

# 중등 소프트웨어 수업에서 컴퓨팅 사고력 기반 창의적 문제해결(CT-CPS) 수업모형의 인지적·정의적 효과성 분석

전용주<sup>†</sup> · 김태영<sup>††</sup>

## 요 약

본 연구는 중학생을 대상으로 컴퓨팅 사고력 기반의 창의적 문제해결(이하 CT-CPS) 수업모형을 통해 설계한 소프트웨어 수업을 적용하여, 인지적, 정의적 영역에서의 효과성을 분석하기 위하여 수행하였다. 이러한 연구의 목적을 달성하기 위해 본 연구자는 이론적 배경 탐색을 통해 인지적·정의적 활동을 수행할 수 있도록 CT-CPS 수업모형 기반의 소프트웨어 수업을 설계하였다. 이후 이를 중학교 소프트웨어 수업에 적용하고, 수업의 전, 후에 3가지 검사를 실시하였다. 연구 결과 3가지 검사의 인지적, 정의적 영역에 해당하는 대부분의 요소에서 통계적으로 유의한 향상이 관찰되었고, 중학생의 소프트웨어 수업에서 CT-CPS 수업모형의 인지적, 정의적 영역에서의 효과성 검증과 시사점을 논할 수 있었다.

주제어 : 소프트웨어 교육, CT-CPS 수업모형, 인지적·정의적 효과성

## The Analysis of Cognitive and Affective Effects on the CT-CPS Instructional Model for the Software Education Class in Middle School

YongJu Jeon<sup>†</sup> · TaeYoung Kim<sup>††</sup>

## ABSTRACT

The purpose of this study is to verify the effectiveness of a CT-CPS(Computational Thinking-based Creative Problem Solving) instructional model on the cognitive and the affective area of middle school students' software class. To achieve our goal, we explored theoretical background and designed a lesson plan based on CT-CPS instructional model. Then we analyzed our experimental results after applying the lesson plan to middle school students. We performed our experiment to an experimental group by using our CT-CPS instructional model-based lesson plans, and we carried out three pre and post tests about cognitive and affective area, i.e. creative problem solving ability, meta cognition and motivation of learning. As a result, most of the test factors were statistically improved, so the effectiveness of the CT-CPS instructional model on the cognitive and the affective area of middle school students' software class was verified.

**Keywords** : Software Education, CT-CPS instructional model, Cognitive and affective effectiveness

---

<sup>†</sup> 정 회 원: 청주 사직초등학교 교사  
<sup>††</sup> 중신회원: 한국교원대학교 컴퓨터교육과 교수 (교신저자)  
논문접수: 2017년 6월 9일, 심사완료: 2017년 7월 24일, 게재확정: 2017년 7월 27일

## 1. 서론

디지털 경제 시대의 도래와 신기술의 등장은 우리의 삶을 송두리째 바꿔가고 있다. 수학적사고와 과학적 사고를 바탕으로 한 지식과 자본이 곧 힘이고, 능력이었던 산업 경제 시대와는 달리 최근의 디지털 경제시대에는 학문, 예술, 산업 등의 영역을 융합한 실질적 산출물들의 성과가 늘어나면서 창조적인 융합능력이 중요시 되고 있다. 특히 각 국가나 기업들은 학문, 예술, 산업영역과 소프트웨어 기술의 융합을 강조하며 이에 대한 투자와 연구를 통한 가시적인 성과 창출을 기대하고 있기도 하다[1][2][3].

이러한 시대적인 흐름은 창조, 융합의 통로인 컴퓨팅 및 소프트웨어에 관한 관심을 전 세계적으로 이슈화 시키고 있으며, 이로 인해 공교육과 민간 기업 및 단체에서도 이와 관련된 다양한 교육을 실시하기에 이르렀다[4][5][6][7][8][9][10][11][12][13][14][15].

국내에서도 소프트웨어교육 운영지침(2015), 2015 개정 교육과정 등에서 소프트웨어 교육과 관련한 사항을 천명하며 그 필요성과 목적에 대해서 다양한 장을 통해 교사, 학부모, 사회 등과 공유하고 있다[10][11]. 또한 교육부와 미래창조과학부(2016)에서는 전국 초, 중, 고등학교 중에서 900개의 선도학교를 선정하고, 소프트웨어 교육과 관련된 실질적인 실천을 할 수 있도록 정책을 전개하고 있다[16][17].

이러한 변화가 이루어지고 있는 시점에서, 현장 교사의 입장에서 생각해 보면, 소프트웨어 교육이 무엇인지 이해가 되고, 그 필요성이나 목적에는 충분히 동의할 수 있으나, 실제로 소프트웨어 교육을 어떻게 디자인하고 적용할 수 있을지에 고민이 되는 것이 사실이다. 이렇다 보니, 인지적, 정의적 영역에서의 균형 있는 역량 함양 보다는 당장 손에 잡히는 다양한 도구 사용법 교육 등에 치우쳐서 인지적, 도구적 측면에만 치우친 교육과정을 계획하고 운영하게 될 수 있다.

