

A Study on the Development and Implementation of Computational Thinking Education Framework

Hyun-Jong Choe *, Tae-Wuk Lee **

Abstract

In this paper, we propose the computational thinking education framework which provides three steps of computational thinking process and three kind of activities about computational thinking learning in class. The key idea of this framework is to provide the guidelines of designing activity steps of teaching and learning computational thinking in class using three axes of framework such as problem area, process of learning, and steps of computational thinking process. After designing a framework, we show that sample course of programming education program containing contents of Informatics subject in middle school by implementing our framework. Proposed framework and programming education program in middle school will be the good case study and guide to implement computational thinking concerned education programs in elementary, secondary, and universities.

▶ Keyword : Computational thinking, Framework for CT education, Informatics subject

I. Introduction

박 대통령이 2014년 7월 23일 'SW 중심사회 실현 전략보고회'에서 언급한 초·중등 교육에서의 SW 교육 필수화는 우리나라 중등학교 정보(Informatics) 교육에 큰 획을 그었다. 이로 인해 교육부는 2015 개정 교육과정에서 중학교 정보를 필수 교과로 지정하였고, 2018년부터 모든 중학생들이 정보 교육을 필수로 이수하게 되었다[1].

정보 과목이 필수 교과로 지정되어 모든 중학생에게 교육되어지게 된 가장 큰 이론적 배경은 바로 컴퓨팅 사고력 (computational thinking) 교육에 있다고 할 수 있다[2]. 컴퓨팅 사고력은 정보 교과목의 목표에 진술되어 있는 인지적 (cognitive) 영역의 교과 목표로 컴퓨터과학의 기본 개념과 원리 및 컴퓨팅 시스템을 활용하여 실생활과 다양한 학문 분야의 문제를 이해하고 창의적으로 해법을 구현하여 적용할 수 있는 능력을 의미한다[3]. 컴퓨팅 사고력은 미래를 살아가야 하는 학생들이 반드시 학습해야 하는 기본 지식의 하나로, 미국의 Wing 박사의 논문으로 인해 한국을 비롯한 미국, 영국 등 많은 선진국에서 컴퓨터과학 교육의 모토로 삼고 있는 핵심 개념이다[4].

컴퓨팅 사고력은 초등학교부터 대학까지의 컴퓨터과학 교육의 기저가 되고 있는 개념이지만, 인지적 영역의 목표이기 때문에 그 과정과 결과를 쉽게 파악하기 어려운 단점이 있다. 이에 컴퓨터과학 교육 현장에서는 이에 관련된 다양한 사례 연구가 이루어지고 있다. 즉, 컴퓨팅 사고력이 무엇이며 어떤 구성 요소로 이루어졌는지, 또한 각 구성 요소를 학습하기 위해서는 어떤 활동이 필요한지에 대한 것이 최근의 연구 주제들이다. 이에 본 연구에서는 컴퓨팅 사고력 교육을 위해 필요한 기본 가정이라 할 수 있는 컴퓨팅 사고의 과정과 컴퓨팅 사고력 교육의 활동 과정을 탐구해 보고, 이를 활용한 컴퓨팅 사고력 교육 프레임워크를 제안해 보고자 한다.

본 연구의 2장에서는 2015 개정 교육과정에서 제시된 중학교 정보 과목의 내용 체제와 컴퓨팅 사고력 교육, 그리고 본 논문과 관련된 배경 연구들을 살펴보았다. 중학교 정보 과목의 내용 체제를 살펴본 이유는 본 연구에서 제안한 컴퓨팅 사고력 교육 프레임워크를 사용하여 중학교 정보 과목에 포함된 프로그래밍 단원의 학습 내용 설계에 적용해 보기 위함이다. 3장은 컴퓨팅 사고력 교육 프레임워크를 제안하고, 이를 중학교 정보 과목의 프로그래밍 단원에 적용해 보았다.

• First Author: Hyun-Jong Choe, Corresponding Author: Tae-Wuk Lee

*Hyun-Jong Choe (blueland@sewon.ac.kr), Dept. of Computer Education, Seowon University

**Tae-Wuk Lee (twlee@knu.ac.kr), Department of Computer Education, Korea National University of Education

• Received: 2016. 09. 08, Revised: 2016. 09. 18, Accepted: 2016. 09. 28.

