J. Korean Earth Sci. Soc., v. 39, no. 4, p. 388−400, August 2018
https://doi.org/10.5467/JKESS.2018.39.4.388

ISSN 1225-6692 (printed edition)
ISSN 2287-4518 (electronic edition)

Exploring Students Competencies to be Creative Problem Solvers With Computational Thinking Practices

Young-Shin Park[1],* and Miso Park[2]

[1]Department of Earth Science Education, Chosun University, Gwangju, 61452, Korea
[2]Science and Culture Exhibition Department, National Science Museum, Gwangju, 61005, Korea

Abstract: The purpose of this study was to explore the nine components of computational thinking (CT) practices and their operational definitions from the view of science education and to develop a CT practice framework that is going to be used as a planning and assessing tool for CT practice, as it is required for students to equip with in order to become creative problem solvers in 21st century. We employed this framework into the earlier developed STEAM programs to see how it was valid and reliable. We first reviewed theoretical articles about CT from computer science and technology education field. We then proposed 9 components of CT as defined in technology education but modified operational definitions in each component from the perspective of science education. This preliminary CTPF (computational thinking practice framework) from the viewpoint of science education consisting of 9 components including data collection, data analysis, data representation, decomposing, abstraction, algorithm and procedures, automation, simulation, and parallelization. We discussed each component with operational definition to check if those components were useful in and applicable for science programs. We employed this CTPF into two different topics of STEAM programs to see if those components were observable with operational definitions. The profile of CT components within the selected STEAM programs for this study showed one sequential spectrum covering from data collection to simulation as the grade level went higher. The first three data related CT components were dominating at elementary level, all components of CT except parallelization were found at middle school level, and finally more frequencies in every component of CT except parallelization were also found at high school level than middle school level. On the basis of the result of CT usage in STEAM programs, we included ‘generalization’ in CTPF of science education instead of ‘parallelization’ which was not found. The implication about teacher education was made based on the CTPF in terms of science education.

Keywords: computational thinking, STEAM, creative problem solver, CTPF (computational thinking practice framework).

Introduction

The revised science curriculum of Korea (MOE, 2015) and the release of the Next Generation Science Standards (NGSS) put the emphasis of equipping students with competencies necessary for surviving in the 4th industrial revolution (NRC, 2012). NGSS included eight different practices for learners’ competencies for authentic investigation in the classroom. Most practices emphasized from NGSS (NGSS Lead States, 2013) are understandable and familiar to in-service teachers except one; computational thinking from 5th practice named by ‘mathematical thinking and computational thinking’ (Park and Park, 2017; Weintrop et al., 2016). In newly 2015 revised curriculum of Korea, government also emphasized the inclusion of software and technology skill from elementary level and similar computational skills are expected to be obtained in science curriculum, which focus on students’ skills of using computer during science. Computational thinking (CT after this) skills are considered as critical competency for students to be equipped with as creative problem solvers now and in the next generation (Park and Hwang, 2017; Weintrop et al, 2016).

However, the inclusion of CT offers little guideline

---

*Corresponding author: parkys@chosun.ac.kr
Tel: +82-62-230-7379
Fax: +82-62-608-5346

for teachers who would train students to be creative problem solvers. In this study, the researchers would introduce how one framework (CTPF) including 9 components of computational thinking and their definitions from the perspectives of science education has been developed, and how this tool, CTPF, has been validated by applying it into science program (the selected STEAM program in this study). This study has its significance in STEAM education. In Korea, many teachers are still struggling in how to implement T and E into STEAM program into science class and CT practice can bridge T and E into STEAM context more easily. Therefore, if a guideline for applying CT into STEAM program is provided, science teachers can easily employ T and E by CT framework for completing STEAM program (Park and Park, 2017; Psycharis, 2018; Weintrop et al, 2016).

We have many specific trends and changes in the 21st century and science education cannot be exceptional to face new trends in teaching and learning (MOE, 2015; NGSS Lead States, 2013). The purpose of science education is called, ‘scientific literacy’, through which we expect students to be more literate scientifically. Students are trained to be more rational decision maker for SSI (social scientific issue) to see if SSI is right or wrong. For this literacy, students learn majority of scientific concepts through experimentation from their curiosity. If necessary, students make claims on the basis of evidences through argumentation and they also consider science ethics in their decision if SSI is right or wrong. Students do not have to take any action by doing science to be literate, which is not explicitly described in the document of scientific literacy (NGSS, 2013; MOE, 2015). In the document of science standards, students are described only to understand the natural and physical phenomena fully and they present their position about any SSI (Park, 2010).