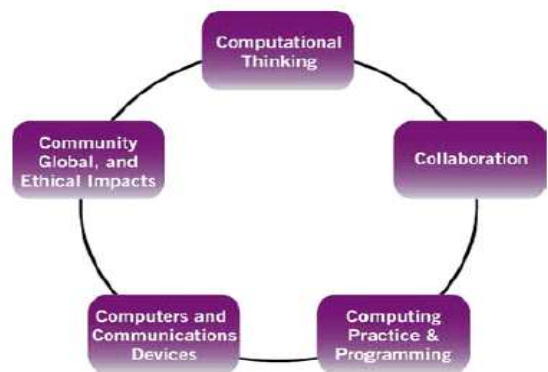
한편 본 연구자는 선행 연구에서 소프트웨어 교육에서 학생들의 인지적, 정의적 역량을 균형 있게 함양하기 위한 컴퓨팅 사고력 기반 창의적 문제해결(CT-CPS) 수업모형을 개발하였는데

[18][19][20][21], 본 연구에서는 이에 대한 후속 연구로 기 개발된 수업모형을 토대로 수업을 디자인하여 중학교 방과 후 소프트웨어 수업 과정에 적용하여, 수업모형의 인지적 측면과 정의적 측면에서의 효과성을 분석하고자 하였다.

## 2. 이론적 배경

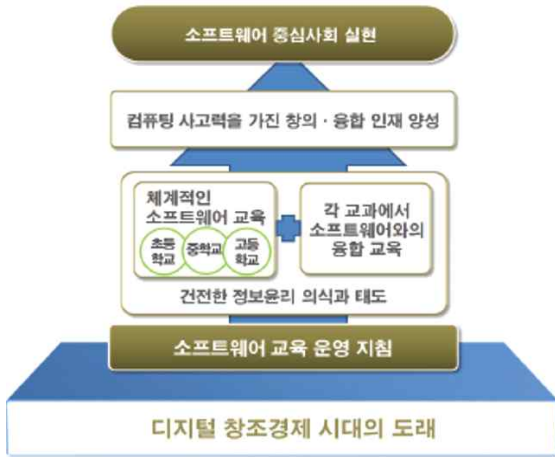
### 2.1 소프트웨어 교육과 인지적 정의적 역량

소프트웨어 교육과 관련된 국외의 컴퓨팅 교육 과정을 살펴보면, 미국의 CSTA(2011)의 컴퓨터과학 수업 표준안에서는 컴퓨팅 사고력과 프로그래밍, 컴퓨팅/의사소통기기 등 인지적 영역 외에도 협업, 지역사회 및 글로벌 사회에 대한 윤리적 영향 등의 학습 영역을 마련하여 인지적, 정의적 역량을 균형 있게 함양하도록 내용을 제시하였음을 알 수 있다[8]([그림 1](#) 참조). 또한 영국(2013)의 국가 컴퓨팅 교육과정에서도 각 Key Stage 별로 인지적인 영역에서의 내용요소와 정보윤리 및 태도와 관련된 정의적인 요소를 함께 제시하였다[9].



[그림 1] 컴퓨터과학 교육의 5개 영역 [8]

교육부(2015)에서 제시한 소프트웨어 교육 운영지침에서도 ‘컴퓨팅’교과로서의 체계적인 소프트웨어 교육과, ‘융합’교육으로서의 소프트웨어 교육을 통해 건전한 정보윤리 의식과 태도를 바탕으로 컴퓨팅 사고력을 가진 창의·융합적인 인재를 양성하도록 제시하고 있어, 정의적 영역과 인지적 영역에서의 균형 있는 역량을 가진 인재를 양성하도록 인재상을 제시하고 있다([그림 2](#) 참조)[10].



[그림 2] 소프트웨어 교육의 방향 [10]

2015개정 중등 정보 교육과정에서도 교과외의 성격을 ‘컴퓨터과학적 지식과 기술의 탐구와 더불어 실생활의 문제해결을 위해 새로운 지식과 기술을 창출하고 통합적으로 적용하는 능력과 태도의 함양을 추구하는 교과’로 규정하고 있으며, 목표에서도 정보윤리의식, 정보보호능력, 정보기술활용능력, 실생활의 문제를 창의적이고 효율적으로 해결하는 능력과 해결하고자 하는 태도를 기르는데 중점을 두고 있다고 진술하고 있다[11].