II. Preliminaries

1. 2015 Revised Curriculum

2015 개정 교육과정은 2015년 9월에 창의·융합형 인재 양성을 목표로 고시되었다. 정보 교과는 이 교육과정을 통해 보통 교과로 전환하게 되는데, 바로 중학교의 필수 교과로 지정된 것이다. SW 교육의 일환으로 초·중학교는 필수 교육 내용이 지정되었고, 고등학교의 정보는 일반 선택 과목으로 전환되어 좀 더 많은 학교들이 선택할 수 있도록 하였다.

중학교 정보 교과의 교육과정은 2007 개정 교육과정부터 강조된 컴퓨터과학 교육, 정보윤리 교육, 컴퓨팅 사고력 교육의 맥을 잇고 있다. 2015 개정 교육과정에 제시된 중학교 정보 교육의 목표는 기초적인 정보윤리의식과 정보보호능력을 함양하고 실생활의 문제 해결을 위해 정보기술활용능력과 컴퓨팅 사고력, 협력적 문제해결력을 기르는 데 중점을 두고 있으며[5], 2015 개정 교육과정에 제시된 중학교 정보 과목의 목표를 구현하기 위한 핵심 개념과 내용체계는 <표 1>과 같다.

Table 1. Informatics subject's contents in middle school

DOMAIN	CONCEPT	CONTENTS
Information Cultures	Information Society	·characteristics of information society and careers
	Information Ethics	·protection of personal information and rights ·cyber ethics
Data and Information	Representation of Data and Information	·types and digital representation of data
	Analysis of Data and Information	·collection of data ·structure of data
Problem-solving and Programming	Abstraction	·understanding of problem ·extracting of core element
	Algorithms	·understanding of algorithms ·representation of algorithms
	Programming	·input and output ·variables and operator ·control structures ·application of programming
Computing System	Operational Principles of Computing system	·structures and operational principles of computing system
	Physical Computing	·implementation of sensor-based program

2015 개정 교육과정은 또한 각 교과의 핵심 역량을 선별하여, 각 교과 교육을 통해 육성해야 하는 미래 인재의 역량을 진술하고 있다. 정보 교과는 정보문화소양, 컴퓨팅사고력, 협력적 문제해결력을 제시하고 있는데, 정보문화소양은 정보윤리의식, 정보보호능력, 정보기술활용능력을 포함하고 있다. 컴퓨팅 사

고력은 추상화능력, 자동화능력, 창의·융합능력을 포함하고 있고, 협력적 문제해결력은 협력적 컴퓨팅사고력, 디지털 의사소통능력, 공유와 협업능력을 포함하고 있다.

2015 개정 교육과정의 정보 교과는 교과의 목표와 내용, 핵심 역량에 컴퓨팅 사고력을 모두 포함시켜 학생들이 반드시 갖추어야 하는 미래의 역량으로 강조하고 있다. 교과의 내용 체계에서 특히 문제해결과 프로그래밍 단원은 정보 교과에서 가장 많은 개념과 내용을 담고 있는 단원으로 컴퓨팅 사고력 교육을 가장 잘 표현해 주고 있다.

2. Computational thinking process and UMC model

컴퓨팅 사고력은 2009년에 고시된 중학교 정보 교과의 목표에 포함되어 진술되면서 중등학교 정보 교육의 중요한 패러다임으로 자리 잡았다. 이 개념은 로고(LOGO)라는 교육용 프로그래밍 언어로 유치원생들의 인지적 사고 교육의 효과를 입증한 Papert 교수가 1996년에 처음 제시하였고[6], 그 이후에 Jeannette Wing 박사가 2006년에 ACM지에 발표한 'Computational Thinking'이란 논문을 통해 컴퓨터과학 교육업계에 커다란 반향을 불러일으켰다[7].

컴퓨팅 사고력은 인지적 사고의 과정이기 때문에 한 마디로 정의하기는 어렵지만, 컴퓨터 과학에 사용되는 기술을 이용한 문제 해결, 컴퓨터를 통해 문제를 해결해 나가는 과정, 컴퓨터 과학 기술을 사용한 문제해결방법 등으로 정의되고 있다[8]. 즉, 컴퓨터와 컴퓨터과학 지식을 활용하여 일상생활 문제를 해결할 수 있는 능력으로 정의할 수 있다.

