

# 21<sup>st</sup> Century Skills와 Computational Thinking 관점에서의 '정보' 교육과정 분석

최숙영<sup>†</sup>

## 요 약

본 연구에서는 21<sup>st</sup> Century Skills와 Computational thinking의 관점에서 컴퓨터 교육의 특성을 분석하고자 하였다. 21<sup>st</sup> Century Skills는 현세계에서 성공적인 삶을 살아가기 위한 필수 능력으로 학습과 관련하여 비판적 사고, 문제 해결 능력, 통신 능력과 협동 능력이 포함되어 있다. Computational thinking은 오늘날 융합 시대에 요구되는 기초 능력이라 할 수 있으며, 컴퓨터과학 교육의 핵심 개념이라 할 수 있다. 본 연구에서는 먼저, 21<sup>st</sup> Century Skills와 Computational thinking의 특징을 살펴보고, 이러한 것들이 현재 중·고등학교에서 컴퓨터교육을 위해 운영되고 있는 '정보' 과목의 교육과정과 어떻게 관련이 되는지를 분석하였다. 이를 통하여 초·중·고에서의 컴퓨터교육에 대한 중요성 및 필요성을 도출해냈다.

주제어 : 21<sup>st</sup> Century Skills, Computational thinking, 컴퓨터과학 교육

## An Analysis of 'Informatics' Curriculum from the Perspective of 21<sup>st</sup> Century Skills and Computational Thinking

Sook-Young Choi<sup>†</sup>

### ABSTRACT

This study analyzed characteristics of computer education from the perspective of 21<sup>st</sup> Century Skills and Computational thinking. 21<sup>st</sup> Century Skills are essential skills for success in today's world. They include critical thinking, problem solving, communication and collaboration. Computational thinking is a necessary ability in the age of convergence and a core concept of computer science education. This study first examined characteristics of 21<sup>st</sup> Century Skills and Computational thinking. Then, it analyzed the relationship between these two skills and 'Informatics' curriculum. 'Informatics' is an elective course in K-12. The results of this study emphasized the importance and the necessity of computer education in the K-12 level.

**Keyword** : 21<sup>st</sup> Century Skills, Computational thinking, Computer Education

---

<sup>†</sup> 종신회원: 우석대학교 컴퓨터교육과 교수(교신저자)

논문접수: 2011년 09월 15일, 심사완료: 2011년 10월 05일, 게재확정: 2011년 11월 04일

\* 본 논문은 2011년 우석대학교 학술연구비 지원에 의하여 수행되었음

## 1. 서론

지금까지 인류 역사를 볼 때 교육은 그것이 수용되는 사회의 사회적 필요를 반영하며 발전되어 왔다고 볼 수 있다. 그런데, 오늘날의 지식정보화 사회는 교육 환경 전체에 대한 변화를 요구하고 있다. 이러한 맥락 가운데, 교육 주체로서 학생과 교사도 그 변화에 있어서 예외일 수 없다. 교사는 기존의 단순한 단방향의 지식 전달자에서, 학생들로 하여금 주어진 문제를 자기 주도적으로 해결할 수 있는 능력을 키워주는 안내자와 협력학습자로서의 역할 변화를 요구받고 있다. 특히, 학습의 주체로서 주도적이고 창의적으로 문제를 해결할 수 있는 능력은 현재 사회에서 매우 요구되고 있다.

이러한 상황속에서 미국의 국가 단체인 Partnership for 21st Century Skills에서는 21세기를 살아갈 학생들이 그들의 생활과 직장에서 성공적인 삶을 살아가기 위해 배워야 할 생존기술로 비판적 사고, 문제 해결 능력, 통신 능력과 협동 능력을 갖추어야 한다고 발표하였다[1][2]. 이러한 기술들을 초·중고 과정에서 효과적으로 습득할 수 있다면 성공적인 삶을 살아가 수 있는 바탕이 될 수 있을 것이다.

한편, Computational thinking(Computational thinking)란 용어는 Jennette Wing(2006)에 의해 컴퓨터학계에 처음으로 소개되었는데, Wing은 이것이 21세기를 살아가는 모든 사람에게 기본적으로 필요한 기술이라고 주장하고 있다[3]. Wing은 읽기, 쓰기, 셈하기와 같은 기본 소양에 Computational thinking을 추가해야 한다고 주장하고 있다. 또한 Wing 교수에 의해 Computational thinking이 소개된 이래 이 용어에 대한 정의를 두고 많은 논의가 있어왔다. 최근에 Cuny, Snyder와 Wing(2010)은 최근 Computational thinking에 대해 문제들이 정보처리 에이전트에 의해 효과적으로 수행될 수 있는 형태로 표현될 수 있도록 문제와 그 해결책들을 형식화하는 것에 관련된 하나의 사고 과정이라고 정의하고 있다[4]. Computational thinking을 컴퓨터과학 분야와 그리고 그 분야를 연결해주는 하나의 기본적인 학문 분야로서 보는 관점과 현재의 기술사회에서 모든 사람들에게 기본적으로 필요한 컴퓨터과학의 집합

(set)을 획득하는 하나의 시도로 해석하기도 한다 [5].

Computational thinking은 컴퓨터과학의 중심 개념이라 할 수 있으며, 최근 미국과 영국을 중심으로 Computational thinking의 중요성을 인식하고 그에 대한 교육을 정규 교육과정에 포함시키기 위한 연구들이 수행되고 있다.

본 연구에서는 21세기를 살아가는 현대인으로서 기본적으로 갖추어야 할 21st Century Skills와 Computational thinking의 특징을 살펴보고, 이것들과 컴퓨터 교육과의 관련성을 분석하고자 한다. 즉, 중·고등학교에서 컴퓨터교육을 위해 운영되고 있는 '정보' 과목의 교육과정이 21st Century Skills 및 Computational thinking과 어떻게 관련되는지를 분석하고 컴퓨터 교육의 필요성을 도출하고자 한다.

## 2. 관련연구

### 2.1 컴퓨터 교과

#### 2.1.1 컴퓨터 교과에 대한 교육과정

컴퓨터 교과에 대한 교육과정은 2007 개정 교육과정에서 큰 변화를 겪었다. 제 6차와 제 7차 컴퓨터 교육과정의 경우 일상생활에서 당면하는 문제 해결에 컴퓨터를 주요 도구로 활용할 수 있는 능력을 기르도록 하였다. 따라서 컴퓨터 교과는 단순히 응용 소프트웨어를 잘 활용하는 것이 교과 성격이자 목표가 되었고, 이로 인해 긍정적 측면보다는 많은 부정적 측면이 대두되었다[6].

2007 개정 교육과정은 컴퓨터 관련 과목의 명칭을 '정보'로 통일하고, 그 동안 잘못 인식되었던 교과의 정체성을 새롭게 정립하는 계기를 마련하였다. 즉, 컴퓨터의 정보과학과 기술의 원리를 통해 지식·정보 사회를 올바르게 이해하고, 창의적 문제해결력 및 논리적 사고력을 신장시키기 위한 교과로 개정되었다[7][8]. 이를 위해 응용 소프트웨어의 단순 기능 교육 비중을 축소하고, 정보과학의 원리와 이해를 증진시킬 수 있는 내용, 문제해결 방법과 절차에 대한 내용, 교육용 프로그래밍 언어, 정보윤리에 대한 내용 등을 강화하였다. <표 1>은 2007 개정 '정보' 교육과정에 대한 성격, 목표, 영역별 세부 목표를 보여주고 있다.

