations. With proper understanding on the level of mathematical development, however, the tasks could have been better targeted for this exact age group.

4.3 CONCLUSION

All in all, the Hello Ruby Summer school was very successful in increasing computational thinking perspectives in the participant children, who were glowing with excitement about technology and the creative collaboration it allows, which makes it a tangible continuation of Papert's legacy. However, a longitudinal research is required to analyse the actual conceptual change after the maturation of the summer school's effects. An interesting future research topic would also be the optimal developmental placement of the different

computational concepts into the timeline of a child's psychological and mathematical development. International, cross-cultural and, from the point of lifelong learning, cross-age group comparisons in the learning of computational thinking would be interesting ventures as well; Whether, for example, Asian people are more open to adopt computational thinking than European people, young people more than the elderly, etc..

The importance of computational thinking as a 21st century literacy cannot be emphasized enough, since it can already be seen so clearly, that we are programming robots to, little by little, take over our jobs. This will again lead to the importance of emphasizing new kinds of, more abstract and social skills, such as computational thinking, in the working life (Torkington, S., 2016; Rotman, D., 2015). An increased understanding of the more computational matters around us, I believe, also manifests as an increased spirit of collaboration, creativity and common understanding of the nature of the human mind and cognition in general. The increased understanding brought about by the summer school may have resulted in the conceptual change concerning artificial intelligence seen in the study, as well. Values, ethics and moral are extremely important matters to take into account at these times, since we are building the humane artificial intelligence based on machines and technologies we use everyday, that are fundamentally programmed to learn about us, our data, which will make it just as beautiful as our very own reflection in the mirror.

5 CONCLUDING REMARKS

This research was funded by Hello Ruby Oy and supervised by Professor of Educational Psychology Kirsti Lonka from the University of Helsinki.

6 APPENDICES

6.1 APPENDICE 1: THE RESEARCH FORM

(Turn the page)

| | Täysin eri mieltä | Eri mieltä | En osaa sanoa | Samaa mieltä | Täysin samaa mieltä | En ymmärrä kysymystä |
|--|-------------------|------------|---------------|--------------|---------------------|----------------------|
| Tietokoneet voivat suunnitella koteja | | | | | | ? |
| Tietokoneilla voi kierrättää | | | | | | ? |
| Tietokoneet ovat hyviä datan prosessoismissa ja käsittelyssä | | | | | | ? |
| Avaruudessa on tietokoneita | | | | | | ? |
| Robotit ovat tietokoneita | | | | | | ? |
| Tiedän mitä ohjelmointi on | | | | | | ? |

Mitä ohjelmointi on? Kerro lyhyesti alla oleville riveille!



Kiitooos!
Voit nyt siirtyä vielä tekemään irrallisen paperin tehtäviä!



Mikä fiilis?



Ei yhtään kiva



Kuin viilipytty



Ihan super!



Hyvin hämmäntynyt



Hei! Hei!
Ymmärräh, että haluat lapsesi onnistuvan ja on valkeaa olla auttamatta, mutta tutkimuksen kannalta on tärkeää, että hän vastaa itse!

Harrastukset, asenteet ja motivaatio

Seuraavat kysymykset liittyvät tämänhetkiseen olotilaasi, harrastuksiisi ja siihen, miten käytät teknologiaa. Ympyröi kohta, joka vastaa mielipidettäsi kustakin väittämästä parhaiten:

| | Täysin eri mieltä | Eri mieltä | En osaa sanoa | Samaa mieltä | Täysin samaa mieltä | En ymmärrä kysymystä |
|--------------------------------------|-------------------|------------|---------------|--------------|---------------------|----------------------|
| Mulla on tosi hyvä olo just nyt | | | | | | ? |
| Olisin mielummin jossain muualla | | | | | | ? |
| Tykkään tietokoneen käytöstä | | | | | | ? |
| Tykkään tablettitietokoneen käytöstä | | | | | | ? |
| Tykkään älypuhelimien käytöstä | | | | | | ? |
| Tykkään peleistä | | | | | | ? |
| Tykkään leikeistä | | | | | | ? |
| Tykkään kaverien kanssa tekailusta | | | | | | ? |