However, making decision is not the end of scientific literacy. We need to go beyond scientific literacy defined in the 20th century. Students must recognize what problems are, how those problems are impacting on human life, how they decompose those problems from the given phenomenon to be researchable, how they develop possible products as solutions, and how they generalize those solutions to be applicable to most contexts. Students need to be exposed to be trained to experience the process of algorithmic problem solving practices and applications with the help of some competencies, which we call computational thinking practices (Barr and Stephenson, 2011; Park and Hwang, 2017; Wing, 2006; Wing, 2008). Students’ competencies of using this practical process can be new purpose of science education and the extended meaning of scientific inquiry in the 21st century, creative problem solvers.

The term of computational thinking was introduced by Wing (2006), saying that computational thinking is taking an approach to solving problems, designing systems and understanding human behavior. STEM/STEAM education across the world have also focused on the similar sequential practices related to computational thinking (Barr and Stephenson, 2011; Park et al., 2017; Weintrop et al., 2016). Students through STEAM education are expected to learn some concepts and apply those concepts to produce artifact as the final product for solving the problem. However, many science teachers in Korea were struggling to employ T and E into STEAM to be completed. So, teachers would only provide students with chances of making artistic products like strong chair made from cardboard, egg dropper etc. However, students in the 21st century are expected to be more creative in solving the problem to find out the practical solution (Park, 2018; Song and Na, 2014; Stohlmann et al., 2012). Students must be motivated to recognize real issues in the community or society they belong to, they need to figure out how those issues can be breakable to be more handled, they design/build model and simulate it to be applicable into most contexts, and finally they generalize the answers at the end (Park, 2018).

Park and Hwang (2017) developed one tool consisting of five computational thinking practices with 23 indicators; *connecting computing, developing computational artifacts, abstracting, analyzing problems and artifacts,* and *communicating and collaborating.* This tool has been developed from theoretical documents including NGSS, which reflect the view of science education area. Since there had been documents about computational thinking from the view of technology or computer science disciplines (ISTE, 2011), the researchers were motivated to develop analytical tool which could be used in science education, which we call, CT_STEAM_AT (computational thinking in STEAM analyzing tool). Park and Hwang (2017) released that we could describe or illustrate how much STEAM program reflect on computational thinking practices and how those CT practices were connected one another. This practice is more descriptive like prescription, saying which kind of CT practice was more dominating than others and how the limited practices could be promoted on the basis of the results through CT_STEAM_AT. The result of this study, for example, showed that STEAM program of climate change theme included rich CT practices such as connecting computing, abstracting, and communicating and collaborating, on the while, developing computational artifacts (DCA) and analyzing problems and artifacts (APA) showed the limited since the pre-developed only one photo bioreactor equipment with green algae was used equally for all students to check how it worked. If students had chances to develop the equipment by their own purpose variously, then DCA and APA might have been promoted in that students evaluate the pros and cons of each their developed equipment and select the best solution on the basis of various views. This study described CT by observable practices in the classroom or program qualitatively so it is not useful to compare which program, which classroom, or who has more CT practices in teaching among them. Therefore, this study can provide another way of describing CT inclusion quantitatively (Park and Lawrence, 2015; Park, 2018).

In this study, the researchers introduce another framework consisting of 9 components of CT defined by ISTE and CSTA (2011); however, those 9 components of CT had been modified with the view of science education at the 1st step then we employed this framework into STEAM program to construct its validation at the 2nd step. This CTPF (component of CT practice framework) can be used as evidences to describe what kinds of CT component are included in science program. More frequently used components can be found to profile if any science program is emphasized in the step(s) of ‘leading’, ‘building’, ‘connecting’, or ‘practicing’ (ISTE and CSTA, 2011). If we pursue to meet the goal of science education, creative problem solvers with scientific literacy, computational thinking practices are pivotal to be equipped with (Wing, 2006; 2008; Park and Hwang, 2017; Park and Park, 2017).

Methodology

There are two steps to develop and validate CTPF (computational thinking practice framework) in this study.