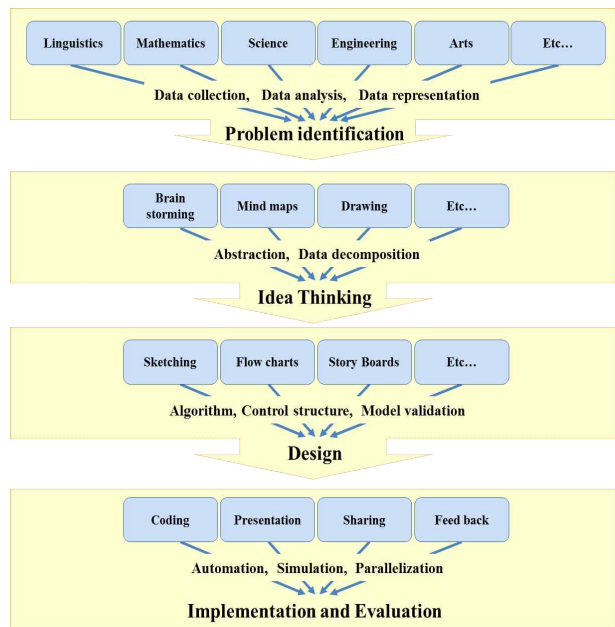
한편, 김지현 외(2016)는 “디지털 시대의 복잡한 문제를 해결하기 위해서는 컴퓨팅 사고력 뿐만 아니라, 지식의 융합과 함께 주변과의 공감을 통해 소통과 협력을 이끌어내고 서로를 인정해주는 배려하는 마음이 필요하다”고 주장하였고[22], 김병조 외(2014)는 컴퓨팅 사고력의 기반한 IT융합형 인재는 실제 문제인식능력, 도덕성, 의사소통능력, 통합적사고, 문제해결능력, 창의성 등이 필요하다고 주장하여[23], 인지적, 정의적 영역의 균형 있는 역량이 필요함을 시사하였다.

이상의 내용을 종합해 보면, 소프트웨어 교육의 내용은 인지적, 도구적인 방향으로만 이루어질 수 없으며, 인지적인 역량과 정의적인 역량이 균형 있게 함양될 수 있도록 적용되어야 함을 알 수 있다.

## 2.2 CT-CPS 수업모형

CT-CPS(Computational Thinking-based Creative Problem Solving) 수업 모형은 정보교과

수업에서 컴퓨팅 사고력을 바탕으로 실생활 문제를 창의적으로 해결해 가는 수업 모형이다 [19][20][21]. 이 수업 모형은 창의적 문제 해결 단계(Osborn 1963; Parnes et al. 1967, 1977; Isaksen & Treffinger 1985)에 컴퓨팅 사고력의 요소를 융합하여 구안한 모형으로, 총 4단계의 수업과정을 거치게 된다. [그림 3]은 CT-CPS의 수업 과정을 도식화한 것이며, <표 1>은 CT-CPS 수업 모형의 단계와 각 단계별 교수학습 활동이다.



[그림 3] CT-CPS 수업모형의 수업 과정 [20][21]

<표 1> CT-CPS 수업모형의 단계 [20][21]

단계	교수·학습 활동
단계 1. 문제인식 및 분석	<ul style="list-style-type: none"> <li>• 동기부여</li> <li>• 수업에서 사용할 도구 선정 및 소개</li> </ul>
	<ul style="list-style-type: none"> <li>• 문제인식                             <ul style="list-style-type: none"> <li>- 문제화 할 주제와 관련된 자료수집 (인터뷰, 설문조사, 현장조사 등)</li> <li>- 수집된 자료를 구조화하고 분석하기 (표, 그래프, 그림 등)</li> </ul> </li> </ul>
	<ul style="list-style-type: none"> <li>• 문제화하기                             <ul style="list-style-type: none"> <li>- 이 문제를 해결해야 하는 목적, 문제와 관련된 지식, 문제의 가치 등</li> <li>- 최종적으로 해결되어야 할 문제를 표현하기(글, 프레젠테이션 등)</li> </ul> </li> </ul>
단계 2. 아이디어 구상	<ul style="list-style-type: none"> <li>• 아이디어 구상하기                             <ul style="list-style-type: none"> <li>- 컴퓨팅을 활용하여 문제를 해결할 수 있는 다양한 아이디어 구상하기 (브레인스토밍, 마인드맵, 그림 등)</li> </ul> </li> </ul>