| | Täysin eri mieltä | Eri mieltä | En osaa sanoa | Samaa mieltä | Täysin samaa mieltä | En ymmärrä kysymystä |
|------------------------------------|-------------------|------------|---------------|--------------|---------------------|----------------------|
| Tietokoneet näkevät | | | | | | |
| Tietokoneet kuulevat | | | | | | |
| Tietokoneissa on sensoreita | | | | | | |
| Ihmiset ohjaavat aina tietokoneita | | | | | | |
| Ihmiset ohjaavat robotteja | | | | | | |
| Tietokoneiden sisällä on ihmisiä | | | | | | |
| Tietokone voi olla kuin ihminen | | | | | | |
| Robotti voi olla kuin ihminen | | | | | | |
| Tietokoneet ajattelevat | | | | | | |
| Robotit ajattelevat | | | | | | |
| Tietokoneet tietävät minusta | | | | | | |
| Tietokoneilla on tietoa minusta | | | | | | |
| Tietokoneet voivat kokata | | | | | | |
| Robotit voivat kokata | | | | | | |
| Tietokoneet ymmärtävät minua | | | | | | |
| Tietokoneet voivat luoda musiikkia | | | | | | |
| Tietokoneet voivat luoda taidetta | | | | | | |

| | Täysin eri mieltä | Eri mieltä | En osaa sanoa | Samaa mieltä | Täysin samaa mieltä | En ymmärrä kysymystä |
|----------------------------------|-------------------|------------|---------------|--------------|---------------------|----------------------|
| Tykkään matematiikasta | | | | | | |
| Tykkään laskemisesta | | | | | | |
| Tykkään numeroiden pyörittelystä | | | | | | |
| Tykkään ohjelmoinnista | | | | | | |

Tiedot, uskomukset ja ymmärrys

Seuraavat kysymykset on tarkoitettu selvittämään ajatuksiasi ja tuntemuksiasi tietokoneista ja teknologiasta, eikä niihin ole välttämättä yhtä oikeaa vastausta. Vastaa siis sen mukaan, mikä sinusta tuntuu oikealta. Ympyröi se kohta, mikä vastaa parhaiten mielipidettäsi.

| | Täysin eri mieltä | Eri mieltä | En osaa sanoa | Samaa mieltä | Täysin samaa mieltä | En ymmärrä kysymystä |
|--|-------------------|------------|---------------|--------------|---------------------|----------------------|
| Tietokone on sama asia kuin internet | | | | | | |
| Tietokone on sama asia kuin kone | | | | | | |
| Tietokone on sama asia kuin sähkö | | | | | | |
| Tietokone on sama asia kuin teknologia | | | | | | |
| Tietokone on matematiikkaa | | | | | | |
| Tietokone on numeroita | | | | | | |
| Tietokone on dataa | | | | | | |
| Tietokoneet tuntevat | | | | | | |
| Tietokoneet aistivat | | | | | | |

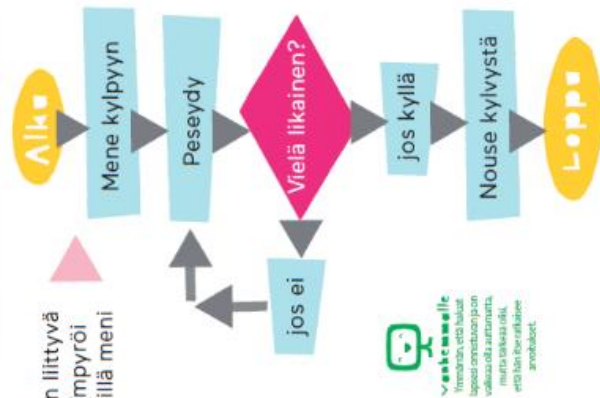


Seuraavassa pingviini pyytää apuasi parissa pulmassa.

Missä meni vikaan?

