Coding in Schools

Comparing Integration of Programming into Basic
Education Curricula of Finland and South Korea

Contents

Introduction

"Coding is one of the most important skills that you can acquire today, and a critical skill for the Fourth Industrial Revolution", said Marc Benioff in a panel discussion at the annual meeting of the World Economic Forum 2016. A major theme of Forum was the **Fourth Industrial Revolution**, which refers to the idea of an era in which technology, such as artificial intelligence or robotics, is blurring the lines of physical, digital and biological spheres. Having the potential to create services, products and businesses never seen before, it may significantly affect our future lives, including job markets. In fact, it has been forecasted that a sizable proportion of current occupations will eventually be computerized and automated. So, nations, along with international organizations (such as European Union), are feverishly preparing to meet the challenges as well as welcome the possibilities of the new era.

Considering the relatively broad consensus on the role of programming skills in the new era, many countries have begun integrating coding in their public curricula or provide, at the very least, after school programs. A report by European Schoolnet (2014) sheds light on several initiatives in Europe to integrate coding into basic education. In this article, however, we take a brief look at to what extent coding or programming are integrated into the basic education systems and curricula in Finland and South Korea (hereafter Korea), which both have a firm reputation for a high quality of basic education (e.g. high performance in PISA tests). We also highlight some of the similarities and differences between these two countries related to the subject.

We start, however, by first outlining the definitions of terms related to coding, and reflect on the some of the reasons it has become an important skill in the 21st century. Why has code been called the 'Next Universal Language'?

1.

WHAT IS CODING, AND WHY HAS TEACHING CODING FOR CHILDREN SUDDENLY BECOME SO IMPORTANT?

There is a lot of terminology surrounding computer programming: code, algorithm, computational thinking, programming, to mention a few. We also encounter the term 'coding', often used in general discussion to refer to the new subject in schools.

- From a technological viewpoint, *computer programming* or *programming* essentially refer to a **problem solving process**, with the objective of creating a sequence of orders (algorithm) and resulting in a computer-generated, automated output of the desired outcome.
- *Computational thinking* (or *algorithmic thinking*) refers to the **thought processes** for analyzing the prerequisites for effectively solving a specific problem: articulating and formulating the problem, creating (coding) an automated expression of the solution and analyzing the outcomes.
- *Coding* refers to the practical **implementation of the algorithms into a programming language** (code), which could be also a visual programming language, i.e. Scratch.

In Finland, the terms used in the National Core Curriculum for Basic Education are *programming* (in Finnish *'ohjelmointi')*, *algorithmic* or *computational thinking* (in Finnish *'algoritminen ajattelu')*, and *ICT competence*, which encompasses wide array of ICT-related skills, including programming. In South Korea, the term *software education* is used in the National Curriculum for Basic Education of the Educational Reform 2015. The more specific terms describing learning objectives of software education include *programming, computational skills, algorithm, coding* and *problem solving.*

Most arguments on the importance of teaching programming to all citizens lean on a deterministic philosophy of the new technological imperative: the accelerating development of technology in the era of 4th Industrial Revolution is inevitable. Consequently, the most effective way to utilize it and maximize job creation is to

adopt and assimilate the new technologies as fast as possible. In the year 2020, it is forecasted that 825.000 jobs in Europe will be vacant in the IT field. In the United States, it is predicted that 47 % of all occupations are at risk to be computerized.

Besides in our future working lives, we tend to increasingly rely on devices and platforms running on algorithms in our present everyday-lives, too. Our daily information retrieval habits, for example searching for a recipe on Google, are steered by algorithms, deciding for us the order of relevance of the millions of search results. As a foundation of the platforms we rely on, code is by no means a neutral tool. As Lawrence Lessig (2000) described, "-- *code, or architecture, sets the terms on which life in cyberspace is experienced. It determines how easy it is to protect privacy, or how easy it is to censor speech*." Introducing programming in basic education strives to empower individuals by providing tools to exercise critical thinking and analyze the 'laws' of cyberspace they are subjected to.