Exploring and developing one framework of CT components for science education

The researchers imported 9 terms of CT used in computer science discipline (CSTA, 2011); *data collection, data analysis, data representation, problem decomposition, abstraction, algorithms & procedure, automation, simulation,* and *parallelization.* The researcher team consisting of science educators and teachers as experts discussed several times if those components of CT can be applicable into science program for the purpose of equipping students with abilities needed as problem solvers and produced one framework consisting of 9 computational thinking terms with the modified operational definition from the perspectives of science education. The team employed this framework into science program to see if each component of CT was applicable and implemented in any science content (10 different programs, each program consists of 3 or 4 blocks of lessons) and the research team confirmed or validated the former proposed CTPF components with definitions. If any CT component was not applicable with definition, the team discussed and add or modified operational definitions, which produced the first version of CTPF.

Validating CT components defined from the view of science education

To construct the validity of CTPF, the researchers implemented this framework to STEAM program (climate change and water shortage) to see if all components of CT could be applied into the context of STEAM program. The reason why the researchers selected STEAM program for its validation was that the goal of using CT and STEAM is the same; train students to be more creative problem solvers. After this validation, more modification or addition were taken to finalize the CTPF.

Results

First, modified 9 components of CTPF in science education are described. Second, the process of validating CTPF and its application to STEAM program are illustrated also. Third, the connection with CT framework at NGSS (2013) will be provided also for another validity process.

Nine components of CT with generalization in science education

The research team (two science educators and three science experienced teachers) discussed all 9 components of CT from computer science and decided if those components could be found in science learning. The first 3 (data collection, data analysis, data representation) were captured easily in science program, but the last 6 (problem decomposition, abstraction, algorithms and procedure, automation, simulation, parallelization) were not typical found in science program. The research team discussed about those possibilities to be applicable in science program. <Table 1> contains various definitions from the theoretical literature. The first three definitions from literature review and the researchers compared them to be finalized in the last one of each cell, 'definition' of each CT. In Table 1, the 1st definition is from KOFAC (2014), 2nd definition from ISTE and CSTA (2011), and the 3rd definition from Barr and Stephenson (2011). The last one was called CTPF_v1 (computational thinking practice framework) which the researchers proposed for CT to science education.

Profiles of CT usage from K to 12 in applying CTPF to STEAM program

STEAM programs at 4 different levels (lower elementary, upper elementary, middle school, and high school) with two topics (climate change, water shortage) were analyzed to see if CTPF_v1 was applicable for constructing the validity.

Profile of CT components in STEAM program at lower elementary level: At elementary level (Table 2), the first three components of CT (data collection, data analysis, data representation) were more frequently used at lower elementary level in both topics, climate change and water shortage. However, the rest of CT components (algorithm and procedure, automation, simulation, parallelization) were not found at all at lower elementary at climate change STEAM program, but there were three times of problem decomposition, once algorithm and procedure, and once simulation at water shortage STEAM program. No parallelization was found.

In climate change, students had chance to watch the video of weather caster or read news article about the weather then they collected the vocabularies which they encountered most about weather. Students then discussed what scientific terms were related to weather. This is the process of students to *collect the data, analyze them to be represented* by investigation. Students also used the given data (records of the amount in precipitation, snow, or number of cloudy and sunny day separately for the last 10 years) to represent them yearly.

In water shortage, students were asked what they would do if there were no water in daily life. Students were motivated to investigate how much water is important in their lives. Students collected the data of water resources in the globe and investigated how

Table 1. Computational Thinking components with Operational Definitions

CT	Definition
Data Collection	1. The process of collecting the right data related to the problem that needs to be addressed. 2. The process of gathering appropriate information 3. Collect data from an experiment → ***Collect relevant data through experiments or investigation to solve the problem***
Data Analysis	1. Understand data, find patterns and draw conclusions 2. Making sense of data, finding patterns, and drawing conclusions 3. Analyze data from an experiment → ***Analyze and understand data to find patterns and draw conclusions***
Data Representation	1. Organize your data into appropriate graphs, charts, text, and pictures 2. Depicting and organizing data in appropriate graphs, charts, words, or images 3. Summarize data from an experiment → ***Transform the analyzed data to be readable or interpretative through graph, chart, writing, or drawing***
Problem Decomposition	1. Divides problems into small solvable problems 2. Breaking down tasks into smaller, manageable parts 3. Do a species classification → ***Divide problems into manageable parts/factors***
Abstraction	1. Identify key elements that are essential to troubleshooting and simplify complexity. 2. Reducing complexity to define main idea 3. Build a model of a physical entity → ***Identify key elements for troubleshooting and simplify complexity by weighing those elements differently***
Algorithms & Procedures	1. A series of steps that are taken to solve a problem or achieve a certain goal. 2. Series of ordered steps taken to solve a problem or achieve some end 3. Do an experimental procedure → ***Explore and choose sequential steps in order to solve a problem or achieve a goal after all possible trial and errors***
Automation	1. Displays solutions in a form that the computing system can perform 2. Having computers or machines do repetitive or tedious tasks 3. Use probe ware → ***Present solutions in a form that suggested solution can be applicable to any context without problem***
Simulation	1. The result of automation, run a model you created to solve the problem and get the results 2. Representation or model of a process. Simulation also involves running experiments using models. 3. Simulate movement of the solar system → ***Run the sequential program on the basis of scenario as the solution***
Parallelization	1. Organize resources to simultaneously work to achieve objectives 2. Organize resources to simultaneously carry out tasks to reach a common goal. 3. Simultaneously run experiments with different parameters → ***Perform operations simultaneously with different parameters to achieve a goal***