Pingviinillä on pieni peseytymiseen liittyvä ongelma. Kuinka auttaisit häntä? Ympyröi ne kohdat ohjeissa, missä pingviinillä meni vikaan.

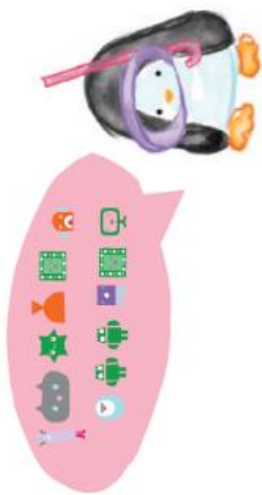
- A
- B
- C
- D
- E
- F
- G
- H
- I
- J
- K
- L
- M
- N
- O



vesikoneille
 Huomenna, että laitat lämpöä annosteluun ja on valittavana automaattinen lämmitys, jotta vesi on lämpimää.

Autteesi mitä?

Nyt pingviinillä on sinulle jotain sanottavaa. Pystytkö selvittämään vierelistä taulukkoa apuna käyttäen, mitä hän haluaa sanoa?



Pingviinin viesti:



Hei! Täyttäisitkö ystävällisesti alla olevat tiedot auttaaksesi tutkimustani?

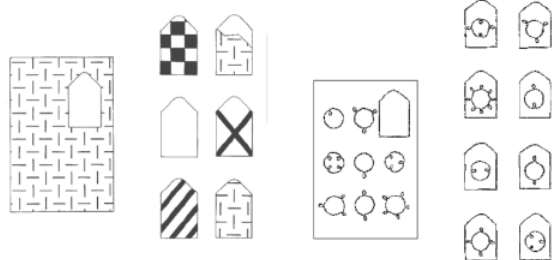
Nimeni on: _____ Olen tyttö Olen poika
 Olen _____ vuotta vanha

Huoltajieni ammatit ovat: _____

Oletko aiemmin tutustunut Hello Ruby -materiaaleihin?

No olenpa hyvinkin!
 En olepa ehkä tutustunut.

Hei, minulla on pieni pulma! Auttaisitko minua ympyröimään kustakin tässä olevasta arvoituksesta siitä puuttuvan palasen? Kiitos hurjasti!



6.2 APPENDICE 2: THE INTERVIEW FRAME

Hello Ruby Summer School: Ohjelmoinnillisen ajattelun haastattelurunko

Aloitus

Hei, mun nimeni on _____ ja mua kiinnostaa nyt tällä kertaa, miten sun ajatukset ja mielipiteet tietokoneista ja teknologiasta ja sun lempiharrastukset on ehkä muuttuneet tän kahden viikon aikana.

Kysyisin sulta seuraavaksi taas muutamia kysymyksiä. Jos joku niistä tuntuu vaikealta tai et ymmärrä tai halua vastata tai haluat lopettaa haastattelun kesken, niin sano mulle, niin voin selventää ja auttaa.

- Mitä sulle kuuluu? Minkäalainen olo sulla on juuri nyt?

Harrastukset, asenteet ja motivaatio

1. Mitkä on sun lempi harrastuksia tai minkä tekemisestä pidät eniten vapaa-ajalla?
 - a. Mikä näistä on sun ehdoton suosikkijuttu?
 2. Mitkä on sun lempi harrastuksia tietokoneella, tabletilla tai älypuhelimella?
 - a. Mikä näistä on sun ehdoton suosikki?
 3. Mistä aineista sä pidät eniten koulussa? Onko sulla yhtä isoa suosikkia? Onko sun ajatukset sun suosikkiaineista muuttuneet tän kahden viikon aikana
- Nyt mä haluaisin, että sä kerrot mulle vapaasti, että miten sä käytät tietokonetta, tablettia tai älypuheliminta tavallisen arkipäivän aikana, ja onko se muuttunut siitä kun sä ekan kerran teit tän haastattelun.