Next, we will take a look at the curricula, some efforts in teacher training, support from organizations outside schools and the challenges Korea and Finland are facing introducing coding into their education systems.

2.

FINLAND: REFORMED CURRICULUM EMPHASIZES TRANSVERSAL SKILLS ALONGSIDE SUBJECT STUDIES

"Opportunities for pupils to develop their information and communication technology skills have been improved in all subjects, with technology being included more in instruction and study. --- Technology plays an increasingly significant role in everyday school routines, thus allowing pupils to be more easily involved in the development and selection of their own learning environments." (Finnish Board of Education 2014)

In Finland, the decennial reform process of the <u>National Core Curriculum for Basic Education</u> (primary and lower secondary) took place during 2012–2014. The new local municipality and school curricula, crafted on the basis of the national guidelines, were implemented from August 2016. The new national curriculum emphasizes, alongside subject studies, developing transversal competences: a combination of knowledge, skills, values, attitudes and will. The transversal competences are comprised of seven dimensions: *thinking and learning-to-learn, interaction and expression skills, managing daily life and taking care of oneself, multiliteracy* (the ability to produce and interpret diverse texts*), ICT competence, working life competence and entrepreneurship,* and *social participation and influence.*

According to the new curriculum, all of the subject studies ideally include developing all of the transversal competences, although the learning objectives of some subjects can emphasize certain competences. This means that all of the subjects should encourage, for example, developing ICT competence. In the curriculum, ICT competence is divided to four main areas of development:

- Students understand the principles and basic concepts of using ICT, and learn to develop practical ICT-skills via producing own content.
- Students are encouraged to use ICT responsibly, safely and ergonomically.
- Students are taught to use ICT in database management and in creative efforts.
- Students gain experiences and practice using ICT in communication and networking.

Programming has been integrated in the <u>National Core Curriculum</u> as part of both general ICT competence as well as the compulsory subject studies of **math**, beginning already in the first grade, and **crafts**, starting from the third grade. There are specifically outlined goals for the ICT competences for grades 1–2, grades 3–6 and grades 7–9. According to these guidelines, pupils start their journey in the world of programming by learning age-suitable programming, starting from the 3rd grade on focusing how human decisions influence the outcomes in programming, and eventually develop their coding skills as part of multiple school subjects.

As for programming in math, the emphasis is on developing algorithmic as well as computational thinking skills. Pupils are learning to first create basic sequences of instructions, then progressing to learn coding their own programs in visual programming environment, and, finally, learning to apply the principles of

algorithmic thinking into programming simple programs. As a learning objective integrated in craft, students practice in grades 3-6 functions based on programming, such as robotics and automatization. In grades 7-9, they learn to implement embedded systems or programming in designing and manufacturing products.

In Finland, most schools operate on public funding. The Finnish National Board of Education funds career updating courses for teachers, some of which also include programming. There are governmental initiatives to help ease the overall adjustment to the changes brought by the education reform, and to support the development of education in Finland overall. In addition to developing teacher training, the Finnish government has, for example, introduced a tutor-teacher system. Every school is granted a tutor teacher, who will support other teachers in actualization of the new pedagogy, and to accelerate the digitalization of teaching.

Apart from the courses and other support provided by the government, there are some grass-root initiatives to support teachers to learn programming. Especially noteworthy is Koodiaapinen, providing free MOOC courses on coding, which has (according to their own statistics) so far trained over 1000 teachers throughout Finland. Also, University of Helsinki provides free MOOC-courses, freely available for anyone interested in programming.

Somewhat exceptionally, there has been quite a lot of support from the IT-industry to help integrate coding into schools. Several IT-companies have sponsored courses in computer programming, providing free workshops in schools for pupils. For example, telecommunications, ICT and online service company Elisa organized an 'Elisa Digikoulu' -training' ('Elisa Digital School'), a free one-day workshop, during which instructors shared information on digital skills and introduced coding to pupils. Another initiative, Koodikoulu, organized by Futurice, Reaktor and Koodikerho, provides free materials and support for organizing after-school coding clubs.