Table 2. Profile of CT components in STEAM program ('climate change' & 'water shortage') for lower elementary level

	Climate change					Water shortages					
	1	2	3/4	5	6	1	2	3	4	5	6
Data Collection		○	○	○				○		○	
Data Analysis	○	○	○	○		○	○	○		○	
Data Representation	○	○	○	○	○	○		○	○	○	○
Problem Decomposition		○	○			○			○		○
Abstraction		○	○								
Algorithms & Procedures							○				
Automation											
Simulation									○		
Parallelization											

much and for what they used water daily through writings or drawings. Students discussed in a group what would happen if they could not use the water and became to know that Korea is water shortage country. Student faced of how to solve this problem of water shortage and they tried to break this problem into manageable factors, like saving water from taking shower or dishwashing, watering to the garden, and they thought of recycling water by refining water and storing it. At this moment, students had chance to *decompose the problem into factors* in saving water and they discussed what kinds of water resource there are and how much they could save, which we can call is the *process of abstraction.*

Students learned some technologies of saving water or storing water from rainfall through the examples of Japan and Germany, and they discussed how they could save water individually or by the unit of community (house or school unit here). Then students tried to make a equipment of saving water by considering concepts they learned, which is *algorithm and procedure.* Students also tested some natural materials efficient for making rainfall or dirty one cleaner and they chose most efficient materials in order to get the clean water, which is *'simulation'* CT component. Since there was not certain guide line or algorithm to be adopted but they just made trial and error for the most cleaned water by cleaning water equipment, therefore, algorithm and procedures with *automation* were missing in choosing the best materials for water purification.

Profile of CT components in STEAM program at upper elementary level: The pattern in using CT components at upper elementary level (Table 3) is similar to those of lower; the first three data related CT components were dominating at both topics of STEAM program, and problem decomposition component with abstraction were found at the last lesson of climate change program, and the last three lessons at water shortage. No parallelization was found.

Climate change STEAM program covered the following contents; why temperature is increased, why North Pole is hotter, why the albedo on earth is changing, and if the earth is green house in each lesson orderly. Students had chances of *collecting the data*, *data analysis, and data representation* in these lessons. For example, students measured the albedo with the use of illuminometer from the ice and water in experimentation, where they *collected the data,* which is CT component. Students also wrote their opinions to be shared in groups about changing seasonal fruit type and harvesting tropical fruits, where *data representation* CT was caught. Students measured the temperature difference between in and out of the filmed box and interpreted why it differed, which was *data analysis* CT component. In the last block of lessons, students read one article about CO_2 role in climate change, which they faced to solve it. Students tried to find out what made CO_2 increase in daily life, which was the process of problem *decomposition.* Then, students recognized which resource from their

Table 3. Profile of CT components in STEAM program ('climate change' & 'water shortage') for upper elementary level

	Climate change						Water shortages					
	1	2	3	4	5	6	1	2	3	4	5	6
Data Collection	○	○	○	○	○	○	○	○	○	○	○	○
Data Analysis	○	○	○	○		○	○	○	○		○	
Data Representation	○					○	○	○			○	○
Problem Decomposition						○				○	○	○
Abstraction						○				○	○	○
Algorithms & Procedures												○
Automation												
Simulation										○		
Parallelization												

lives caused CO_2 to be produced from the most to the least, which is *abstraction*.