<표 1> 2007 개정 '정보' 교육과정

| 항목              | 2007 개정 교육과정  |   |  |
|-----------------|---|---|--|
| 교과명             | 정보(중, 고등학교)   |   |  |
| 성격              | 1. '정보'는 정보 기술의 활용을 통해 미래 지향적 사고력, 논리적 사고력, 창의적 사고력, 의사 결정력 등을 함양시키는 과목이다.<br>2. '정보'는 정보의 기본적인 개념과 원리를 습득하고 정보 처리를 위한 기능을 익힘으로써 창의적이고 실질적인 문제해결 능력을 신장시키는 과목이다.<br>3. '정보'는 정보기기의 다양한 특성과 장점을 활용하여 정보를 효율적으로 처리할 수 있는 기본적인 능력을 기르는 과목이다.<br>4. '정보'는 정보사회의 일원으로 갖추어야 될 정보윤리, 정보 보호에 대한 내용을 이해하고 실천할 수 있는 태도를 고취시키는 과목이다. |   |  |
| 총괄 목표           | 정보 처리의 기본 원리와 올바른 정보 활용 지식을 습득하여 자신의 생각을 다양한 형태의 정보로 표현하고 실생활에서 일어나는 문제를 창의적이고 능동적인 방법으로 해결할 수 있는 능력과 태도를 기른다.  |   |  |
| 교과 영역 및 영역별 목표  | 영역  | 영역별 목표  |  |
|                 |   | 중학교   | 고등학교   |
|                 | 1. 정보기기 의 구성과 동작  | 컴퓨터의 구성과 동작 원리, 운영체제의 원리와 기능, 네트워크의 이해를 통해 정보기기를 직접 다루고 조작할 수 있다.   | 논리 연산과 논리 회로, 프로세스와 기억 장치, 네트워크와 보안 기술의 이해를 통해 정보기기의 구성과 동작 원리를 이해할 수 있다.  |
|                 | 2. 정보의 표현과 관리   | 자료구조 및 정보 표현의 원리를 이해하고 문제해결을 위해 다양한 방법으로 정보를 표현하고 구조화할 수 있다.  | 논리와 추론 방법 및 원리, 관계와 함수, 데이터베이스 관리 방법에 대해 이해하고 현실 세계의 정보를 컴퓨터가 처리할 수 있는 형태로 표현하고 구조화할 수 있으며 대량의 정보를 체계적으로 관리할 수 있다. |
|                 | 3. 문제해결 방법과 절차  | 실생활에서 발생하는 다양한 문제를 정보 처리의 관점에서 이해하고 정보 처리의 지식과 기능을 활용하여 창의적이고 능동적으로 문제를 해결할 수 있다.                                   | 실생활에서 발생하는 다양한 문제를 구조화하고 해결 전략을 비교 분석할 수 있다. 또한 구조적 프로그래밍과 객체 지향 프로그래밍을 활용하여 창의적이고 능동적으로 문제를 해결할 수 있다.             |
| 4. 정보사회 와 정보 기술 | 정보윤리 및 정보의 공유와 보호의 중요성을 인식하고 이를 준수하며, 원하는 정보를 수집하고 다양한 형태로 가공하여 다른 사람에게 전달할 수 있다.   | 올바른 정보 사회를 선도하기 위해 정보윤리 및 직업의 변화를 이해하고, 웹의 관리 및 운영, 다양한 웹 기술 활용 방법을 익혀 웹에서 건전하고 올바른 정보를 지속적으로 활용할 수 있는 능력과 태도를 기른다. |  |

2.1.2 컴퓨터 교과에 대한 기존 연구

컴퓨터 교과의 교육과정에 관해 그동안 여러 연구들이 수행되었다. 특히, 2007년 개정 교육과정이 나오기 전인 제 6차와 제7차 교육과정에서 컴퓨터 교과의 교육과정에 관한 문제점과 이를 개선하기 위한 다양한 방안을 제시하는 연구들이 있었다.

신수범, 이태욱(2005)는 컴퓨터 교과의 내용이 소프트웨어 활용 중심으로 되어 있는 것에 대한 문제점을 지적하고 있으며, 컴퓨터 교과의 교육내용을 학문적 가치를 확대하고 내재적 지식 구조를 분명히 하여 초·중등 교육 목표에 부합할 수 있도록 재구조화의 필요성을 제안하였다[9]. 김홍진(2005)은 7차 교육과정의 문제점에 대한 개선 방안으로 교과목별로 전문 담당관을 확충하여 각 교과목을 심도있게 연구할 수 있도록 체계 구축의 필요성을 제안하고 있다[10]. 박정호 외 2인(2006)은 중학교 컴퓨터 교육과정의 문제점을 체계성, 계열성 부족으로 지적하고 소프트웨어 기능 습득 위주의 교육과정에 대해 지적하고 있다. 기존의 교육

과정을 수정 및 보완하여 컴퓨터 원리, 알고리즘 및 프로그래밍, 정보 윤리 등의 영역을 강화해야 한다고 제안하고 있다[11].

위의 컴퓨터 교과의 교육과정에 대한 연구들은 소프트웨어 중심의 활용교육에서 벗어나 프로그래밍 중심의 컴퓨터과학으로서 학문 중심적 교육내용을 제안하고 있으며, 체계성 및 계열성을 고려한 교육과정의 개선 방안들을 제시하고 있다.

한편 컴퓨터교육의 본질과 필요성에 대해서 고려한 연구들로 이원규 외 6인(2007)에서는 컴퓨터 교육의 본질을 컴퓨팅의 개념과 정보 기술 유창성으로 설명하고 있다[12]. 이 연구에서는 정보의 개념을 컴퓨터를 이용하여 정보가 처리되는 절차적 과정에 대한 이해로 제시함으로써 보다 컴퓨터 과목의 본질적인 성격에 근접하고 있다는 점이 주목할 만하다. 이영준과 이은경(2008)은 컴퓨터교육의 정체성을 위협하는 가장 큰 오개념은 컴퓨터 과학이 컴퓨터에 관한 학문이라는 도구적 관점과 컴퓨터과학이 곧 프로그래밍이라는 관점에서 비롯된다고 지적하고 있다[13]. 컴퓨터교육에 관한 외부적

인식의 전환이 요구되지만, 이에 앞서 컴퓨터교육의 본질에 대한 심도 있는 고찰이 요구된다고 주장하고 있다. 컴퓨터교육이 기초 교과로서의 성격을 지니는 독립적인 과학의 한 영역으로 모든 사람이 필수적으로 배워야 함을 주장하고 있으며, 이를 위한 근거들을 제시하고 있다.