Free materials have also been created to support specifically integrating coding into basic education. The founder of Rails Girls and author of Hello Ruby Linda Liukas has, together with Juhani Mykkänen, co-written a free guide book 'Koodi2016' giving 'first aid' and practical advice for the teachers who are uncertain how to approach the new field. As another initiative, a wikibook describing the objectives of programming in

basic education has been produced to support teachers in integrating programming into their teaching.

Despite of multiple initiatives, there are, however, also several challenges. The newest National Core Curriculum has been implemented quite recently, and included major reforms. Although the government has funded projects dedicated to support the development of local curricula and incorporate the new changes, it has simultaneously introduced major cuts on public sector funding, including education. The Trade Union of Education has voiced concerns about the overall sufficiency of government-funded refresher courses.

Although Finland has been well-known for equality in basic education, the PISA 2015 results show that the gap in performance levels between genders as well as different regions in Finland is widening. Girls have high skills in the areas tested in PISA, clearly outperforming boys in all areas. Among the OECD countries they were, however, the least interested in a career in a science- or technology-related field. Also some extra-curricular initiatives in Finland support specifically girls to learn programming (i.e. Rails Girls). On a larger scale, the recent education reforms aim to increase student engagement, and the future years will reveal how the changes influence the overall student motivation.

3.

SOUTH KOREA: SOFTWARE EDUCATION EMERGES INTO THE PUBLIC SCHOOL SYSTEM

Korea is one of the countries which have a reputation for high level of digital technology and infrastructure. According to Measuring the Information Society Report 2016 by ITU, Korea tops the ICT Development Index rankings in 2016 as the second consecutive year. This index reflects Korea's easy access to ICT, high frequency of using ICTs in society and high level of ICT skills. In spite of the high attention for ICT, programming or coding-related education has not been conceived as an important element so far for the national school curriculum of Korea.

However, Korea is now in the process of transforming its education system under the 2015 Educational Reform, and software education, which includes coding and programming, is one of the major changes over the whole reforms. Software education will be part of all school level's curriculum, including elementary, middle and high school, and even tertiary education, by 2019. The new curriculum is focused on developing computational thinking, coding skills, and creative expression by utilizing softwares. It is in line with the direction with the 2015 educational reform, which intends to cultivate individual's competences for future society.

In elementary schools, software education will be included in the course 'Sil-Gwa' which covers practical skills for daily-life (like Home-Economic). The amount of time is currently 12 hours, but it will be increased to 17 hours per year. The new curriculum states problem solving, algorithm, programming as the main contents. Unplugged education (e.g., board games) and visual programming languages will be used to stimulate students' interest in coding and software. Also, it will cover the information technology ethics, such as understanding software's copyright and desirable information technology usage. Middle school also has software education as a mandatory course with 34 hours per year. The course content focuses on problem solving with computational thinking, basic algorithms and computer program development. For high school, it will remain as a normal elective course and the main content will be using interdisciplinary thinking to build algorithms and computer programs.

Software Education under the 2015 educational reform is currently in its pilot phase. The Ministry of Education and the Ministry of Science, ICT and Future Planning selected 218 of leading software schools in 2015 and each selected school received financial support for the development of software education. In 2016, 682 additional schools were added, and currently 900 schools are supported by the government.

A critical issue for successful software education in Korea is teacher training. At the elementary school level, all subjects are taught by one responsible class-teacher, whereas the class teaching is led by a subject-teacher for the middle and high school level. Elementary school teachers, particularly, are faced with a challenges, as it is the first time for them to teach the subject without any previous experience or intensive training related to computer science. Aware of the situation, the Ministry of

Education set a plan to support the teachers. By 2018, 60,000 elementary school teachers (30%) will receive specialized training of software education and 6,000 among them will receive in-depth training. In addition, 1,800 middle school teachers who are certified as IT-teachers will receive additional training on software education.