Water shortage STEAM program consisted of what water cycle is, what would happen if water cycle is broken, why city water is not enough, what solution can be possible for water shortage, how to plan community with rich water, and how we can be water guard. Students experienced *data collection, data analysis, and data representation* related to water cycle and its unbalance. Students became to face that water is running out of due to the decrease of wetland, increase of farming land, increase of residence area from development of city. Students were motivated to develop community of eco-friendly housing, low CO_2 and green growth, this was abstraction CT component. Students, first, felt the necessity of storing water and created it with the use of materials they investigated, where simulation was captured indirectly. Students test it and took it as the solution for efficient water storage. Finally, students planned the community by reducing water loss and storing water through special technology, which distributed rainwater management. Students learned how eco-friendly porous sidewall influence, how recycled permeable pressed block for water tire is developed for water saving, and how rooftop green technology is benefit, all of which processes included *abstraction, but indirectly.* Finally students became to know what roles they could take individually for water shortage in the community (algorithms and procedure), where students selected some skills best for their community.

Profile of CT components in STEAM program at middle school level: At middle school level (Table 4), all components of CT were covered except parallelization; components of CT (data collection, data analysis, data representation, decomposing, abstraction, algorithm and procedure) were found from the 1st block to 7th one, automation was found in 7th and 8th blocks, and simulation was found at 9th block of lessons in climate change STEAM program. Similar patterns of CT components usage in water shortage STEAM program was found; all except parallelization were found from the 1st to the 10th block of lessons. Two times of *simulations* were found at 5th and 9th each. We can say that there is more chance for students at middle level to experience more components of CT than those of elementary levels. Students were trained in *collecting data, analyzing data, representing data, decomposing problems* to be abstracted for developing algorithms and procedures, all of which were similarly dominant overall. However, two CT components (*automation and simulation*) were included but not dominant. The last one parallelization was not found.

In climate change STEAM program at middle school levels, students investigated to see how the temperature was changing for the last 50 years, they experimented to see which gas was the main one for greenhouse effect, and they also experimented to select the most real material for greenhouse effect (similar to atmosphere in the real world) and the first three data related CT components were dominating. Then students faced the problem for climate change

Table 4. Profile of CT components in STEAM program ('climate change' & 'water shortage') for middle school level

	Climate change									Water shortages									
	1	2	3	4	5	6/7	8	9	10	1	2	3	4	5	6	7	8	9	10
Data Collection	○		○	○	○					○		○				○			
Data Analysis		○	○	○	○					○	○	○				○			
Data Representation		○							○	○		○				○			○
Problem Decomposition	○	○			○	○				○	○	○	○		○		○		
Abstraction	○	○			○	○				○	○	○	○		○		○		
Algorithms & Procedures					○	○							○	○	○			○	
Automation						○	○						○	○					
Simulation							○							○				○	
Parallelization																			

and discussed its solution, which consisted of reducing CO_2 or consuming it actively (*problem decomposition*), then finally students and the teacher decided to use photo bioreactor for both of solution (*abstraction*). Students learned about green algae and they discussed about photo bioreactor's efficiency in their resident area and they made it in a group (*algorithms and procedure*). Students provided green algae for two weeks and checked its photosynthesis by color in the tank (*simulation*). However, students did not have a chance to produce oil from green algae after drying them.

In water shortage, students learned many concepts about water shortage and they became to design the equipment for purifying rainfall to fresh water to make up for water shortage. There had been many chances for students to get to know concepts of water by reading articles and doing simple experimentation (here, we can find data related CT components during learning concepts of water). It was introduced one article showing how Singapore solved water shortage by water cluster project which motivated students create idea of how to solve the same water problem for the community and our country (where problem *decomposition* CT component was found). Students investigated what factors cause Korea into water shortage (*abstraction*). Students investigated all possible factors to the solution (precipitation pattern, total amount of water resources /volume and usage), characteristics of water supply and causes of water shortage, and available water resources. Then students evaluated water status of individual home and school as well as community and they learned all new and old technology helpful for water shortage according to situation (problem *decomposition, abstraction, algorithms and procedures* CT components were possible). In this case, Students created plan or scenario of how our home/school/community could save the water (automation is possible, since they did not run the output but designed it with all consideration). One simulation was from students' developing seawater desalination equipment. However, no parallelization was found.

In general, elementary levels has more dominating usage of the 1st three data related CT components, on the other hand, middle school levels has distributed usage of CT components dominating over the lessons except the last two (automation, simulation), which appeared limited and the parallelization was not found at all at elementary and middle school levels.