Tiedot, uskomukset ja ymmärrys

Seuraavaksi mä pyytäisin sua taas kertomaan mulle, että mitä nää seuraavat jutut sun mielestä on ja miksi ne on olemassa. Jos joku on erityisen vaikea tai et ymmärrä jotain näistä, niin kerro mulle, ja voin selventää tai auttaa sua. Tärkeää ei oo antaa oikeaa vastausta, vaan että sä kerrot vapaasti mitä sä näistä jutuista ajattelet.

1. Tietokone?
 2. Teknologia?
 3. Sähkö?
 4. Kone?
 5. (Äly)puhelin?
 6. Tabletti?
 7. Ohjelma?
 8. Ohjelmointi tai koodaus?
 - a. Jos osaa selittää, pyydä selittämään myös algoritmi ja syöte/tuloste, data.
 9. Sovellus/Appi?
 10. Peli?
 11. Media?
 12. Virus?
 13. Bugi?
 14. Taika?
 15. Robotti?
- Minkälainen olo sulla on nyt? Oliko nää kysymykset susta vaikeita? Oliko joku erityisen vaikea selittää? Tuntuuko että ne oli nyt helpompia kun edellisessä haastattelussa?

- Minkälaisia ajatuksia sulla on tietokoneista ja teknologiasta nyt tän parin viikon kesäkoulun/leirin/materiaalien lukemisen jälkeen?
 - Miten sun ajatukset on muuttuneet siitä, kun sä ekan kerran tulit tänne ja minkälaisia uusia ajatuksia sulla on herännyt?

Taidot

- Seuraavaksi mä pyytäisin sua tekemään vähän tehtäviä, ja tärkeätä ois, että sä puhut taas ääneen ajatuksiasi niitä tehdessäsi. Eli et pelkästään ratkaise niitä mielen sisässä, vaan puhut ääneen jokaisen askeleen. Se saattaa myös jopa helpottaa tehtävien ratkaisemista.

(Esimerkki)

(Aloita louhostehtävästä. Ei tarvitse tehdä jokaista tehtävää, jos tuntuu, että menee liian kauan tai haastateltava ahdistuu tai ei osaa ollenkaan tehdä. Pingviinitehtävän voi jättää pois jos menee liian kauan. Pingviinitehtävässä MUISTA KYSY MITÄ PINGVIINI OLISI HALUNNUT SANOA!)

Lopetus

Oliko sulla vielä jotain kommentoitavaa lopuksi näihin aiheisiin liittyen?

Kiitos kun osallistuit tähän haastatteluun!