Software education is supported from outside of school system, too. There are 30 of local media centers supported on municipality and government levels. They are running courses for coding or other programming-related activities for students. There are also many public-private collaborations to develop digital education platform. As an example, I-scream is a digital educational platform provided by Sigongmedia, a leading company in education market in Korea. It provides huge amount of digital contents for in-class teaching, and teachers can customize their coding lessons easily. Naver and Samsung are running Play with Software and Junior Software Academy respectively, so that students can learn coding or programming for free.

Korea is among the first countries to start the process of integrating programming into the national curriculum. There are, however, some challenges possibly hindering the smooth integration. One of the biggest obstacles could be the university entrance exam-centered school system. Despite the recent educational reform in 2015, the Korean educational policy continues to focus on the university entrance exams. Due to this, several major courses like Korean language, Mathematics, English are still regarded as a top priority. This hidden curriculum affects students not to spend their energy in practicing coding or programming.

Another challenge for teaching coding or programming in Korean school is the lack of consensus among students, schools, government and parents on the ultimate goal of software education. While software education in Korea was initially intended to develop students' ICT competence in the practical context for preparing future society, it has now resulted in the emergence of numerous private coding academies, helping students to achieve higher scores in the subject test. It shows that some of students and parents are regarding software education just as a specific subject they should master for exams, rather than as a practical tool which can be utilized to participate in future society.

4.

HEADING TO A NEW ERA

As we have so far observed, Finland and Korea seem to be, as a whole, sharing very similar objectives in terms of teaching programming and ICT skills. Both countries have proceeded to integrate programming in their national basic education curricula as a mandatory subject via profound curriculum changes in recent years. In addition to this, the goal of teaching programming is the same: to increase students' ICT competence, empowering them to actively participate in the future societies.

Especially in Korea, teacher training has gained considerable attention from the government. Software education is taught as a separate course, making it easy to design topic-specific and targeted teacher refresher courses. The Finnish government also supports teacher training, and those initiatives are supported by several grass-root and privately sponsored endeavors to provide training in programming. The Finnish education system is undergoing a fundamental reform, attempting to change the foundations of education, and moving towards simultaneously teaching transversal competencies alongside the subject studies.

There are also differences in how the countries approach integrating programming in the curricula for basic education. Finland has chosen a rather holistic approach by integrating programming as a transversal ICT competence, basically covering all subjects, accompanied by compulsory subject studies (*math* and *craft*) with detailed, subject-specific learning objectives. In Korea, on the other hand, software education is taught as a specific course, emphasizing the independent status of the subject The objectives of software education tend to emphasize how to utilize various software or how to do programming, but don't specifically encourage it as the method for learning other subjects, such as math or science.

Public, nor private, comprehensive schools are by no means the only places where it's possible to learn coding. Apart from private academies (especially popular in Korea), for example various clubs offer alternative ways to start developing one's skills in programming. It's also quite possible for even young children to take their first steps

in the world of code using, for example, games or vast free online resources, such as online tutorials, with their peers or by themselves.

During the era of 4th industrial revolution, coding is not only necessary 'survival skill'. Although coding undoubtedly ranks high in the priority list, the unpredictable future calls for cultivating and developing also other 21st century skills, such as collaborative skills, empathy, multiliteracy and learning to learn. We will sum up our thoughts with a quote from *Sheena Vaidyanathan*, a teacher in Los Altos School District in California:

"CS education (especially at the K-8 level) is definitely not about turning out coders or computer scientists. It is about teaching everyone to succeed in the new digital world, enabling them to solve problems using this powerful strategy."

5.

SOURCES

신상규. (2016). 자율기술과 플로리디의 정보 윤리. *철학논집, 45*(단일호), 269-296.