	<ul style="list-style-type: none"> <li>• 해결 아이디어의 단순화</li> <li>- 해결가능한 수준, 핵심요소 추출</li> </ul>
단계 3. 설계	<ul style="list-style-type: none"> <li>• 해결 아이디어의 설계</li> <li>- 해결방안의 시각적 설계 (그림형태, 프리젠테이션 등)</li> <li>- 해결방안의 논리적 설계 (자연어, 순서도 등)</li> <li>• 시각적 설계와 논리적 설계를 모두 포함한 스토리보드 제작</li> </ul>
단계 4. 구현 및 평가	<ul style="list-style-type: none"> <li>• 해결 아이디어의 구현</li> <li>- 설계를 기반으로 프로그램을 구현하기 (프로그래밍 도구 사용)</li> <li>- 디버깅 (교사 및 동료)</li> <li>• 결과 공유 및 시연 (프리젠테이션 등)</li> <li>• 평가 및 피드백 공유</li> <li>- 교사의 관찰 및 동료평가 등</li> </ul>

CT-CPS 수업은 문제를 스스로 발견하는 것으로 시작한다. 이 과정에서는 컴퓨팅 사고력 중 데이터 수집, 분석, 표현의 요소가 활용되며, 문제에 관한 데이터를 수집하여 다뤄보고, 이를 시각적으로 표현하면서 내가 해결해야 할 문제를 더 정확하고 깊이 있게 이해하게 된다. 두 번째로는 그 문제를 해결하기 위한 방법과 아이디어를 다양하게 구상해보고, 이를 추상화하여 표현해본다. 세 번째로는 이 아이디어를 실제화 하기 위한 방법을 설계에 옮겨 본다. 각 장면을 그림이나 글로 묘사해 보거나, 논리적인 요소를 알고리즘화 하여 표현한다. 마지막 단계에서는 프로그래밍 도구를 활용하여 설계된 아이디어를 직접 구현해보고, 동료나 교사와 함께 수정 및 평가하는 과정을 거치게 된다. 전체의 과정은 개인별로 진행할 수도 있고, 팀을 이루어 진행할 수도 있게 된다. 평가의 경우 교사의 과정평가나 발표평가, 동료평가 등을 활용할 수 있으며 최종 결과물을 다른 학생의 결과물과 비교해보면서 스스로 평가를 할 수도 있게 된다.

### 2.3 기존 소프트웨어 교육에서의 수업모형과 CT-CPS 수업모형의 차별성

기존에 소프트웨어 교육과 관련하여 개발된 대표적 수업 모형으로는 한국교육개발원과 한국교육학술정보원(2015)에서 개발한 SW교육 교수·학습모형이 있다. 이 모형은 행동주의, 인지주의, 구

성주의적 관점을 고려하여 학습목표 영역인 지식, 기능, 태도 중 기능 영역을 중심으로 수업모형을 개발하되, 기능 이외에 지식 및 태도 영역이 포함될 수 있도록 개발되었다[24]. 특히 SW교육 교수 학습 모형은 컴퓨팅 사고 신장이라는 목표를 달성하기 위해 각 단계별 활동 내에 컴퓨팅 사고 구성요소(분해, 패턴인식, 추상화, 알고리즘, 프로그래밍)를 포함하도록 구성하는 등의 기본방향을 설정하고 총 5개의 모형을 개발하여 제안하였다. 즉, 컴퓨팅 사고의 구성요소가 모든 학습의 목표이자 기초가 되어야 한다는 점에서 CT요소중심모형(DPAA(P) 모형), 시연중심모형(DMM 모형), 재구성중심모형(UMC 모형), 개발중심모형(DDD 모형), 디자인중심모형(NDIS 모형) 등 5가지로 특성화하여 개발하였다.

이러한 SW교육 교수·학습모형은 컴퓨팅 사고력 신장을 위해 컴퓨팅 사고력의 각 요소에 기반하여 교실에서 수업에 적용할 수 있는 수업 모형이다. 그러나 이 모형에서는 컴퓨팅 사고력의 요소 중 자료 수집, 자료 분석, 자료 표현에 대한 구체적인 과정이 드러나 있지 않으며, 문제를 발견하기 보다는 주어진 문제를 컴퓨팅 사고력을 통해 해결하는 과정을 교수·학습의 과정으로 표현하고 있다.

반면 CT-CPS 수업 모형은 문제의 인식 및 분석 단계에서 자료 수집, 분석, 표현에 대한 다양한 활동을 진행하면서, 스스로 문제를 찾아 컴퓨팅 사고력을 통해 해결하도록 한다는 점에서 그 차별성이 있다고 할 수 있다.

## 3. CT-CPS 모형 기반 수업 프로그램 개발

### 3.1 수업 프로그램 개발의 방향

본 연구에서는 실생활 및 정보교과의 교육과정과 연계 되고, 학습자의 인지적, 정의적 측면에서의 균형 있는 역량강화를 위한 적절한 내용을 구성하고자 하였다.

첫째, 수업의 주제를 현재 및 앞으로의 정보교과 교육과정 및 실생활과 관련 있는 내용으로 선정하고자 하였다. 즉, ‘저작권’과 관련된 소재를 사용하여 이를 알아보고, 이것을 위반하는 것에 대한 심각성을 일깨워 줄 수 있는 방법을 직접 구

현해보는 과정을 수업으로 구성하였다.