Profile of CT components in STEAM program at high school level: At high school level, all components of CT except parallelization were found more frequently in each lesson at both of STEAM programs, climate change and water shortage. The difference between middle and high school levels was that there were more blocks where students could experience all components of CT except parallelization. We can illustrate that more components of CT were found in each lesson at high school level than middle school one. For example, the number of CT components from 10 to 60% were found in each lesson at middle school levels. In the 1st lesson, there were 5 CT components and there were only 1 in the last lesson. On the other hand, the number of CT components from 60 to 100% (all of 8 components were found in one block) were found at high school levels at both topics of STEAM program. In the 4th and 5th lessons of climate change and 6th and 7th lessons of water shortage, all CT components except parallelization were included.

In climate change STEAM program, the content was much overlapped with that of middle school one. Students did a few different experiments about CO_2 causing climate change mathematically with the use of formulas. Students became to know that we can develop plastics from CO_2 which is one of ways to consume CO_2. Students became to know that there are two ways for solving climate change; one is consuming CO_2 and another reducing CO_2, both of which were used in photo bioreactor equipment. Students discussed of how to save our community (all CT components were included except parallelization).

In water shortage, students learned what waterworks are and how long they had been used. Students learned the best example of waterworks in the world

Table 5. Profile of CT components in STEAM program ('climate change' & 'water shortage') for high school level

	Climate change									Water shortages									
	1	2	3	4-5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
Data Collection	○	○	○	○		○	○		○	○	○	○	○	○	○	○	○		○
Data Analysis	○	○	○	○		○	○		○	○	○	○	○	○	○	○	○		○
Data Representation	○	○	○	○		○	○		○				○	○	○	○	○		
Problem Decomposition	○	○	○	○		○	○		○			○	○	○	○	○	○		
Abstraction	○	○	○	○		○	○		○			○	○	○	○	○	○		
Algorithms & Procedures		○		○					○				○	○	○	○	○		
Automation		○		○				○	○				○	○	○	○	○		
Simulation				○	○			○	○					○		○		○	
Parallelization																			

and they discussed how waterworks were good or not to our country. Students constructed water pipes for waterworks in the community through application games and they also learned about sewage. Students also learned how water pollution was caused from sewage. Students faced problem (to save community/our country from water shortage) and decomposed this into manageable factors (water purification, water reuse, storage, and sewage) to plan some procedures to save our community/school. For water purification, students picked 5 from 8 materials which were good in efficiency and economics for water purification on the basis of students' decision. Then students carried out the experiment consisting of 5 different material effective for water purification from rainfall. All of these processes, students experienced *algorithm*, *automation*, *and simulation* but parallelization.

In general, the profile of CT components in STEAM program showed the CT spectrum covering from data collection to parallelization as levels went higher. The first three data related CT components were dominating at elementary levels, all components of CT except parallelization were found at middle school, and finally more frequencies in each component of CT were found at high school level than middle school one. Therefore, CTPF_v1 (without parallelization) was well defined practice framework which was applicable to decide if any program include components of CT, how much it includes, and how it can be promoted. Finally, the researchers discussed if parallelization could be included or excluded. In STEAM program, the research team concluded that generalization can be more appropriate instead of parallelization. Generalization was added with the following definition; apply the product in various types of context if it is working or not. This process will validate its product as a solution.

Now, there are operational definitions with 9 components of CT in science education through discussion with science educator and science teachers. One thing with big change is 'generalization' from 'parallelization', which were not found at any science programs selected and analyzed in this study. In science program, the meaning of parallelization by computer science (run the same program at different computers to save time) was not appropriate and the research team did not find any exact parallelization in any science program. In addition, Barr and Stephenson (2011) defined this parallelization as follows; run experiments simultaneously with different parameters, which is the process for collecting data. For example, after students in a group do experimentation to find out the factors influencing plants' photosynthesis with different parameters such as the amount of water, the amount of light and its lasting time, the amount of nutrition and so on, then this process is for collecting the data to be combined for holistic description of the phenomenon. Therefore, the researchers defined that generalization is normally coming after simulation, where students are expected to run each simulation at different context to be confident that simulation as the solution is applicable to most cases of contexts, therefore, generalization into final *CTPF* was added at the end.