컴퓨팅 사고의 과정은 2006년에 Wing 박사가 제안한 추상화(abstraction)와 자동화(automation)의 단계가 있다. 추상화는 문제해결을 위해 문제를 이해하여 분석하거나 중요한 부분을 추출하는 활동 등을 통해 해결해야하는 문제의 복잡성을 효과적으로 단순화해 나가는 과정을 의미하고, 자동화는 추상화된 개념이나 절차, 방법 등을 컴퓨터가 실행할 수 있도록 알고리즘화하는 과정을 의미한다[9]. 하지만, 최근 컴퓨터과학 교육학계에서는 <그림 1>과 같이, 두 단계에 분석(analysis) 단계를 추가하여 추상화, 자동화, 분석을 컴퓨팅 사고의 과정으로 제시하고 있다[10].

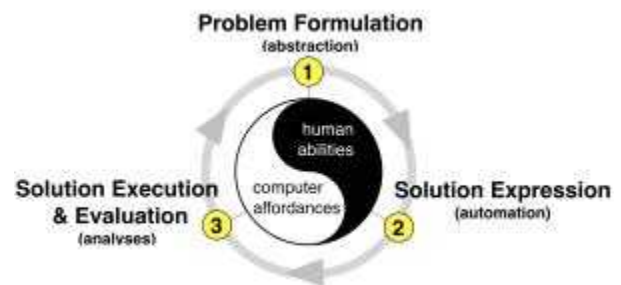


Fig. 1. Computational thinking process

프로그래밍의 예를 들어 살펴보면, 문제를 이해하고 분석하는 추상화 단계를 거쳐 알고리즘을 표현하고, 이를 프로그래밍 언어로 표현하는 과정이 자동화 단계이다. 지금까지는 이 2개의 과정이 곧 컴퓨팅 사고 과정이라고 하였다. 하지만 실제 프로그램의 개발 절차나 프로그래밍 학습에서는 프로그래밍의 결과를 실현해 보고, 그에 따라 문제의 분석 과정이나 알고리즘 표현 과정을 평가하는 과정이 반드시 뒤따르게 된다. 따라서 추상화와 자동화 과정 이후에 해결 방법을 실행해 보고, 평가해 보는 분석 과정도 컴퓨팅 사고의 과정에 포함되어 진술되어야 한다는 것이다. 이에 본 연구에서는 컴퓨팅 사고의 과정을 추상화, 자동화, 분석의 단계로 제안하여 컴퓨팅 사고력 교육의 프레임워크를 설계하고자 한다.

Irene Lee 외(2011)는 컴퓨팅 사고 교육을 학교에서 실행하는 과정에서 학습자가 경험하는 단계로 <그림2>와 같이 사용(USE), 수정(MODIFY), 창작(CREATE)을 UMC 모델로 제시하여 설명하고 있다[11].

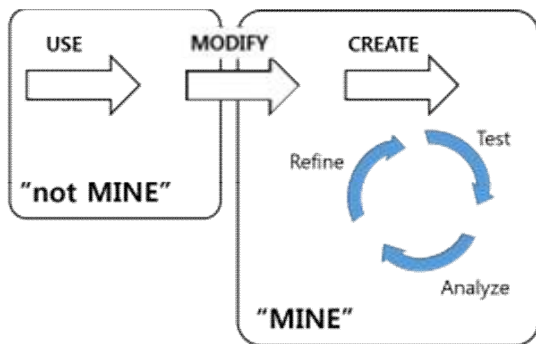


Fig. 2. Use-Modify-Create process

프로그래밍 학습의 예를 들면 첫 번째로 사용 단계는 이미 만들어진 프로그램을 사용하여 경험해 보는 것이다. 두 번째 단계는 수정 단계인데, 경험해 본 프로그램을 일부 수정해 보는 것이다. 입·출력되는 자료의 값을 수정해 보거나, 프로그램의 로직을 수정해 보는 단계이다. 프로그램을 수정하기 위해서는 프로그램의 코드와 로직에 대한 이해가 선행되어야 하는데, 이 과정이 컴퓨팅 사고의 추상화와 자동화가 약한 수준에서 이루어지게 되는 것이다. 다양한 형태의 수정 단계가 지속되면 학습자는 스스로 프로그래밍 언어의 표현과 프로그램에 대한 자신감이 증가하게 되는데, 이는 창작 단계로 넘어가기 위한 전단계가 된다. 세 번째 단계는 창작 단계이다. 이 단계는 학습자 자신 스스로 추상화, 자동화, 분석의 단계를 통해 자신만의 프로그램을 개발하는 단계이다. 사용과 수정 단계를 넘어 창작 단계가 되어야만 학습자는 스스로 자신만의 프로그램이 되는 것이다. UMC 모델은 미국의 National Science Foundation에서 컴퓨팅 사고 교육을 위한 연구과제에 속한 초·중등학교에서 로봇, 프로그래밍 및 STEAM 교육에 적용되었다.