둘째, CT-CPS 수업 모형을 활용하여 인지적인 측면과 정의적인 측면에서의 균형 있는 역량 강화를 위한 활동들을 포함하였다. 즉, 인지적인 측면에서는 다양한 내용지식과 구현 방법을 학습하도록 하였으며, 정의적인 측면에서는 이 문제를 왜 해결해야하고, 이것을 해결함으로써 나에게 어떤 변화가 생기는지 생각해 볼 수 있도록 구성하였다.

셋째, 학습자들은 기본적인 소프트웨어 도구(엔트리)의 기본 기능에 대해서 수업을 이미 이수하였기 때문에, 기본 기능에 대한 학습은 최소화하고 이를 활용하여 산출물을 구성해 내는 수업으로 계획하였으며, 수업 내용은 총 4차시로 구성하였는데, 각 차시간의 연계와 학생의 집중도 확보를 위해 1주 2시간씩 총 2주간 진행할 수 있도록 구성하였다.

### 3.2 개발된 수업 프로그램

위와 같은 방향성아래 “저작권 시뮬레이션 제작

을 통한 정보윤리학습”이라는 주제로 수업을 개발하였으며, 전체적인 수업 프로그램은 <표 2> 와 같다.

첫 번째 차시에서는 문제와 관련된 사례를 재미있는 에피소드 형식으로 제시하면서, 수업에 관한 동기를 유발한다. 이 후 자연스럽게, 관련 경험이나 지식이 있는지 간단한 웹 설문을 제시하고, 응답한 결과를 확인하면서 현재 저작권과 관련된 우리 학급의 현황을 파악하고, 본 수업에서 해결해 보아야 할 문제를 명확하게 설정하였다.

두 번째 차시에서는 설정된 문제를 구체적으로 어떻게 해결하고, 구현할 것인지에 관한 아이디어 구상과 설계를 진행하였다. 모듈별로 브레인스토밍을 통해 아이디어를 모으고 이를 구조화하여 표현하였다. 또한 이 아이디어에 관한 전체 토론을 통해 어떤

장면을 표현할 것인지 정하고, 해당 장면을 종이에 단순화하여 직접 그려봄으로써 시각적인 설계를 진행했다. 또한 논리적으로 어떤 단계를 거쳐야 하는지 교사와 함께 알고리즘을 계획하여

<표 2> 본 연구에 적용된 CT-CPS 수업 모형기반의 소프트웨어 수업 설계

차시	CT-CPS 단계	교수·학습 활동	인지적 활동	정의적 활동
1	문제인식 및 분석	<ul style="list-style-type: none"> <li>• 동기부여</li> <li>- 저작권 위배에 관한 엔트리 애니메이션 시청하기</li> </ul>		○
		<ul style="list-style-type: none"> <li>• 문제인식</li> <li>- 우리 반 학생을 대상으로 저작권에 대한 지식과 경험에 관한 설문조사하기</li> <li>- 설문결과 분석을 통해 문제점을 인식하기</li> </ul>	○	○
		<ul style="list-style-type: none"> <li>• 문제화하기</li> <li>- 저작권 위반의 심각성에 관해 일깨워줄 수 있는 시뮬레이션을 제작하기</li> </ul>	○	○
2	아이디어 구상	<ul style="list-style-type: none"> <li>• 컴퓨팅을 활용하여 문제를 해결할 수 있는 다양한 아이디어 구상하기</li> <li>- 모듈별로 시뮬레이션을 만들기 위한 아이디어 토의하기</li> </ul>	○	○
		<ul style="list-style-type: none"> <li>• 해결아이디어를 단순화하기</li> </ul>	○	
	설계	<ul style="list-style-type: none"> <li>• 해결아이디어를 설계하기</li> <li>- 그림으로 해결 아이디어를 표현하기</li> <li>- 논리적인 흐름을 표현하기</li> </ul>	○	
<ul style="list-style-type: none"> <li>• 시각적, 논리적 설계를 포함한 스토리보드 제작하기</li> </ul>		○		
3	구현 및 평가	<ul style="list-style-type: none"> <li>• 해결아이디어를 구현하기</li> <li>- 프로그래밍 도구를 사용하여 설계한 내용을 프로그램으로 구현하기</li> <li>- 선생님, 친구들과 디버깅하기</li> </ul>	○	
<ul style="list-style-type: none"> <li>• 결과 공유 및 시연하기</li> <li>- 모듈별 결과 공유 및 시연하기</li> </ul>		○	○	
<ul style="list-style-type: none"> <li>• 평가 및 피드백 공유하기</li> <li>- 교사 관찰 평가, 동료평가, 자기평가 등</li> </ul>		○	○	

자연어 형태로 표현해 보았다.