위의 기존의 컴퓨터교육에 관한 연구들을 살펴보면 크게 그동안 시행되어 왔던 컴퓨터교육과정에 관한 문제점 및 개선방향에 대해 제시한 연구들과 컴퓨터교육의 본질과 정체성에 대해 고찰한 연구들로 구분할 수 있다. 그러나, 이러한 연구들은 왜 컴퓨터교육이 오늘날의 지식정보화 사회를 살아가는 모든 사람들에게 필요한지에 대한 근본적인 고찰은 이루어지지 않았다. 또한, 21st Century Skills와 Computational thinking의 관점에서 2007년 개정 교육과정의 내용이 어떠한 특성과 의미를 지니고 있는지에 대한 구체적인 분석 등에 관한 연구는 그동안 이루어지지 않았다.

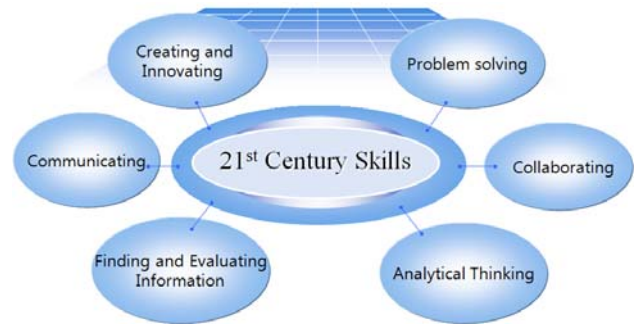
## 2.2 21<sup>st</sup> Century Skills

교육은 사회의 큰 변화에 따라 영향을 받아왔다고 볼 수 있다. 산업사회에서의 직업과 삶을 준비하기 위해 설계되었던 학습은 기술의 급속한 발전에 따른 오늘날 지식의 혁명의 시대에 맞게 다시 설계되어야 할 것이다.

한 사회의 도구들은 학습의 주체이자 수단이 되기 때문에 어떤 시대에 유행하고 있는 기술들은 항상 교육과 밀접한 관련이 있어왔다. 미디어, 텔레커뮤니케이션, 학습과학과 연결된 네트워크 기술들은 급속하게 현대 생활을 위해 필요한 정보들을 획득하기 위한 강력한 학습 시스템으로 진화되고 있다.

이러한 급격한 사회 전반적인 변화에 발 맞춰 교육이 이루어질 수 있도록 Partnership for 21st Century Skills에서는 21세기를 살아갈 학생들이 그들의 생활과 직장에서 성공적인 삶을 살아가기 위해 갖추어야 생존 기술, 지식과 전문성에 관한 것을 포함하는 하나의 프레임워크를 개발하였다[1][2].

프레임워크는 크게 1) 코어 과목과 21세기 주제에 대한 내용 2) 학습과 혁신의 기술 3) 정보·미디어, 기술 소양 4) 삶과 직업에 관한 소양으로 구분하여 제시하고 있다.



<그림 1> 21<sup>st</sup> Century Skills에서 학습과 혁신의 기술

본 연구에서는 2) 학습과 혁신에 관련된 소양의 관점에서 보다 구체적으로 살펴보고자 한다. 21세기의 삶과 직업을 위해 갖추어야 할 기본 학습 소양으로 크게 ‘창의성과 혁신’, ‘비판적 사고 및 문제 해결력’과 ‘의사소통 및 협동’ 3가지로 제시하고 있다. 먼저, ‘창의성과 혁신’의 경우 창의적으로 사고하고, 다른 사람과 창의적으로 작업하는 것을 포함하고 있다. ‘비판적 사고와 문제 해결력’은 효과적인 추론, 시스템적 사고와 합리적이고 효과적인 판단과 결정을 통한 문제 해결을 포함하고 있다. ‘의사소통 및 협동’에서는 분명하게 의사소통을 하고 다른 사람과 협력하는 것을 포함하고 있다.

이러한 학습과 혁신에 관련된 소양들은 최근의 사회적 특징과 더불어 그 중요성과 필요성이 많이 부각되고 있다.

## 2.3 Computational thinking

### 2.3.1 Computational thinking에 대한 정의

Wing 교수는 Computational thinking의 핵심요소로 추상화(abstraction)와 자동화(automation)를 설명하고 있다[3]. 추상화는 문제해결을 위해 문제를 분해하거나 중요한 부분을 끌어내는 것 등을 통하여 해결해야 할 문제의 복잡성을 효과적으로 해소시켜 나갈 수 있도록 하는 것이다. 자동화는 추상 개념이 수행해야 할 일들을 컴퓨팅 기기가 수행할 수 있도록 해결과정을 알고리즘화하는 과정을 의미한다.

Wing 교수에 의해 Computational thinking에 대한 용어가 소개된 이래 이에 대한 정의를 어떻게 내릴 것인가에 대한 많은 논의가 있어왔다. Google에서는 Computational thinking의 중요성을

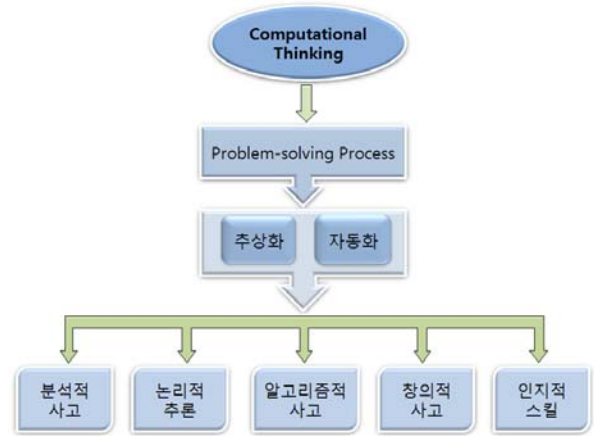
인식하고 Computational thinking을 가르치는 K-12 교사들을 위해 다양한 소스와 수업자료 등을 제공하는 사이트를 운영하고 있다. 최근 Barr, Harrison, & Coney(2011)의 연구 결과를 토대로 ISTE에서 Computational thinking에 대한 운영적 관점에서의 정의를 제시하였다[14]. 제시된 정의에 의하면 Computational thinking이란 다음과 같은 특징을 가지는 하나의 문제 해결 과정이라고 보고 있다.

- 문제 해결을 위해 컴퓨터와 다른 툴들의 사용이 가능하도록 문제를 체계화함
- 데이터를 논리적으로 조직화하고 분석함
- 모델링과 시뮬레이션과 같은 추상화를 통해 데이터를 표현
- 알고리즘적인 사고를 통해 해결과정을 자동화함
- 자원과 처리과정의 가장 효과적이고 효율적인 조합을 위해 가능한 해결책들을 식별하고 분석하고 구현함
- 문제해결 과정을 다양한 문제 상황으로 일반화함

미국 국가 연구 위원회(National Research Council)는 Computational thinking이 보통의 사람이 가져야 할 하나의 인지 기술이라고 하였다[15]. 이와 유사하게 Bundy는 Computational thinking 개념은 문제 해결 과정을 통해 다른 전공의 교육에서도 사용되어 왔으며, 계산적으로 사고하는 능력은 모든 분야에서 기본적으로 필요하다고 주장하였다. 또한, Cuny, Snyder와 Wing은 최근 Computational thinking에 대한 정의를 문제들이 정보처리 에이전트에 의해 효과적으로 수행될 수 있는 형태로 표현될 수 있도록 문제와 그 해결책들을 재구성하는 것에 관련된 하나의 사고 과정이라고 기술하고 있다.