3. Related works

컴퓨팅 사고력 교육에 관한 논문은 최근 SW 교육과 정보 교과의 필수화로 인해 다양한 유형이 발표되고 있다. 본 연구의 주제와 관련하여 살펴보면, 교수·학습 모델을 제시하는 경우와 다양한 교과에서 컴퓨팅 사고 교육의 사례를 제시하는 경우로 구분하여 살펴볼 수 있다.

먼저 컴퓨팅 사고력 향상을 위한 모델 연구로는 성영훈(2016)의 CT-SPI 모델이 대표적인데, 이 모델은 초등학생의 아두이노를 활용한 컴퓨팅 사고 교육을 위한 것이다[12]. CT-SPI 모델은 시스템사고(System Thinking), 프로토타입핑(Prototyping), 인터랙션(Interaction)으로 구성된 모델로 피지컬 컴퓨팅 교육에 특화된 모델이라고 할 수 있다. 또한 전영주와 김태영(2015)은 컴퓨팅 사고 교육을 위한 교수학습 과정을 제안한 CT-CPS 프레임워크를 제안하였다[13]. CT-CPS 프레임워크는 창의적 문제해결을 위해 문제 확인과 분석, 아이디어 생각, 설계, 적용과 평가의 학습 단계를 제안하고 있는데, 이를 초등학생들의 스크래치 교육에 적용하여 긍정적 효과를 발견하였다.

컴퓨팅 사고력 교육의 사례에 관한 연구로는 컴퓨터 관련 교과와 비 관련 교과로 구분지을 수 있는데, 컴퓨터 비 관련 교과의 연구로는 전성균과 이영준(2014)이 교육용 프로그래밍 언어로 예술 중심의 STEAM 교육 프로그램을 제안하였는데, 팝아트 작품과 우연성 음악을 창작하는 수업을 적용한 사례연구이다[14]. 이해영과 이태욱(2016)은 컴퓨팅 사고력 신장을 위해 중학교 과학 교과에 프로그래밍 학습을 도입한 STEAM 교육 프로그램을 제안하기도 했고[15], 김영중과 이태욱(2016)은 초등학교 실과 과목의 생활과 전기전자 단원에 적용할 수 있는 융합 교육 프로그램을 제안하기도 하였다[16].

컴퓨팅 사고력 교육과 직접적 관련이 가장 깊은 학습 요소인 프로그래밍 관련 연구도 비교적 많은 연구들이 이루어지고 있다. 김경규와 이종연(2016)은 중학교에 적용할 수 있는 컴퓨팅 사고력 중심의 프로그래밍 프로그램을 개발하여 적용하였다[17]. 또한 신수범(2015)의 경우에는 초등학교 학생들을 대상으로 스크래치 소프트웨어 교육을 적용하여 의미있는 컴퓨팅 사고력 향상을 보이기도 하였다[18]. 유인환(2014)은 로봇과 앱 인벤터의 연동을 통해 SW 교육 방안을 제시하였는데, 초등학생의 컴퓨팅 사고력 교육 방안으로 동기 부여와 흥미 측면에서 효과적인 수단이 된다는 것을 강조하였다[19]. 로봇에 관한 교육 프로그램으로 심규현 외(2014)는 아두이노를 활용한 교육 프로그램을 개발하여 초등학생을 대상으로 적용한 결과, 흥미도와 관심도가 유의미하게 증가하였음을 제시하고 있다[20].

컴퓨팅 사고력 교육의 다양한 사례 연구는 많이 이루어지고 있으나, 컴퓨팅 사고력 교육의 기본 이론이 될 수 있는 체제와 프레임워크에 관한 연구는 아직 미비한 상황이다. 이에 본 연구는 프로그래밍, 로봇, STEAM 등 다양한 학문 분야에 적용할 수 있는 일반적인 컴퓨팅 사고력 교육 프레임워크를 제안하고자 한다.