세 번째, 네 번째 차시에서는 설계된 아이디어를 엔트리 프로그램으로 직접 구현하였다. 이 과정에서 학생들이 모듈별로 협력하여 작업을 하였으며, 완성 후에는 모듈별로 자신들이 작업한 결과물을 시연하고, 비교하면서 더 수정하거나 개선해야 할 사항에 대해서 스스로 생각해 볼 수 있도록 하였다.

## 4. 연구 방법

### 4.1 연구 대상

본 연구는 CT-CPS 수업모형을 토대로 수업안을 디자인하여 I시의 S 중학교 일반학생 20명을 대상으로 방과 후 소프트웨어 수업 과정에 적용하였다. 적용기간은 2015년 6월 중 2주 동안 총 4차시에 걸쳐서 진행하였으며, 매 주 2시간씩 수업을 진행하였다.

인원 구성은 20명 모두 남학생이었으며, 1학년 9명, 2학년 11명으로 구성되었고, 20명 전원 사전에 엔트리에 대한 학습경험이 있는 것으로 조사되었다.

### 4.2 연구 설계

본 연구는 단일 집단(N=20)내에서 사전, 사후 결과를 비교하는 연구로 수행되었다. <표 3>은 본 연구의 적용을 위한 연구 설계이다.

<표 3> 연구 설계

G1	O1	X1	O2
----	----	----	----

- G1 : 실험 집단
- O1 : 사전검사(창의적문제해결력, 메타인지, 학습동기검사)
- X1 : 수업 적용(CT-CPS기반의 소프트웨어 수업)
- O2 : 사후검사(창의적문제해결력, 메타인지, 학습동기검사, 수업에 대한 의견)

### 4.3 검사도구 및 분석방법

본 연구의 검사도구는 인지적 영역을 대상으로 창의적 문제해결력 검사, 메타인지 검사를, 정의적 영역을 대상으로는 창의적 문제해결력 검사의 일부 영역과 학습동기 검사를 활용하였다.

창의적 문제해결력 검사는 한국교육개발원(2001)에서 발간한 ‘간편 창의적 문제해결력 검사

개발 연구’(2001)를 기반으로 서울대학교 심리 연구실 MI연구팀(2004)에서 개발한 ‘간편 창의적 문제해결력 검사지’를 사용하였다[25]. 이 검사지는 각 하위 요인별로 특정 영역의 지식 및 사고 능력과 기술의 이해 및 숙달, 확산적 사고, 논리·비판적 사고, 동기적 요소로 각각 5문항씩으로 구성된 총 20개의 문항을 측정하도록 되어 있고, Likert 5점 평정척도를 사용하였다.

메타인지 검사는 Jr. MAI Schraw Dennison(1994)이 성인을 대상으로 메타인지를 검사하기 위해 제작한 Metacognitive Awareness Inventory(MAI)를 초등학교 3학년에서 중학교 3학년까지의 학생들에게 적합한 단어로 변환한 질문지를 활용하였다[26]. 본 검사는 인지에 대한 지식에 관한 9문항, 인지에 대한 조절에 관한 11문항으로 구성된 총 20개의 문항을 측정하도록 되어 있고, Likert 5점 평정척도를 사용하였다.

마지막으로 학습동기 검사는 Pintrich와 그의 동료들(1991)이 제작한 학습동기 전략 설문지(MSLQ : Motivation Strategies for Learning Questionnaire)를 강둘래(1996)가 학습동기 요소를 적절히 추출하여 초등학교 수준에 맞게 번안, 수정한 것을 사용하였다. 본 검사지를 사용하기 위해 박도인(2007)은 초등학교 교사 1인과 교육공학 전문가 2인에게 내용에 대한 타당도를 재검증 받았다[27]. 학습동기 검사의 하위요인은 내적목표, 외적목표, 학습신념의 통제, 그리고 자기효능감으로 각각 4문항씩 총 16문항이며, 각 문항은 Likert 5점 평정척도를 사용하였다.

본 연구에서의 분석도구는 IBM SPSS Statistics 21 버전을 사용하여 분석을 실시하였다. 본 연구에서 실험 집단 내의 창의적 문제해결력, 메타인지, 학습동기의 변화를 알아보기 위해 대응표본 t-검정을 실시하였으며, 검정에 사용한 유의 수준은  $p < 0.05$  으로 하여 분석하였다.