위의 Computational thinking에 대한 정의를 종합해 볼 때 Computational thinking이란 문제 해결을 위해 단순히 컴퓨터를 흉내 내는 것이 아니라 창조적이고 다중의 추상화 차원에서 사고하는 것을 의미한다. 또한, 해결과정을 컴퓨팅 시스템에서 처리될 수 있는 형태로 알고리즘화 하는 것으로 논리적 사고의 과정이라고 할 수 있다. 뿐만 아니라, 문제를 해결하기 위해 컴퓨팅의 기본 개념들을 사용하고 컴퓨터와 다른 툴들을 사용하는

인간의 인지 능력을 포함하는 스킬이라고 볼 수 있으며, 이러한 스킬은 오늘날 모든 분야에서 기본적으로 필요하다고 볼 수 있다.



<그림 2> Computational thinking의 개념

Computational thinking은 학생들로 하여금 도전적이고 복잡한 문제를 해결하는 능력을 깊이 있게 갖출 수 있도록 도울 수 있다. 또한, 인지적 능력을 확장시키기 위해 컴퓨팅 기술들을 사용하기 위한 창의적인 방법들을 점차적으로 발전시킬 수 있는 기회를 제공할 수 있다.

이러한 Computational thinking의 특징들은 앞에서 기술된 21<sup>st</sup> Century Skills에서 요구하고 있는 '창의성과 혁신' 및 '비판적 사고와 문제 해결력'의 소양들과 일맥상통한다고 볼 수 있다.

### 2.3.2 Computational thinking에 대한 교육의 필요성

오늘날 지식정보화 사회의 한 특징으로 컴퓨팅 기술을 기반으로 한 융합이 여러 분야에 걸쳐 활발히 이루어지고 있는 것을 볼 수 있다. 컴퓨팅 기술은 융합을 촉진시키는 하나의 촉매제로서 거의 모든 분야에서 사용되고 있다. 컴퓨팅 기술은 이전엔 해결이 불가능하게 보이던 복잡하고 어려운 문제에 도전할 수 있게 하였다. 2001년의 인간 게놈(genome)의 배열은 분자 생물학의 획기적인 성취라고 할 수 있다. 그러나, 이 또한 'shotgun' 알고리즘과 같은 컴퓨팅 기술 및 Computational thinking 없이는 가능하지 않았을 것이다.

모든 분야의 직업들은 문제 해결의 창의적 활동에 밀접하게 관련되어 있다. 다양한 상황에서의 문

제 해결을 위해서는 Computational thinking 능력이 절실히 요구되고 있다. 즉, Computational thinking 능력에 의해 복잡하고 불필요한 부분을 적절히 추상화하여 적합한 추상 개념을 만들고 그 추상 개념을 컴퓨팅 기기에 의해 처리될 수 있는 형태로 자동화함으로써 문제 해결에 이를 수 있다.

Computational thinking은 이미 모든 과학과 공학 분야의 연구에 영향을 미쳐왔다. 컴퓨터 모델링과 시뮬레이션의 사용을 시작으로 하여 대용량의 데이터를 분석하기 위한 데이터 마이닝과 기계 학습과 같은 Computational 처리는 이론, 실험과 함께 과학의 세 번째 축으로 인식되고 있다[16]. 뿐만 아니라, Computational thinking은 과학과 공학 분야를 넘어 다양한 전문 분야에 영향을 끼치고 있다. 예를 들어 의학, 고고학, 경제, 법, 사회과학, 언론 분야와 같은 다양한 분야에 Computational thinking과 기법들을 연결한 연구들이 활성화되고 있다. Bundy는 Computational thinking이 물리, 생물, 의학, 철학, 건축, 교육 등 다양한 분야에 어떤 영향을 주고 있는지를 분석하였다[17]. 그는 Computational thinking에 의한 혁신이 이미 심층적으로 진전되어 사고하는 방식을 바꾸고 있다고 진단했다.

한편, Wing은 컴퓨팅 분야에서의 발전은 모든 분야의 연구자들에게 새로운 문제 해결 기법들을 상상할 수 있도록 하며 가상과 현실의 세계에서 새로운 해결책들을 테스트 할 수 있게 한다고 주장하고 있다[18]. 또한, 모든 사람이 Computational thinking 능력을 갖추으로써 다음과 같은 이점을 가질 수 있다고 주장하고 있다.

- 컴퓨팅에 적합한 문제의 여러 관점을 이해할 수 있다.
- 컴퓨터를, 기술, 문제 사이에 관련성을 평가할 수 있다.
- 컴퓨터와 기술의 능력과 한계를 이해할 수 있다.
- 컴퓨터를이나 기술을 새로운 분야에 적용하거나 응용할 수 있다.
- 새로운 기법으로 컴퓨팅을 사용할 기회를 인식 한다.
- 어느 도메인에서도 분할 정복(divide and conquer)과 같은 정보처리 전략을 적용할 수 있다.

Computational thinking의 이점에 대해서 Phillips(2008)는 다음과 같이 기술하고 있다[19].

- 학생들을 단순한 기술 소양의 차원에서 벗어나 보다 높은 지적인 수준으로 나아가게 한다.
- 소프트웨어 사용자 대신에 문제 해결자들을 양성할 수 있게 한다.
- 지식을 창조하고 자동화될 수 있는 프로세스들을 설계하는 것을 강조하게 된다.
- 창의력과 문제 해결력을 향상시킨다.
- 우리가 이미 알고 있고 가르치고 있는 문제 해결 기법들을 향상시킬 수 있다.

또한, Perkovic와 3인(2010)은 모든 분야에 종사하는 사람들이 Computational thinking 능력을 갖추으로써 얻어지는 이점에 대해 다음과 같이 제시하고 있다[20].

- 기존의 문제들 특히, 그동안 해결할 수 없었던 문제들을 새로운 관점에서 바라보고 접근할 수 있다.
- 단지 정보를 사용하기보다 지식을 생성할 수 있는 능력을 키울 수 있다.
- 창의적으로 문제들을 해결할 수 있는 능력을 키울 수 있다.
- 다양한 분야에서의 혁신을 용이하게 할 수 있다.

### 2.3.3 Computational thinking에 대한 교육 현황

교육기관에서 Computational thinking을 교육시키기 위한 노력들이 미국을 중심으로 하여 이루어지고 있으며, 최근 영국에서도 Computational thinking을 일선 학교 현장에서 교육시키기 위한 방안을 모색하기 위해 국가 연구 재단에서 프로젝트를 지원하고 있다[18].