Connection to CT practices of NGSS to CTPF for the validity

Lastly, the connection between CT practices of NSGG and CT components included in STEAM program in this study was added to construct CTPF's validity. In NGSS (2013), there are description about expected CT practices in grades from K to 12. For example, at lower elementary level, students could have chance to decide if they could collect the data quantitatively or qualitatively, and actually they collect the data. In STEAM program of elementary levels, 1

Table 6. Connection between CT practices of NGSS and CT components included in STEAM program

K-2	3-5	6-8	9-12
Decide with to use qualitative vs. quantitative data. ■ ■	Decide if qualitative or quantitative data are best to determine whether a proposed object or tool meets criteria for success. ● ▲ ▲ ● ●		
• Use counting and numbers to identify and describe patterns in the natural and designed world(s) ■ ■ ■ ■	• Organize simple data sets to reveal patterns that suggest relationships. ■ ■ ▲ ■ ■ ■ ●	• Use digital tools(e.g., computers) to analyze very large data sets for patterns and trends ● ▲ ▲ ▲ ▲	• Create and/or revise a computational model or simulation of a phenomenon, designed device, process, or system ▲ ▲
• Describe, measure, and/or compare quantitative attributes of different objects and display the data using simple graphs. ■ ■ ■ ■	• Describe, measure, estimate, and/or graph quantities such as area, volume, weight, and time to address scientific and engineering questions and problems ● ● ● ■ ■ ●	• Use mathematical representations to describe and/ or support scientific conclusions and design solutions ● ▲ ▲ ● ▲	• Use mathematical, computational, and/or algorithmic representations of phenomena or design solutions to describe and /or support claims and/or explanations ▲ ▲ ▲ ▲
• Use quantitative data to compare two alternative solutions to a problem. ■ ■ ■ ■ ■	• Create and/or use graphs and/or charts generated from simple algorithms to compare alternative solutions to an engineering problem ■ ■ ● ●	• Create algorithms (a series of ordered steps) to solve a problem • Apply mathematical concepts and/or processes (such as ratio, rate, percent, basic operations, and simple algebra) to scientific and engineering questions and problems ▲ ● ▲ ▲ • Use digital tools and/or mathematical concepts and arguments to test and compare proposed solutions to an engineering design problem ● ● ● ▲ ▲ ● ●	• Apply techniques of algebra and functions to represent and solve scientific and engineering problems • Use simple limit cases to test mathematical expressions, computer programs, algorithms, or simulations of a process or system to see if a model "makes sense" by comparing the outcomes with what is known about the real world. ● ● ▲ ▲ • Apply ratios, rates, percentages, and unit conversions in the context of complicated measurement problems involving quantities with derived or compound units (such as mg/mL, kg/m^3, acre-feet, etc.)

■ Elementary level program of STEAM
● Middle school level program of STEAM
▲ High school level program of STEAM

or 2 dominating CT components from each lesson (a lesson did not show any dominating CT component, for example, students just presented all findings at the last lesson. In this case, there was not found CT component) were marked in each cell of NGSS CT description table.

In Table 6, there are different types of shapes indicating the relationship between CT description of each level in NGSS and those of CTPF analyzed in STEAM program. Square represents dominating CT component at elementary level, Circle at middle school level, and Triangle at high school level. The first line with different shapes (square, circle, and triangle) were from climate change analysis. The second line with different shapes from water shortage analysis.

The researchers picked one dominating/outstanding CT component from each lesson and marked each shape to the applicable description of CT practices in NGSS (2013). The pattern of CT inclusion in STEAM from elementary to high school is changing more and more in its frequently usage from K to 12 in NGSS. The CT component of elementary level are covering from K to 5, CT of middle school level from 3 to 8, and CT of high school level from 6 to 12. STEAM program of middle school levels can include all CT descriptions of K to 5 of NGSS and STEAM program of high school levels can also over all CT descriptions of 6 to 8. Overall, the CTPF with generalization is validated in its usage, therefore, it is significant to employ this framework to see if any program has CT component or not.

Conclusions and Implication

In conclusion, first, CT in science education has 9 different components including generalization instead of parallelization. The first three CT components are data related one, which students were familiar with and they employed these three CT components while forming scientific concepts. On the other hand, the rest of CT components were used when students faced problem, built certain model by considering all possible factors influential to the solution. There was more frequently used in all components except generalization (which was added at the end) as levels went higher, which means that CT components in a spectrum can appear in order, which means CT components work as sequential competencies. Simulation, for example, can come after automation, it never comes before automation.

Second, CT components were differently used at different levels. Therefore, it is critical for students to learn required CT practices at expected levels from lower levels. Without learning CT practices at elementary level, students cannot complete all required CT practices as levels go higher. In addition, students experienced full CT components when they were exposed to contexts through which they faced problem, they decomposed problems into be manageable ones to be researched, and they built the best model by considering all possible factors to the solution. Therefore, it is very critical for science program (STEAM/STEM program since those programs are pursuing to find the practical solution) to be developed with the use of CT components to meet the goal of equipping students to be creative problem solvers with scientific literacy for the 4th industrial revolution.