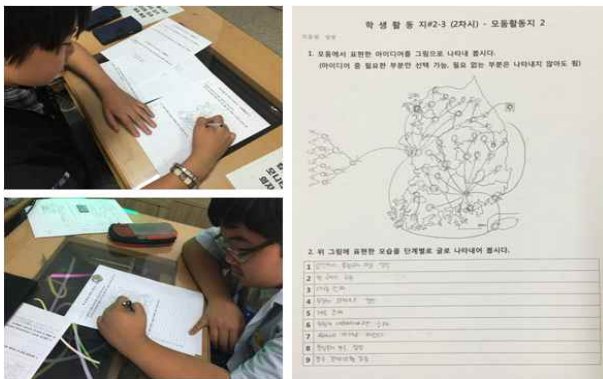
## 5. 연구 결과 및 논의

### 5.1 수업 적용 장면

본 연구에서 적용된 수업의 단계별 장면은 [그림 4], [그림 5], [그림 6] 과 같다.



[그림 4] 저작권에 관한 설문 (1차시)



[그림 5] 아이디어 구상 및 설계 (2차시)



[그림 6] 프로그래밍 예시 결과 (3~4차시)

## 5.2 창의적 문제해결력 검사결과

실험 집단의 창의적 문제해결력 검사결과를 대응표본 t-검정으로 사전, 사후 검사를 비교한 결과는 <표 4> 와 같다.

<표 4> 창의적 문제해결력 검사결과

하위영역	M		SD		t	p
	사전 (N=20)	사후 (N=20)	사전 (N=20)	사후 (N=20)		
특정영역의 지식, 사고기능, 기술의 이해 및 숙달	3.44	3.74	.758	.619	-1.243	.229
확산적 사고	3.35	3.84	.797	.376	-2.901	.009**
논리·비판적 사고	3.81	4.25	.437	.498	-2.643	.016**
동기적 요소	3.66	4.23	.776	.487	-2.833	.011**

(\* $p < .05$ , \*\* $p < .01$ )

실험집단의 창의적 문제해결력 사전, 사후 검사 결과를 분석한 결과 하위 영역 중에서 확산적 사고의 p값이 0.009, 논리·비판적 사고의 p값이 0.016, 동기적 요소의 p값이 0.011로서 유의 수준 0.05 미만에서 창의적 문제해결력의 하위 영역 중, 확산적 사고와 논리·비판적 사고, 동기적 요소가 통계적으로 유의하게 향상되었음을 알 수 있었다. 그러나 특정영역의 지식, 사고기능, 기술의 이해 및 숙달 영역에서는 통계적으로 유의한 향상이 관찰되지 않았다.

## 5.3 메타인지 검사결과

실험 집단의 메타인지 검사결과를 대응표본 t-검정으로 사전, 사후 검사를 비교한 결과는 <표 5> 와 같다.

<표 5> 메타인지 검사결과

하위영역	M		SD		t	p
	사전 (N=20)	사후 (N=20)	사전 (N=20)	사후 (N=20)		
인지에 대한 지식	3.68	4.04	.698	.459	-2.111	.048*
인지에 대한 조절	3.51	3.90	.668	.282	-2.579	.018*

(\* $p < .05$ , \*\* $p < .01$ )

실험집단의 메타인지 사전, 사후 검사결과를 분석한 결과 하위 영역 중에서 인지에 대한 지식의 p값이 0.48, 인지에 대한 조절의 p값이 0.018로서 유의 수준 0.05 미만에서 메타인지의 모든 하위 영역에서 통계적으로 유의하게 향상되었음을 알 수 있었다.

5.4 학습동기 검사결과

실험 집단의 학습동기 검사결과를 대응표본 t-검정으로 사전, 사후 검사를 비교한 결과는 <표 6> 과 같다.

<표 6> 학습동기 검사결과

하위영역	M		SD		t	p
	사전 (N=20)	사후 (N=20)	사전 (N=20)	사후 (N=20)		
내적 목표	3.78	4.24	.877	.462	-2.234	.038*
외적 목표	3.78	4.34	1.015	.635	-2.199	.040*
학습 신념의 통제	3.81	4.30	.811	.368	-2.795	.012*
자기 효능감	3.65	4.28	.817	.512	-3.508	.002**

(\*p < .05, \*\*p < .01)

실험집단의 학습동기 사전, 사후 검사결과를 분석한 결과 하위 영역 중에서 내적 목표의 p값이 0.38, 외적 목표의 p값이 0.040, 학습 신념의 통제의 p값이 0.012, 자기효능감의 p값이 0.002로서 유의 수준 0.05 미만에서 학습동기의 모든 하위 영역에서 통계적으로 유의하게 향상되었음을 알 수 있었다. 특히 자기효능감 영역의 p값은 0.002로서 매우 높은 차이를 보였다.

5.5 수업에 대한 학생의 의견

실험 집단의 수업이 적용된 후, 수업에 참가한 학생들을 대상으로 수업에 대한 의견을 4점 척도 객관형 문항과 개방형 주관식 문항으로 조사하였으며, 이에 대한 결과를 정리한 내용은 <표 7>, <표 8>과 같다.