III. Framework Design and its Adaption

1. Computational thinking education framework

컴퓨팅 사고력은 추상화, 자동화, 분석의 과정을 통해 이루어지고, 이를 위한 학습 과정은 사용, 수정, 창작의 단계를 거친다는 것을 선행 연구를 통해 확인할 수 있었다. 컴퓨팅 사고는 프로그래밍만을 대상하는 것이 아니라, 일상 생활에서 컴퓨터를 활용하는 다양한 상황에 적용될 수 있는 인지적 과정이다. 따라서 컴퓨팅 사고력 교육은 컴퓨터의 프로그래밍, 로봇, STEAM 등을 포함한 다양한 학습 과제나 연구 분야에 활용될 수 있을 것이다. 이에 본 연구에서는 컴퓨팅 사고의 과정, 컴퓨팅 사고력 학습의 유형, 적용 가능한 문제 분야라는 세 가지 축을 기준으로 하는 컴퓨팅 사고력 교육 프레임워크를 <그림 3>과 같이 제안한다.

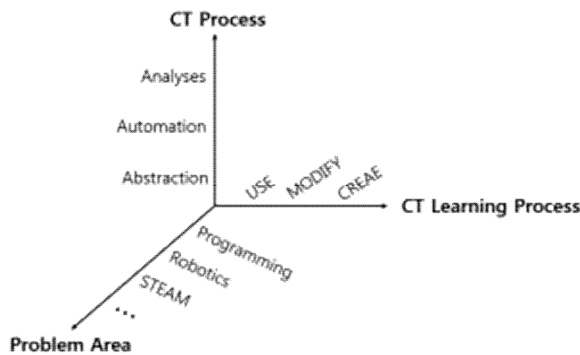


Fig. 3. Computational thinking framework

본 연구에서 제안한 프레임워크는 컴퓨팅 사고력 교육이 필요한 다양한 학문 분야에서 적용할 수 있는 모델로, 컴퓨팅 사고력의 단계에 기반하여 필요한 학습 단계를 함께 제시하고 있다는 특징을 가지고 있다. 예를 들어, 이 프레임워크를 적용한 로봇 교육 프로그램을 설계하면 <표 2>와 같다.

Table 2. Sample of robotics program

	USE	MODIFY	CREATE
Abstraction	-	Finding condition to change direction of robot	Finding condition of robot vacuum
Automation	-	Representing condition to change direction of robot	Representing algorithms of robot vacuum
Analyses	Executing a sensor based robot	Verifying direction of sensor based robot	Verifying execution of robot vacuum

사용 단계에서는 단순히 만들어져 있는 로봇을 실행시키는 수준이기 때문에 컴퓨팅 사고력의 분석 과정이 적용된 것이고,

수정과 창작 단계는 컴퓨팅 사고력의 전 단계가 사용된다.

2. Design of programming program based on the framework

본 연구에서는 컴퓨팅 사고력 교육 프레임워크의 적용성과 일반성을 확인하기 위해 2015 개정 교육과정의 중학교 정보 과목의 프로그래밍 단원 내용을 예제로 설계하여 제시하고자 한다. 중학교 정보 과목의 프로그래밍 단원은 입력과 출력, 변수와 연산, 제어 구조, 프로그래밍 응용으로 구성되어져 있다. 4개의 학습 내용 요소의 수업을 위한 학습 내용 설계는 <표 3>과 같다. 4개의 학습 내용 요소에는 컴퓨팅 사고의 단계인 추상화(Ab), 자동화(Au), 분석(An)의 단계에 따라 사용, 수정, 창작 활동을 구상하였다.

Table 3. Adaptation of framework in programming of Informatics subject

		USE	MODIFY	CREATE
Input & Output	Ab	-	Finding condition of printing name and Hello	Finding condition of printing input characters
	Au	-	Representing algorithms of printing name and Hello	Representing algorithms of printing input characters
	An	Executing Hello program	Verifying of printing name and Hello	Verifying to print input characters
Variables & operators	Ab	-	Finding condition of minus program	Finding condition of calculation
	Au	-	Representing algorithms of minus program	Representing algorithms of calculation program
	An	Executing add program	Verifying of minus program	Verifying calculation program
Control structures	Ab	-	Finding condition of 5 times table	Finding condition of times table
	Au	-	Representing algorithms of 5 times table	Representing algorithms of times table
	An	Executing 2 times table program	Verifying 5 times table program	Verifying times table program
Programming	Ab	-	-	Finding condition of vending machine
	Au	-	-	Representing algorithms of vending machine
	An	-	-	Verifying vending machine program