미국의 경우 Computational thinking 교육을 위해 대학의 컴퓨터과학과에 있는 교과과정을 수정하고 있다. 즉, 컴퓨터과학 전공에 있는 첫 번째 강좌를 단지 프로그래밍이 아니라 Computational thinking을 위한 근본적인 원리와 개념들을 포함하도록 변경하고 있다[18]. 예를 들어, 카네기멜론의 경우 비전공자들을 위해 Computational thinking을 향상시키기 위한 1학년 강좌들을 개설하였다. 또한, 대학과정을 넘어 K-12 과정에

Computational thinking을 결합하기 위한 프로젝트들을 수행하고 있다. 또한, 국가 차원에서 K-12 학생 및 교사와 대학교 1, 2학년 학생의 Computational thinking 능력을 발전시키기 위해 미국 국가 과학 재단(NSF)에서는 ‘21세기를 위한 컴퓨팅교육(CE21)’ 프로젝트를 2010년도 9월에 착수하였다. CE21는 CS 10K Project를 지원하는 활동에 특별히 강조점을 두고 있는데, CS 10K는 중요과목으로서 컴퓨터과학에 관한 새로운 AP 강좌의 개설과 함께 고등학교 교과과정의 개정을 촉진하고, 2015까지 10,000 명의 컴퓨터과학 교사들을 준비하는 것을 목표로 하고 있다.

한편, 미국의 경우 산업체에서 Computational thinking의 비전이 활성화될 수 있도록 여러 관점에서 돕고 있다. 2006년 이래로 Google과 Microsoft의 도움을 받아 카네기멜론 대학, UCLA, 워싱턴 등의 대학교에서는 고등학교 선생님들을 대상으로 Computational thinking에 관한 워크숍 ‘CS4HS’을 운영하고 있다.

영국의 경우 2010년 8월에 영국 국립 과학 재단(Royal Society)에서는 컴퓨팅 분야의 커뮤니티, 전문가 단체, 대학, 산업체 등의 24개 조직들의 지원을 받아 학교에서 Computational thinking을 교육시키기 위한 방법들을 모색하기 위한 프로젝트를 착수하였다.

### 2.3.4 Computational thinking의 교육내용

미국에서는 일선교육 현장에서 Computational thinking을 교육시키기 위한 일환으로 Computational thinking의 중심 개념들을 정립하기 위한 시도들을 수행하였다. 그러한 시도들은 크게 K-12 수준에서와 대학 수준으로 구분될 수 있다.

특히, K-12 수준에서의 노력들은 Computational thinking이 모든 분야에 걸쳐 접근되고 적용될 수 있는 하나의 문제 해결 방법으로 간주하고 Computational thinking의 중심 개념들을 끌어내는 노력을 하였다[5]. 이 연구에서 제안하고 있는 중심 개념으로 Data collection, Data analysis, Data representation, Problem decomposition, Abstraction, Algorithm & procedures, Automation, Parallelization과 Simulation이 있다.

이에 대한 자세한 내용은 <표 3>에서 볼 수 있다.

또한, K-12 수준의 학생들에게 Computational thinking를 확산시키기 위한 일환으로 ITEST (International Society for Technology in Education) 워킹 그룹에서는 3가지 도메인을 선택하여 Computational thinking에 관련된 예제들을 개발하고자 하였다. 이 세 가지 도메인은 game design, simulations and model과 robotics 분야이다. 또한 이 그룹에서는 학생들의 Computational thinking을 발전시키기 위한 하나의 프레임워크를 제안하고 있는데, 그것은 3 단계 즉, use 단계, modify 단계, create 단계로 구성된다[21].

대학수준에서의 Computational thinking을 위한 교육과정의 틀을 마련하는 차원에서 DePaul 대학의 Perkovic외 3인(2010)은 Denning에 의해 정의된 컴퓨팅의 7가지 원리인 Computation, Communication, Coordination, Coordination, Recollection, Automation, Evaluation, Design에 기초하여 수정과 보완을 거쳐 관련 세부 개념들을 <표 2>와 같이 정의하였다[20].

<표 2> 대학과정에서 Computational thinking을 위한 중심 개념

| CT concepts   | Keywords   |
|---------------|--|
| Computation   | state and state transition, algorithm, program, recursion and iteration, decision tree, problem complexity                         |
| Communication | messages, sender/receiver, communication protocol, message compression, message encryption, communication channel, encoder/decoder |
| Coordination  | interacting processes/agents, inter-process protocols, synchronization, concurrency  |
| Recollection  | storage media, data hierarchy, data manipulation, data locality and caching  |
| Automation    | mapping of algorithm to physical computing object  |
| Evaluation    | data analysis, statistics, data mining, recommender system   |
| Design        | abstraction, modeling, modularity, information hiding, class, underlying structure   |

## 3. 컴퓨터과학과 Computational thinking

오늘날의 지식정보화 사회에서 컴퓨터는 우리가 살아가고, 생각하고, 행동하는 방법에 지대한 영향

을 미치고 있다. 개인이나 사회가 해결해야 할 문제들도 그와 관련된 대상들을 디지털화함으로써 컴퓨팅 원리나 개념, 기술 등이 적용될 수 있게 되었다. 또한 지식정보화 사회의 특징으로 컴퓨팅의 핵심요소들을 활용해 특정 학문과 컴퓨터과학을 접목시키는 융합이 활발하게 이루고 지고 있다 [6]. 이에 따라, 과거에는 해결하지 못하였던 복잡하고 대단위의 문제들을 해결할 수 있게 되었다.

21세기 직업들의 대부분은 컴퓨터과학의 이해를 요구하며, 문제 해결의 창의적 활동에 밀접하게 관련되어 있다. Bundy(2007)는 컴퓨팅에 의한 혁신이 이미 심층적으로 진전되어, 다양한 학문 분야에서 생각하는 방식을 바꾸고 있다고 진단했다. 그리고 컴퓨터과학 분야의 개념들이 각 분야의 가설과 이론을 기술하는 새로운 언어로 활용되고 있고 컴퓨팅이 인지 능력의 확장을 가져다 준다고 주장하고 있다[17].

컴퓨터과학의 핵심은 Computational thinking의 신장이다. 즉, 문제 해결과정 자체에 대해서 학생들이 생각하도록 가르치는 것이다. Computational thinking는 컴퓨팅의 기본 개념들을 활용해 문제를 해결하고, 시스템을 설계하고, 시스템 상황 속에서 인간 행동을 이해하는 접근방법이다.

Computational thinking의 요소들은 컴퓨터과학 분야에서 자주 발견되는 Computational 개념, 원리, 방법, 언어, 모델과 툴들을 포함한다. 따라서, Computational thinking은 이러한 요소들을 이용하여 어려운 문제를 재정의 하는 것을 포함하게 된다[22].

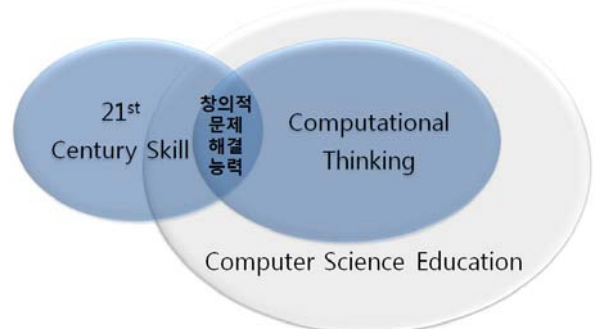
그러한 요소들에는 축소와 변형, 병렬처리, 타입 검사, 모델 검사, 문제의 추상화와 분해, 문제 표현, 모듈화, 에러 방지, 테스트, 디버깅, 시뮬레이션, 휴리스틱 추론, 스케줄링, 검색 전략, 알고리즘과 처리과정의 복잡도의 분석 등이 포함된다. 뿐만 아니라, 알고리즘, 프로세스, 상태머신, 기계 학습, 재귀적 호출, 파이프라인, 최적화 등과 같은 컴퓨터과학의 개념들이 Computational thinking의 문제 해결과정에 포함된다.