6.3 APPENDICE 3: THE QUALITATIVE CHANGE T-TEST RESULTS

Independent Samples Test

| | | Levene's Test for Equality of Variances | | t-test for Equality of Means | | | | | | |
|---|-----------------------------|---|------|------------------------------|--------|-----------------|-----------------|-----------------------|---|---------|
| | | F | Sig. | t | df | Sig. (2-tailed) | Mean Difference | Std. Error Difference | 95% Confidence Interval of the Difference | |
| | | | | | | | | | Lower | Upper |
| Liittyvätkö selitykset kesäkouluun/hello ruby - materiaaleihin | Equal variances assumed | ,882 | ,368 | 11,153 | 11 | ,000 | 3,15000 | ,28244 | 2,52835 | 3,77165 |
| | Equal variances not assumed | | | 10,693 | 7,513 | ,000 | 3,15000 | ,29459 | 2,46292 | 3,83708 |
| Onko ohjelma-käsitteen selitys muuttunut tietokonekeskeiseksi? | Equal variances assumed | 7,474 | ,019 | 1,062 | 11 | ,311 | ,67500 | ,63564 | -,72403 | 2,07403 |
| | Equal variances not assumed | | | ,834 | 4,196 | ,449 | ,67500 | ,80971 | -1,53224 | 2,88224 |
| Jos osaa selittää ohjelmoinnin/koodauksen, osaako myös algoritmin? Datan? | Equal variances assumed | ,011 | ,917 | ,988 | 11 | ,344 | ,92500 | ,93636 | -1,13590 | 2,98590 |
| | Equal variances not assumed | | | ,988 | 8,630 | ,350 | ,92500 | ,93651 | -1,20745 | 3,05745 |
| Osaako selittää useampia käsitteitä? | Equal variances assumed | 1,359 | ,268 | 3,034 | 11 | ,011 | 2,02500 | ,66738 | ,55610 | 3,49390 |
| | Equal variances not assumed | | | 2,862 | 7,115 | ,024 | 2,02500 | ,70755 | ,35738 | 3,69262 |
| Onko käsitteet selitetty enemmän oikein/tarkasti ja yksityiskohtaisesti? | Equal variances assumed | ,445 | ,519 | 13,823 | 11 | ,000 | 3,07500 | ,22246 | 2,58537 | 3,56463 |
| | Equal variances not assumed | | | 13,038 | 7,115 | ,000 | 3,07500 | ,23585 | 2,51913 | 3,63087 |
| Onko käsitteet helpompi selittää? | Equal variances assumed | 3,026 | ,110 | 1,356 | 11 | ,202 | 1,05000 | ,77445 | -,65455 | 2,75455 |
| | Equal variances not assumed | | | 1,461 | 10,538 | ,173 | 1,05000 | ,71888 | -,54075 | 2,64075 |
| Onko lempiharrastukset /kiinnostukset vaihtuneet? | Equal variances assumed | 15,374 | ,002 | 3,103 | 11 | ,010 | 2,42500 | ,78157 | ,70478 | 4,14522 |
| | Equal variances not assumed | | | 3,859 | 8,478 | ,004 | 2,42500 | ,62842 | ,98997 | 3,86003 |

6.4 APPENDICE 4: THE ANCOVA TABLES

Descriptive Statistics

Dependent Variable: T2_EmbeddedCyberPhysicalSystems

| Vastaajaryhmä | Mean | Std. Deviation | N |
|---------------|--------|----------------|----|
| Hello Ruby | 3,9205 | ,69173 | 22 |
| Comparison | 3,1000 | 1,11332 | 20 |
| Total | 3,5298 | ,99573 | 42 |

Levene's Test of Equality of Error Variances^a

Dependent Variable: T2_EmbeddedCyberPhysicalSystems

| F | df1 | df2 | Sig. |
|------|-----|-----|------|
| ,092 | 1 | 40 | ,764 |

Tests the null hypothesis that the error variance of the dependent variable is equal across groups.

a. Design: Intercept + T1_EmbeddedCyberPhysicalSystems + RyhmäID

Tests of Between-Subjects Effects

Dependent Variable: T2_EmbeddedCyberPhysicalSystems

| Source | Type III Sum of Squares | df | Mean Square | F | Sig. | Partial Eta Squared |
|---------------------------------|-------------------------|----|-------------|--------|------|---------------------|
| Corrected Model | 21,518 ^a | 2 | 10,759 | 21,931 | ,000 | ,529 |
| Intercept | 6,195 | 1 | 6,195 | 12,629 | ,001 | ,245 |
| T1_EmbeddedCyberPhysicalSystems | 14,466 | 1 | 14,466 | 29,487 | ,000 | ,431 |
| RyhmäID | 3,522 | 1 | 3,522 | 7,179 | ,011 | ,155 |
| Error | 19,133 | 39 | ,491 | | | |
| Total | 563,938 | 42 | | | | |
| Corrected Total | 40,650 | 41 | | | | |

a. R Squared = ,529 (Adjusted R Squared = ,505)