UMC 모델의 사용 단계에서는 추상화와 자동화의 사고력이 사용되지 않기 때문에 분석 단계만 기술하였고, 프로그래밍 단원의 마지막 학습 요소인 프로그래밍 응용은 학습자가 스스로 프로그램을 설계하여 작성하는 요소이기 때문에, 사용과 수정 단계는 생략하고, 창작 단계만 작성하였다.

본 연구에서 제안한 컴퓨팅 사고력 교육 프레임워크는 컴퓨팅 사고력의 과정인 추상화, 자동화, 분석 단계와 컴퓨팅 사고력 학습 과정인 사용, 수정, 창작의 관점에서 컴퓨팅 사고력 학습 활동을 설계하고, 평가할 수 있다는 장점이 있다. 예제로 작성된 <표 3>을 보면 컴퓨팅 사고력 교육을 위한 활동을 컴퓨팅 사고의 관점과 학생들의 활동의 관점이라는 2가지 측면에서 설계하거나 평가할 수 있다. 즉 컴퓨팅 사고력 교육을 담당하는 교사나 교수들은 자신이 설계, 수행한 교육 활동을 평가할 수 있는 가이드라인으로 본 연구에서 제안한 프레임워크를 활용할 수 있을 것이다.

IV. Conclusions

본 연구는 최근 SW 교육의 핵심 패러다임으로 자리 잡고 있는 컴퓨팅 사고력 교육 프로그램 설계를 위한 프레임워크를 제안하였다. 교육에 관한 프레임워크는 관련된 다양한 분야의 교육 프로그램 개발을 위한 가이드라인과 참조 모델이 될 수 있어야 하기 때문에, 일반화가 가능한 컴퓨팅 사고력 과정과 컴퓨팅 사고력 교수·학습 단계를 참고하였다. 그리고, 개발된 프레임워크의 현장 적합성을 살펴보기 위해 2015 개정 교육과정에 제시된 중학교 정보 교과 프로그래밍 단원의 내용 요소를 교육하기 위한 학습 활동을 설계해 보았다. 설계된 학습 활동 프로그램은 학습자의 컴퓨팅 사고 과정인 추상화, 자동화, 분석 단계와 컴퓨팅 사고력 학습 과정인 사용, 수정, 창작의 단계를 순차적으로 학습할 수 있는 것으로 판단되었다.

컴퓨팅 사고의 과정과 컴퓨팅 사고력 관련 학습 활동인 UMC 모델을 별도로 생각하는 것보다 두 개의 이론을 하나의 프레임워크에 적용한 결과, 실제 교수·학습 활동이 인지 과정과 학습 활동의 순서 및 단계에 맞게 설계되었다. 특히, 학습자의 컴퓨팅 사고력 수준이 낮거나 학습자 연령이 적은 경우에는 사용과 수정 활동 위주로 교육 프로그램을 설계하고, 학습자의 수준이 높은 경우에는 수정과 창작 활동 위주로 교육 프로그램을 설계할 수도 있다. 이는 교육 프레임워크가 갖는 유연성으로 학습자의 수준과 문제 분야에 맞게 적절하게 적용할 수 있는 장점으로 해석된다.

향후, 본 연구는 컴퓨팅 사고력 교육 프레임워크를 좀 더 다양한 분야로 확대하여 관련 교육 프로그램을 개발하고 실행하여, 연구 결과의 일반화 가능성을 높일 필요가 있다. 이를 토대로 초·중등 및 대학에서 컴퓨팅 사고력 교육이 학습자의 수준과 학문 분야에 맞게 컴퓨팅 사고력 과정과 활동이 이루어졌으면

하는 바램이다. 추가 연구를 통해 모처럼 맞이한 SW 교육 패러다임이 교육계와 학계, 산업계에 긍정적인 효과를 발휘하기를 바란다.