#### 4. 21<sup>st</sup> Century Skills와 Computational thinking의 관점에서 ‘정보’ 교육과정 분석

##### 4.1 교과 성격 및 목표 관점에서 분석

21<sup>st</sup> Century Skills와 Computational thinking의 핵심적인 사항을 살펴보면 다음과 같다. 21<sup>st</sup> Century Skills에서 ‘학습과 혁신’ 부분의 주요 핵심내용은 ‘창의성’, ‘협동능력’과 ‘문제해결력’을 들 수 있는데, 창의성은 창의적으로 사고하고 다른 사람들과 창의적으로 일을 하는 것으로 정의하고 있다. 즉, 새롭고 다양한 관점에서 열린 사고를 하며, 새로운 아이디어를 다른 사람들과 효과적으로 교류하고 그 아이디어를 실현하는 것을 의미한다. 문제 해결력은 효과적으로 추론하고, 시스템적인 관점에서 문제를 분석해보는 것이다. 또한, 여러 가지 정보들과 주장들을 분석 및 평가하여 창의적이고 혁신적인 방안을 마련함으로써 문제를 해결하는 것이다.

Computational thinking의 주요 개념은 컴퓨팅의 관점에서의 문제 해결이다. 즉, 문제 해결을 위해 가장 적합한 추상화 단계를 확인하고 추상적 개념의 자동화 수행을 위해 여러 가지 개념, 원리, 방법 등을 통한 해결과정을 단계별로 알고리즘화 하는 것이 된다. 또한 이 문제 해결을 위해 가장 적합한 자원과 과정을 선택하고 조합하는 과정이 포함된다.



<그림 3> 21<sup>st</sup> Century Skills, Computational thinking, 컴퓨터과학교육과의 관련성

21<sup>st</sup> Century Skills와 Computational thinking에서 공통적인 주요 핵심 개념은 ‘창의성’과 ‘문제해결’이라 볼 수 있다.

그런데, ‘정보’ 과목의 성격과 특징을 분석해보면 ‘정보’ 과목 역시 창의적 문제 해결력 향상이 교과 추구하는 주요 핵심 개념이라 볼 수 있다. <표 1>에서 볼 수 있는 바와 같이 과목의 성격 정의에서 ‘정보’ 과목을 “정보의 기본적인 개념과 원리



를 습득하고 정보처리를 위한 기능을 익힘으로써 창의적이고 실질적인 문제해결 능력을 신장시키는 과목이다”라고 제시되어 있다.

또한, 구체적으로 교과 영역 및 영역별 목표에서 그 핵심 특징을 분석해보면, 특히, 제 3영역의 ‘문제해결 방법과 절차’에서 영역의 목표를 ‘실생활에서 발생하는 다양한 문제를 정보처리의 관점에서 이해하고 정보처리의 지식과 기능을 활용하여 창의적이고 능동적으로 문제를 해결할 수 있다’로 되어 있다.

한편 문제를 정보처리 관점에서 창의적으로 사고하는 능력과 방법을 길러주기 위해서는 컴퓨터의 구성요소와 그 동작 과정을 체계적으로 학습함으로써 생활 속에 적용된 정보처리 관련 문제를 스스로 찾고 응용력을 길러주는 것이 필요한데 이에 관한 내용을 ‘정보기기의 구성과 동작’ 영역에서 다루고 있다. 또한, ‘정보의 표현과 관리’ 영역에서는 일상생활에서의 정보표현과 다양한 형태와 방법을 이해함으로써 정보들을 논리적으로 조직화할 수 있고 문제들에 새로운 정보처리 방법을 적용할 수 있는 능력을 키울 수 있도록 정의하고 있다.

이와 같이 ‘정보’ 과목의 성격, 영역별 목표를 분석해볼 때 ‘정보’ 과목은 21<sup>st</sup> Century Skills와 Computational thinking 능력을 효과적으로 신장시킬 수 있는 핵심 교과목이라고 볼 수 있다. 따

라서, 21세기의 성공적인 삶을 위해서는 ‘정보’ 과목에 대한 체계적인 교육이 절실히 요구된다.

#### 4.2 교육과정 내용의 관점에서 분석

미국의 K-12 수준에서 제시한 Computational thinking의 중심 개념들과 2007년 개정 ‘정보’ 교육과정의 내용을 분석하면 <표 3>과 같다. <표 3>에서 볼 수 있는 바와 같이 Computational thinking의 중심 개념들과 ‘정보’ 교과 교육과정에서 제시하고 있는 4가지 영역의 내용들을 관련 지을 수 있다. Computational thinking의 중심 개념들과 ‘정보’ 과목의 내용이 일대일 대응이 되지는 않지만 대부분의 Computational thinking의 중심 개념들이 ‘정보’ 과목에서 다루어진다고 볼 수 있다. 특히, ‘정보의 표현과 관리’ 영역과 ‘문제해결 방법과 절차’ 영역이 유사함을 볼 수 있다. ‘Data representation’에 관련된 내용으로 ‘정보’ 교과에서는 ‘정보의 표현과 관리’ 영역을 두어 ‘선형구조’와 ‘비선형구조’로 구분하여 자료 구조 내용을 다루고 있다. ‘Problem decomposition’, ‘Abstraction’과 ‘Algorithm & procedures’와 관련된 개념들은 ‘정보’ 교과의 ‘문제해결 방법과 절차’ 영역에서 ‘문제의 분석과 표현’, ‘알고리즘의 개요와 실제’, ‘문제 해결 전략’과 ‘객체지향 프로그래밍’으로 다루고 있다.