7 REFERENCES

- Anto, E.; Penttinen, M., & Mikkilä-Erdmann, M. (2010). Kognitiivinen konflikti oppikirjatekstin lukuprosessissa ja silmänliikkeissä [Cognitive conflict in reading of textbook text revealed by eye movements]. *Kasvatus* 41(5), 445–455.
- 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.
- Bretton, C. (Oct 1, 2016). How deep learning allowed computers to see. TechCrunch. Retrieved from: <https://techcrunch.com/2016/10/01/how-deep-learning-allowed-computers-to-see/> .
- Burn-Callander, Rebecca (2015, June 6). Why women make gifted coders. The Telegraph. Retrieved from: <http://www.telegraph.co.uk/technology/11643191/Why-women-make-gifted-coders.html> .
- Chi, M. (1992). Conceptual change within and across ontological categories: Examples from learning and discovery in science. In R. Giere & H. Feigl (eds.), *Cognitive Models of Science*. University of Minnesota Press 129-186 (1992).
- Dewey, J. (2004). *Democracy and education*. Courier Corporation.
- Disessa, A. A., & Sherin, B. L. (1998). What changes in conceptual change?. *International journal of science education*, 20(10), 1155-1191.
- Dredge, Stuart (2014, January 27). Hello Ruby kids' coding book raises \$185k (and counting) on Kickstarter. The Guardian. Retrieved from <http://www.theguardian.com/technology/2014/jan/27/hello-ruby-kids-coding-book-kickstarter> .
- Eccles, J. S., & Wigfield, A. (2002). Motivational beliefs, values, and goals. *Annual review of psychology*, 53(1), 109-132.
- Elmes, D. G., Kantowitz, B. H. & Roediger, H. L., (2012). *Research methods in psychology*. Wadsworth: Cengage Learning.
- Gordon, G. (2014). Well Played: The Origins and Future of Playfulness. *American Journal of Play*, 6(2), 234-266.

- Guzdial, M. (2015). Learner-Centered Design of Computing Education: Research on Computing for Everyone. *Synthesis Lectures on Human-Centered Informatics*, 8(6), 1-165.
- Hakkarainen, K., Lonka, K. & Lipponen, L. (2004). *Tutkiva oppiminen: järki, tunteet ja kulttuuri oppimisen sytyttäjinä*. Helsinki: WSOY.
- Hakkarainen, K., Hietajärvi, L., Alho, K., Lonka, K., & Salmela-Aro, K. (2015). Socio-digital revolution: Digital natives vs digital immigrants. In J. D. Wright (editor-in-chief) *International encyclopedia of the social and behavioral sciences* (pp. 918-923). 2nd Edition, Vol 22. Oxford, England: Elsevier.
- Hello Ruby kids' coding book raises \$185k (and counting) on Kickstarter. (2014, January 27). *The Guardian*.
- Jackson, F. (1986). What Mary didn't know. *The Journal of Philosophy*, 83(5), 291-295.
- Jacobs, J., & Buechley, L. (2013, April). Codeable objects: computational design and digital fabrication for novice programmers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1589-1598). ACM.
- Kleine-Cosack, C. (2008). Recognition and simulation of emotions. Archived from the original on May, 28.
- Kupiainen, R., Kotilainen, S., Nikunen, K. ja Suoninen, A. (toim.), 2013. *Lapset netissä*. Helsinki: Mediakasvatusseura.
- Kurland, D. M., & Pea, R. D. (1985). Children's mental models of recursive LOGO programs. *Journal of Educational Computing Research*, 1(2), 235-243.
- Liukas, L. (2015). *Hello Ruby*. Helsinki: Otava.
- Lonka, K. (1997). *Explorations of constructive processes in student learning*. Helsinki: Yliopistopaino.
- Lukka, T. J., Tossavainen, T., Kujala, J. V., & Raiko, T. (2014). ZenRobotics Recycler—Robotic sorting using machine learning. In *Proceedings of the International Conference on Sensor-Based Sorting (SBS)*.
- Merikivi, J., 2013. *Still believing in virtual worlds*. Turku: Suomen yliopistopaino Oy – Juvenes Print.