Ahshar, Vala (2016): "Coding and A Rookie Mindset Are Critical Skills for The Fourth Industrial Revolution". The Huffington Post 8.1.2016.

Bae, S. & Lee, J. (2014). *Research on institutional measure for youth's media education for digital age*. Seoul: National Youth Policy Institution.

Björksten, Tuomo (2016). *Uudet Pisa-tulokset: Suomen tytöt maailman toiseksi parhaita, vaikka ei edes huvita.* Yle. 6.12.2016.

European Commission (2015). Coding - the 21st century skill.

European Commission (2017). *Finland. Funding in Education.* Eurydice.

European Schoolnet (2014). Computing our future Computer programming and coding - Priorities, school curricula and initiatives across Europe. Belgium: European Schoolnet.

Finnish Ministry of Education (2016). *PISA 2015: Finnish youth still at the top despite the drop*.

Finnish Ministry of Education (2016). *Developing basic education over the years 2015 - 2019*.

The Finnish National Board of Education (2014). *The new curricula in a nutshell*.

The Finnish National Board of Education (2014). *Perusopetuksen opetussuunnitelman perusteet 2014*.

Frey, Carl Benedikt & Osborne, Michael (2013). *The Future of Employment: How Susceptible are jobs to computerization?* Oxford, United Kingdom: Ofxord Martin School.

Gillespie, Tarleton, Boczkowski, Pablo & Foot, Kirsten (2014). *The Relevance of Algorithms.* Cambridge, MA: MIT Press.

Ilomäki-Keisala, Ilona (2016). *Curriculum Design 2016 – Support Programme for the Curriculum Design Process*. Helsinki University Viikki Teacher Training School.

International Telecommunication Union (2016). *Measuring the Information Society Report 2016*. Geneva, Switzerland.

Konttinen, Matti (2016). *Koodaus puskee peruskoulun opetussuunnitelmaan – ministeriö turvautui yritysten apuun.* 20.2.2015. Yle.

Korean Ministry of Education (2015). *Korean National Educational Reform in 2015*.

Lessig, Lawrence (2000). *Code is Law.On Liberty in Cyberspace*. Harvard Magazine.

Liukas, Linda (2015). *Poetry of Programming*. TedxCern.

OECD (2016). *PISA 2015 Results (Volume I) - Excellence and Equity in Education*. Paris, France. DOI:10.1787/9789264266490-en

Ouakrim-Soivio, Najat, Rinkinen, Aija & Karjalainen, Tommi (2015). *Tulevaisuuden koulu*. Publications of the Ministry of Education in Finland 2015:8. ISBN 978-952-263-339-2

Rufina Kyung Eun Park (2016). *Preparing students for South Korea's creative economy: The Successes and challenges of educational reform*. Asia Pacific Foundation of Canada. Vancouver, Canada.

Schwab, Klaus (2015). *The Fourth Industrial Revolution - What It Means and How to Respond*. 12.12.2015.

Vaidyanathan, Sheena (2016). *What's the Difference Between Coding and Computational Thinking?* 6.8.2016.

Tiessalo, Paula (2016): *"OAJ: Opettajien täydennyskoulutus retuperällä" – peruskoulun uudistus vaatii henkilökunnalta paljon*. Yle. 4.8.2016.

World Economic Forum (2017). *Davos 2017 - Preparing for the Fourth Industrial Revolution*.

Further reading and resources

CS Unplugged

Finnish Ministry of Education and Culture

Hang Out with Software (소프트웨어야 놀자)

Hello Ruby

I-scream's (Sigong Media for Children)

Koodiaapinen

Koodikoulu

Korean Ministry of Education

Korean Ministry of Science, ICT and Future Planning

Koodi2016

Rails Girls

Samsung Junior Software Academy

Sigong Media (시공 미디어)

MOOC Courses in Programming (University of Helsinki)

Wikibook: Peruskoulun ohjelmointi ('Programming for elementary school')