<표 3> 미국의 K-12 수준에서 제시한 Computational thinking의 중심 개념들과 우리나라 2007년 개정 ‘정보’ 교육과정과의 비교

| 미국                     |  | 한국           |            |                             |
|------------------------|--|--------------|------------|-----------------------------|
| CT concepts            | Keywords   | 교과 영역        | 중학교 교과 내용  | 고등학교 교과 내용                  |
| Data collection        | data source  | 정보사회와 정보기술   | 정보의 수집과 전달 |                             |
| Data analysis          | statistical calculations   |              |            |                             |
| Data representation    | data structure such as array, linked list, stack, queue, tree, and graph | 정보의 표현과 관리   | 선형구조       | 비선형구조                       |
| Problem decomposition  | objects, methods, main, functions  | 문제해결 방법과 절차  | 문제의 분석과 표현 | 문제 해결 전략, 구조적 프로그래밍, 객체지향   |
| Abstraction            | procedure, encapsulation   |              |            |                             |
| Algorithm & procedures | algorithm, implementation  |              | 알고리즘의 개요   | 함수를 이용한 프로그래밍<br>객체지향 프로그래밍 |
| Automation             |  |              | 알고리즘의 실제   |                             |
| Parallelization        | threading, pipelining, dividing up data                                  | 정보기기의 구성과 동작 |            | 프로세스의 관리                    |
| Simulation             | algorithm animation, parameter sweeping                                  |              |            |                             |

하지만, ‘시물레이션’ 개념에 대응되는 내용이 ‘정보’ 교육과정에는 제시되어 있지 않다. Moursund (2009)에 의하면 Computational thinking안에 기저를 이루고 있는 개념은 ‘해결하려고 하는 문제들의 모델들과 시물레이션들을 개발하는 것이다’라고 주장하고 있다[23]. 즉, 하나의 문제를 해결하기 위해서는 문제 상황에 존재하는 여러 가지 요소들의 상호작용들을 고려하고 이를 시스템으로 모델링 혹은 시물레이션 하는 작업이 필요하다. 이런 관점에서 볼 때 향후 교육과정의 개정 시 이에 개념들을 포함시키는 것도 바람직하다고 볼 수 있다.

## 5. 컴퓨터 교육의 현황 및 발전 방안

### 5.1 컴퓨터 교육의 현황 및 문제점

우리나라 컴퓨터교육의 실태를 정보·컴퓨터 과목의 이수율, 편성 시수, 전공 교사의 확보의 관점 등의 3가지 관점에서 살펴보면 다음과 같다[6]. 먼저, 정보·컴퓨터 과목의 이수율의 경우를 살펴보면, 2009년에 ICT 활용교육 지침이 교육과학기술부의 행정규제철폐로 인해, 초·중고교에서 선택과목으로 컴퓨터 관련 교과를 우선적으로 배정해야 한다는 지침이 삭제되면서 컴퓨터 관련 교과의 이수 비율이 급격히 낮아지고 있다. 서상기 의원의 자료에 의하면 2009년의 경우 중학교는 62.1%, 고등학교는 12.4%만이 이수하고 있는 것으로 나타났다. 2007년부터 초중등학교의 컴퓨터 관련 과목의 이수율이 해마다 떨어지고 있는 실정이다.

정보·컴퓨터 과목의 편성 시수를 살펴보면, 중등학교에서 정보·컴퓨터 과목을 이수하는 경우에도 교육과정에 제시된 학습 내용을 교육하기에는 많은 시간이 부족한 실정이다. 2007 개정 교육과정의 경우 중학교는 정보1, 정보2, 정보3으로 3단계로 구성되어 있지만 3년 동안 1단위(1학기 1시간)만 교육하는 중학교가 이수 학교의 52.4%이다. 이는 중학교 정보의 학습 내용 3단계 중에서 1단계도 미치지 못하는 시간수이다.

마지막으로 정보·컴퓨터 전공 교사의 확보의 관점에서 살펴보면, 중등학교 정보·컴퓨터교육의 문제점 중의 하나인 전공 교사의 부족이다. 교육

현장에서 실제 교육을 담당하고 있는 정보·컴퓨터 교사의 경우 비전공 교사가 연수를 통해 부전공 자격을 얻어 교육하는 상치 교사의 비율이 지역에 따라 큰 편차를 보이고 있다. 울산과 강원도의 경우 상치 교사의 비율이 58.1%와 68.3%까지 이루고 있다.

이와 같이 우리나라 컴퓨터교육의 현황을 살펴보면, 과거에 비해 컴퓨터과목의 이수 비율이 현저히 낮아지고 있는 추세이다. 이러한 배경에는 IT 분야에 대한 소극적 지원 및 입시위주의 교육과정 등 국가·사회적으로 여러 가지 요인이 작용하고 있다. 뿐만 아니라, 그동안 컴퓨터교육의 많은 부분이 비전공 교사들에 의해 교육됨으로 인하여, 체계적이고 심도 있는 컴퓨터교육이 이루어지기 보다는 단편적인 지식을 전수하는 경우가 많았다. 이로 인하여, 실제 컴퓨터교육에 대한 만족도는 그리 높지 않은 것으로 조사되고 있다.

한편, 2009년 개정 교육과정이 개편되어 고시됨에 따라 2011년부터 교과별 교육과정이 적용된다. 2009년 개정 교육과정을 실제 적용하면서 선택 과목의 축소 운영으로 인하여 정보교과가 더욱 축소 또는 폐지가 되고 있는 실정이다.

### 5.2 컴퓨터 교육의 발전 방안

Computational thinking은 21세기를 살아가는 모든 사람이 갖추어야 할 하나의 기본 소양이라고 볼 수 있다. 모든 분야의 직업들은 문제 해결의 창의적 활동에 밀접하게 관련되어 있다. 이러한 다양한 상황에서의 문제 해결을 위해서는 Computational thinking 능력이 절실히 요구되고 있다.

그렇지만, 우리나라의 경우 아직까지 Computational thinking에 대한 개념에 대해 생소하게 느끼거나, 알고 있더라도 잘못 인식하고 있는 경우가 많은 듯하다. 즉, Computational thinking을 단지 프로그래밍으로 보거나 혹은 하드웨어와 소프트웨어로 잘못 인식하는 경우가 많다. 따라서, 이에 대한 올바른 개념 정립이 우선적으로 요구된다. 이와 함께 Computational thinking 능력이 21세기에 우리가 접하게 되는 다양한 문제들을 정의하고, 이해하고 해결하는 능력

을 개발하는데 핵심 역량이 된다는 인식을 확산시키는 노력이 필요하다.

또한 Computational thinking이 컴퓨터과학으로부터 나왔고, 컴퓨터과학 교육이 이 Computational thinking을 신장할 수 있다는 인식과 함께 이를 활성화시키기 위한 노력이 따라야 할 것이다. 컴퓨터 교육에 대한 우리사회의 일반적인 인식은 컴퓨터를 전공할 학생에게만 필요한 과목이라고 생각한다는 점이다. 따라서, 이에 대한 인식의 전환이 우리사회 전반에서 필요하다고 볼 수 있다.

미국과 영국의 경우 국가 과학 재단에서 Computational thinking이 모든 전공에 걸쳐 필요하고 다른 학문과의 융합을 위해서는 이것이 기본이 된다는 인식아래 이에 대한 교육을 초·중고에서 체계적으로 하기 위한 연구들에 최근 적극적으로 지원하고 있는 점도 주지할 만하다.

현재 우리나라의 초·중등 교육과정에서 Computational thinking 교육을 담당해야 할 교과는 컴퓨터과학을 그 주 영역으로 하고 있는 ‘정보’ 교과이다. ‘정보’ 교과를 활성화시키기 위한 방안으로는 ‘정보’ 교과의 성격을 Computational thinking의 관점에서 보다 세부적이고 미시적인 목표를 가지도록 설정하여야 할 것이다. 또한, Computational thinking에 대한 기본 개념의 학습을 위한 체계적인 교육 내용에 대한 연구가 이루어져야 한다. 특히, 초·중등 과정에서 어떤 개념들을 포함시켜 교육시킬 것인지에 대한 심도 있는 연구가 요구된다. 뿐만 아니라, Computational thinking을 효과적으로 가르치거나 학습시키기 위한 교수 모델 및 교수 학습 방법에 대한 연구가 필요할 것이다.