- Mikkilä-Erdmann, M. (2001). Improving conceptual change concerning photosynthesis through text design. *Learning & Instruction*, 11(3), 241-257.
- Mikkilä-Erdmann, M. (2002). Textbook text as a tool for promoting conceptual change in science. Turun yliopiston julkaisuja. *Humaniora*, B, 249. (Dissertation).
- Mironov, V., Boland, T., Trusk, T., Forgacs, G., & Markwald, R. R. (2003). Organ printing: computer-aided jet-based 3D tissue engineering. *TRENDS in Biotechnology*, 21(4), 157-161.
- Nagel, T. (1974). What is it like to be a bat?. *The philosophical review*, 83(4), 435-450.
- Newell, A., Perlis, A. J., & Simon, H. A. (1967). Computer science. *Science*, 157(3795), 1373-1374.
- Oord, A. V. D., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., ... & Kavukcuoglu, K. (2016). WaveNet: A generative model for raw audio. arXiv preprint arXiv:1609.03499.
- Papert, S. (1972). Teaching children thinking. *Programmed Learning and Educational Technology*, 9(5), 245-255.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc..
- Penttinen, M., Anto, E., & Mikkilä-Erdmann, M. (2013). Conceptual change, text comprehension and eye movements during reading. *Research in Science Education*, 43(4), 1407-1434. doi:10.1007/s11165-012-9313-2
- Opetushallitus [The Finnish Board of Education] (2014). Perusopetuksen opetussuunnitelman perusteet. Helsinki. Retrieved from: <https://eperusteet.opintopolku.fi/#/fi/perusopetus/419550/sisallot/466344?valittu=428781>,
- Pea, R. D., & Kurland, D. M. (1984). On the cognitive effects of learning computer programming. *New ideas in psychology*, 2(2), 137-168.
- Picard, R. W. (1995). *Affective computing*. M.I.T Media Laboratory Perceptual Computing Section Technical Report No. 321.
- Prensky, M. (2001). *Digital Natives, Digital Immigrants*. On the Horizon, MCB University Press, 9(5).

- Resnick, M. (2006). Computer as paint brush: Technology, play, and the creative society. *Play = learning: How play motivates and enhances children's cognitive and social-emotional growth*, 192-208.
- Rotman, D. (June 16, 2015). Who Will Own the Robots? MIT Technology Review. Retrieved from: <https://www.technologyreview.com/s/538401/who-will-own-the-robots/>.
- Schmitz, A. (2014). Secret Consumer Scores and Segmentations: Separating Consumer 'Haves' from 'Have-Nots'. *Michigan State Law Review*, 1411.
- Searle, J. R. (1980). Minds, brains, and programs. *Behavioral and brain sciences*, 3(03), 417-424.
- Södervik, I., Mikkilä-Erdmann, M. & Vilppu, H. (2014). Promoting the understanding of photosynthesis among elementary school student teachers through text design. *Journal of Science Teacher Education* 25(5), 581-600. doi:10.1007/s10972-013-9373-9.
- Södervik, I., Virtanen, V., & Mikkilä-Erdmann, M. (2015). Challenges in understanding photosynthesis in an introductory biosciences class. *International Journal of Science and Mathematics Education*, 13(4), 733-750. doi:10.1007/s10763-014-9571-8.
- Stone, H. S. (1973). An efficient parallel algorithm for the solution of a tridiagonal linear system of equations. *Journal of the ACM (JACM)*, 20(1), 27-38.
- Torkington, S. (September 2, 2016). The jobs of the future – and two skills you need to get them. World Economic Forum. Retrieved from: <https://www.weforum.org/agenda/2016/09/jobs-of-future-and-skills-you-need> .
- Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59(236), 433-460.
- Vosniadou, S. & Brewer, W. F. (1992). Mental Models of the Earth: A Study of Conceptual Change in Childhood. *Cognitive Psychology* 24. 535-585.
- Vosniadou, S. (2013)(ed.) *International handbook on conceptual change*. Routledge.
- Wigfield, A., & Eccles, J. S. (2000). Expectancy–value theory of achievement motivation. *Contemporary educational psychology*, 25(1), 68-81.
- Williams, D., Ducheneaut, N., Xiong, L., Zhang, Y., Yee, N., & Nickell, E. (2006). From tree house to barracks: The social life of guilds in World of Warcraft. *Games and Culture*, 1, 338-36.