Third, two steps of using CT components appeared in STEAM program. The first step of using CT is one of forming science concepts with curriculum-based through mainly three CT components (data collection data analysis, data representation). The second step of using CT is one of applying science concepts with curriculum-revised through the rest of CT components (decomposing, abstraction, algorithm and procedure, automation, simulation, and generalization). Therefore, it must be pivotal to include global issues (SSI; social scientific issue/climate change and water shortage in this study) for STEAM program so that students could experience all possible CT components to be creative problem solvers.

Fourth, CT practice is cognitive practice rather than physical practices with the use of computer as Wing (2008) indicated. Students could have chances to collect the data, analyze them, and represent them at the beginning of STEAM program with the use of concrete equipment and computer. Computer has been used during all program but it helped for students to carry out the project in ease and in detail. Students could have chance to build city with waterworks and water purification by scenario, which made them experience the latter parts of CT components cognitively without real construction. At least, students must be trained to find out the solution practical to the real life though they could not build the city as architects. Therefore, the CT components can be classified into two categories; direct CT practice with cognitive and physical practices (which means students use computer and other equipment) and indirect CT practice with cognitive practice only (which means students could have chance only to provide the product cognitively through drawings, and other representations). In both cases, students can have chances to be creative problem solvers. It is suggested to provide real context where students could try to develop concrete products.

Finally, more research about CTPF application to real teaching and to STEAM program is encouraged for teachers' expertise through professional development program. In NGSS, computational thinking as one of important 8 practices is emphasized to be included into science program so that students have opportunities to experience those practices to be creative problem solvers. This research can give any hint or guideline to people who want to develop/teach STEAM program with computational thinking practices.

Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (2016R1A2B4013063). The only data in this study are on the basis of the 2nd author's master thesis in Chosun University.

References

Barr, V. and 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 p.

Computer Science Teacher Association, 2011, Computational Thinking Leadership Toolkit.

Ministry of Education, 2015, 2015 revised information curriculum

National Research Council, 2012, A framework for K-12 Science education, Washington. D.C: The National Academies Press.

NGSS Lead States, 2013, Next Generation Science Standards: For states, by states. Washington, DC: The National Academy Press.

Park, Y-S., 2010, Secondary beginning teachers' views of scientific inquiry: With the view of hands-on, minds-on, and hearts-on. Journal of the Korean Earth Science Society, 31(7), 798-812 p.

Park, Y-S. and Hwang, J., 2017, The preliminary study of developing computational thinking practice analysis tool and its implementation. Journal of Korean Society of Earth Science Education, 19(2), 140-160 p.

Park, Y-S., May, 2018, The study of elementary science program with computational thinking practices and its understandings by elementary teachers, The 1st Korean Geoscience Union conference 2018, Hong-Cheon, Kangwon-Do, Korea.

Park, M., Park, G., Green, J., and Park, Y-S., September, 2017, Development of computational thinking practice tool and its application in science education, The biannual KESS conference 2017, G & R Hub, Chonnam National University, Korea.

Park, Y-S. and Park, M., July, 2017, Exploring the characteristics of computational thinking included in STEAM program, The 72nd KASE conference 2017, Gyungsang National University, Tong-Young campus, Tong-Young, Korea.

Park, Y-S. and Lawrence, B. F., October, 2015, The development of computational thinking practice observational protocol and its application, The 8th International EASE (East-Asian Association for Science Education) Conference 2015, Beijing Normal University, Beijing, China Mainland.

Song, J. and Na, J., 2014, The meaning of science fusion of window and direction of science classroom culture. Journal of Research in Curriculum Instruction, 18(3), 827-845 p.

Stohlmann, M., Moore, T. J., and Roehrig, G. H., 2012, Considerations for teaching integrated STEM education. Journal of Pre-College Engineering Education Research, 2(1), 28-34 p.

The International Society for Technology in Education, 2011, Computational Thinking teacher resources second edition.

The International Society for Technology in Education & Computer Science Teacher Association, 2011, Computational Thinking leadership toolkit.

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., and Wilensky, U., 2016, Defining computational thinking for mathematics and science classrooms. Journal of Science Education and Technology, 21(1), 127-147 p.

Wing, J. M., 2006, Computational thinking, Communication of the ACM, 49(3), 33-35 p.

Wing, J.M., 2008, Computational thinking and thinking about computing. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 366(1881), 3717-3725 p.

Manuscript received: July 16, 2018
Revised manuscript received: August 24, 2018
Manuscript accepted: August 27, 2018