현재 2007 개정 교육과정에 맞춰 개발된 ‘정보’ 과목의 교과서의 경우, 기존의 교과서에 비해 학생들의 논리적 사고력, 창의력, 문제 해결력을 신장시키기 위한 부분들이 포함되어 있다. 하지만, 내용 구성을 보면, 대학의 컴퓨터과학 전공과목들의 기초내용을 나열한 것과 같이 기술되어 있는 경우가 많다. 그러므로, Computational thinking의 관점에서 보다 체계적인 교육내용에 대한 연구와 함께 적절한 교과서가 집필 되어야 할 것이다.

또한, 국가적인 차원에서 ‘정보’ 교과에 대한 교

사 양성·체제를 수립하고 Computational thinking에 기초한 창의적 문제해결에 관련된 교육 내용을 다루도록 하는 것이 요구된다. 뿐만 아니라, 타교과 교사들에 대해서도 교원 연수시 Computational thinking의 개념을 소개하고 ‘정보’ 교과에 대한 교육현장의 이해를 높일 수 있도록 분위기를 확산시키는 것이 필요하다.

## 6. 결 론

본 연구에서는 21세기를 살아가는 현대인으로서 기본적으로 갖추어야 할 21<sup>st</sup> Century Skills와 Computational thinking의 특징을 살펴보고, 이것들이 컴퓨터 교육과의 어떤 관련성이 있는지를 분석하였다.

21<sup>st</sup> Century Skills와 Computational thinking의 경우 핵심 개념은 창의적 문제 해결력의 신장이라고 볼 수 있다. 이러한 창의적 문제 해결력 신장을 위한 핵심 과목이 2007년 개정된 ‘정보’ 교과이다. ‘정보’ 교과의 주요 성격이 창의적이고 실질적인 문제해결 능력을 신장시키는 것이며, 교과내용도 이를 중심으로 하여 구성되었다. 따라서, 21세기를 살아가는 현대인이 성공적인 삶을 영위하기 위해서는 ‘정보’ 과목에 대한 체계적인 교육이 절실히 요구된다.

한편, ‘정보’ 교과를 활성화시키기 위해서는 ‘정보’ 교과의 성격을 Computational thinking의 관점에서 재설정하는 노력과 함께 그것의 기본 개념들을 교육시키기 위한 체계적인 교육 내용에 대한 연구가 이루어져야 할 것이다.

현재 우리나라 교육 현장의 상황은 ‘정보’ 교과가 축소되고 있는 실정이다. ‘정보’ 교과의 축소는 미래의 개인의 삶뿐만 아니라 우리나라 국가 경쟁력의 하락이라는 큰 문제를 가져올 수 있다. 이에 따라 ‘정보’ 교과를 활성화시키기 위한 다각적인 측면에서 노력과 연구가 필요하다고 볼 수 있다.

## 참 고 문 헌

- [1] Partnership for 21<sup>st</sup> Century Skills, The Intellectual and Policy Foundations of the 21 Century Skills Framework, Retrieved

from <http://www.p21.org/>

[2] Partnership for 21<sup>st</sup> Century Skills, Retrieved from [http://www.p21.org/documents/P21\\_Framework\\_Definitions.pdf](http://www.p21.org/documents/P21_Framework_Definitions.pdf)

[3] Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 19(3), 33-35.

[4] Cuny, J. Snyder, L. & Wing, J. M. (2010). Demystifying Computational thinking for Non-Computer Scientists, work in progress.

[5] 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 archive*, 2(1).

[6] 한상렬 외 6인 (2010). 융합시대의 과학적 사고력 신장을 위한 컴퓨터교육 발전방안. 한국공학한림원 정책연구보고서.

[7] 김경훈 외 6인 (2006). 중고 컴퓨터 선택과목 교육과정 시안 연구 개발. 한국교육과정 평가원.

[8] 유현창, 길준민 (2007). 2007년 개정교육과정의 개선에 관한 연구. *한국 컴퓨터교육 학회지*, 2(1), 3-15.

[9] 신수범, 이태욱(2005). 컴퓨터교과의 성격 분석과 교육과정 구성 전략. *컴퓨터교육학회 논문지*, 8(3).

[10] 김홍진(2005). 제 7차 교육과정에서 컴퓨터 교과 내용과 문제점. *교원교육-한국교원대학교*, 20(4).

[11] 박정호, 이재윤, 이태욱 (2006). 컴퓨터과학교육을 위한 중학교 컴퓨터교육과정 연구. *컴퓨터교육학회 논문지*, 9(2).

[12] 이원규, 김현철, 정순영, 유승욱, 한희섭, 김종혜, 전수진 (2007). *정보교육론*. 서울: 홍릉과학출판사

[13] 이영준, 이은경 (2008). 정보교육의 본질과 전망. *한국 컴퓨터교육학회 논문지*, 11(3), 2008.

[14] Bar, D., Harrison, J. & Conery, L.,(2011). Computational thinking: A Digital Age Skill for Everyone. *Learning & Learning with Technology*, 38 (6), 20-23.

[15] National Research Council (2010). Report of a Workshop on the Scope and Nature of Computational thinking. Retrieved from <http://www.nationalacademies.org/cp/project/view.aspx?key=48969>

[16] President's Information Technology Advisory Committee (PITAC) (2005). Report to the

President on computational science: Ensuring America's competitiveness. Washington, DC: U.S. Government Printing Office.

[17] Bundy, A. (2007). Computational thinking is Pervasive. *Journal of Scientific and Practical Computing*, 1(2), 67-69.

[18] Wing, J. (2011). Computational thinking – What and Why?. *CMU Research Notebook*. Retrieved from <http://link.cs.cmu.edu/article.php?a=600>

[19] Phillips, P.(2007). Computational thinking: A Problem-Solving Tool for Every Classroom. Present in the International Conference on NECC.

[20] PerKovic, L., Settel, A., Hwang, S., & Jones, J. (2010). A Framework for Computational thinking across the curriculum, *Proc. of International Conference on ITiCSE*.

[21] Lee, I., et al., (2011). Computational thinking for youth in practice. *ACM Inroads archive*, 2(1).

[22] Deek, F., et al.,(2006). A Model curriculum for k-12 computer science. Final Report of the ACM K-12 Task Force Curriculum Committee.

[23] Moursund, D. (2009). Computational Thinking. Retrieved from *IAE-pedia*. [http://iae-pedia.org/Computational\\_Thinking](http://iae-pedia.org/Computational_Thinking)



## 최 속 영

1998 전북대학교 전산학과 (이학사)  
 1991 전북대학교 전산학과 (이학석사)  
 1996 충남대학교 전산학과(이학박사)  
 2008 Nova Southeastern University  
 교육공학 및 원격교육(교육학박사)  
 1996~현재 우석대학교 컴퓨터교육과 교수  
 관심분야: 지능형 시스템, 유러닝, 컴퓨터과학교육