REFERENCES

- [1] eNewsToday, 2015 revised national curriculum, <http://www.ewestoday.co.kr/news/articleView.html?dxno=422766>
- [2] TaeWuk Lee and HyunJong Choe, "Informatics Subject Education, Revision", pp. 192-193, 2016.
- [3] Ministry of Education, 2015 revised national curriculum of Informatics.
- [4] JeongWon Choi, et al., "Computing Education in Korea-Current issues and endeavors", ACM Transaction On Computing Education, Vol. 15, No. 2, pp. 8:1-8:22 Apr. 2015.
- [5] National Curriculum Information Center, Informatics subject curriculum. <http://ncic.go.kr>
- [6] Seymour Papert, "An Exploration in the Space of Mathematics Education", International Journal of Computers for Mathematical Learning, Vol. 1, No. 1, pp. 95-123, 1996.
- [7] Jeannette Wing, "Computational Thinking", Communcation of the ACM, Vol. 19, No. 3, pp. 33-35, 2006.
- [8] TaeWuk Lee and HyunJong Choe, "Informatics Subject Education, Revision", p. 194, 2016.
- [9] Sookyoung Choi, "An Analysis of "Informatics" Curriculum from the Perspective of 21st Century Skills and Computational Thinking", The Journal of Korean association of computer education, Vol. 15, No. 6, pp. 19-30, Dec. 2011.
- [10] Computational Thinking, https://en.wikipedia.org/wiki/Computational_thinking
- [11] Irene Lee, et al., "Computational Thinking for Youth in Practice", ACM Inroads, Vol. 2, No. 1, pp. 32-37, Mar. 2011.
- [12] YoungHoon, Sung, "Development and Application of CT-SPI Model for Improving Computational Thinking for Elementary School Students", Journal of The Korea Society of Computer and Information, Vol. 21, No. 1, pp. 169-180, Jan. 2016.
- [13] YongJu Jeon and Taeyoung Kim, "The Development of

the CT-CPS Framework for Creative and Integrative Software Education”, Korean Journal of Teacher Education, Vol. 31, No. 3, Sep. 2015.

- [14] SeungKyun Jeon and YoungJun Lee, “Art based STEAM Education Program using EPL”, Journal of The Korea Society of Computer and Information, Vol. 19, No. 4, pp. 149-158, Apr. 2014.
- [15] HyeYoung Lee and TaeWuk Lee, “Development of Science Subject Program based on Programming Learning to Improve Computational Thinking Ability in middle school”, Journal of The Korea Society of Computer and Information, Vol. 20, No. 12, pp. 181-188, Dec. 2015.
- [16] Myungjung Kim and Taewuk Lee, “Development of Convergence Education Program of the Life and Electricity/Electron Unit in Practical Arts Textbook to Enhance Computational Thinking”, Journal of The Korea Society of Computer and Information, Vol. 21, No. 1, pp. 199-205, Jan. 2016.
- [17] KyungKyu Kim and JongYun Lee, “Analysis of the Effectiveness of Computational Thinking-Based Programming Learning”, The Journal of Korean association of computer education, Vol. 19, No. 1, pp. 27-39, Mar. 2016.
- [18] SooBum Shin, “The Improvement Effectiveness of Computational Thinking through Scratch Education”, Journal of The Korea Society of Computer and Information, Vol. 20, No. 11, pp. 191-197, Nov. 2015.
- [19] InHwan Yoo, “Design a Programming Education Plan for SW Education Using Robot and Mobile Application Development Tool”, The Journal of Korean Association of Information Education, Vol. 18, No. 4, pp. 615-624, Dec. 2014.
- [20] KyuHeon Shim, et al., “Development and Evaluation of a STEAM Curriculum Utilizing Arduino”, The Journal of Korean association of computer education, Vol. 17, No. 4, pp. 23-32, Dec. 2014.

Authors



Hyun Jong Choe received the B.S. degree in Elementary Education from Kongju National University of Education in 1993 and M.S. and Ph.D. degrees in Computer Education from Korea National University of Education in 2001 and 2005, respectively. Dr. Choe is currently Professor of the Department of Computer Education at Seowon University since 2006. He is interested in Informatics Education.



Tae Wuk Lee received the B.S. degree in Science Education from Seoul National University in 1978 and M.S. and Ph.D. degrees in Computer Science and Computer Education from Florida Institute of Technology, U.S.A. in 1982 and 1985, respectively. Dr. Lee is currently the Professor of the Department of Computer Education at Korea National University of Education, Korea since 1985. He is interested in Computer Science Education and Knowledge Engineering.