<표 7> 수업에 대한 학생 의견 (객관식, N=20)

질 문	매우 그렇지 않다	그렇지 않다	그렇다	매우 그렇다
불법 복제 시뮬레이션 수업을 통해 새롭게 배운 것이 있는가?	0 (0%)	0 (0%)	11 (55%)	9 (45%)
불법 복제 시뮬레이션 수업이 재미있었는가?	0 (0%)	2 (10%)	7 (35%)	11 (55%)
불법 복제 시뮬레이션처럼 일상생활의 다른 문제도 프로그래밍으로 해결할 수 있을까?	0 (0%)	0 (0%)	3 (15%)	17 (85%)
불법 복제 시뮬레이션으로 구성된 4차시 수업을 정보교과 시간에 적용할 수 있을 것이라 생각하는가?	0 (0%)	1 (5%)	9 (45%)	10 (50%)
불법 복제 시뮬레이션 수업을 통해 정보교과에 대한 이미지가 긍정적으로 변화했다고 생각하는가?	0 (0%)	0 (0%)	5 (25%)	15 (75%)

<표 8> 수업에 대한 학생 의견 (주관식, N=20)

질 문	내 용
불법복제 시뮬레이션 수업과 기존의 정보교과 수업의 차이점이 있다면?	- 활동이 많고 재미있다. - 책을 보지 않고 문제를 직접 찾아 해결한다. - 직접 시뮬레이션을 만들어서 확인한다. - 내가 주도적으로 참여해서 문제를 해결한다. - 조금 더 어려운 것 같다. 등
수업에서 개선해야 할 부분이 있다면?	- 프로그래밍 하는 시간이 조금 부족했다. - 더 많은 것을 해보고 싶다. 등
불법 복제 시뮬레이션 수업을 참여하고 난 후 달라진 점이 있다면?	- 프로그래밍 언어를 활용하여 저작권 문제에 대한 해결을 시도해볼 수 있다는 것이 신기했다. - 내가 스스로 문제를 찾고, 그림과 문제해결 절차로 해결과정을 표현해 보는 것이 재미있었다. - 불법복제가 생각보다 심각한 문제라는 것을 알게되었다. 등

수업에 대한 학생의 의견에 대해 조사한 결과를 살펴보면, 모든(100%) 학생이 ‘본 수업을 통해 새롭게 배운 것이 있는가’ 라는 질문에 ‘그렇다’ 이상의 응답을 했다. 또한 2명의 학생을 제외한 모든(90%) 학생이 ‘수업이 재미있는가’ 라는 질문에 ‘그렇다’ 이상으로 응답했으며, 모든(100%)학생이 ‘프로그래밍을 통해 일상생활의 다른 문제를 해결할 수 있을 것인가’라는 질문에 ‘그렇다’ 이상의 응답을 했다. 한편 본 수업의 정보교과 적용 가능성에 대해 1명을 제외한 모든(95%) 학생들이 ‘그렇다’ 이상의 응답을 했으며, 모든(100%) 학생이 ‘본 수업을 통해 정보교과에 대한 이미지가 긍정적으로 변화했는가’ 라는 질문에 ‘그렇다’ 이상의 응답을 했다.

주관식 응답에서는 기존 정보교과의 수업과 차이점에 대해 ‘활동이 많음’, ‘문제를 직접 찾고 해결함’ 등의 응답을 보였으며, 수업 후 달라진 점에 대해서는 ‘프로그래밍으로 저작권 문제를 해결하는 것이 신기함’, ‘그림과 문제해결 절차로 해결과정을 표현해 보는 것이 재미있음’, ‘불법복제가 생각보다 심각한 문제라는 것을 알게 됨’ 등의 응답을 했다.



5.6 결과 해석 및 논의

실험집단의 세 가지 검사의 사전, 사후 결과를 비교, 분석한 결과와 각 하위 영역의 인지적, 정의적 영역 해당 여부를 요약하면 <표 9> 와 같다.

<표 9> 검사 결과 종합

검사	하위 영역	통계적 유의도 관찰 여부	인지적/ 정의적 영역
창의적 문제해결력	특정영역의 지식, 사고기능, 기술의 이해 및 숙달	×	인지적
	확산적 사고	○**	
	논리·비판적 사고	○	
	동기적 요소	○	정의적
메타인지	인지에 대한 지식	○	인지적
	인지에 대한 조절	○	
학습동기	내적 목표	○	정의적
	외적 목표	○	
	학습 신념의 통제	○	
	자기 효능감	○**	

(\*\* :  $p < .01$ )

위 결과를 해석해 보면, 세 가지 검사의 하위 영역 중에서 특정영역의 지식, 사고기능, 기술의 이해 및 숙달영역의 제외한 모든 영역에서 유의한 향상이 관찰되었는데, 이 중 확산적 사고와 자기효능감 영역은 유의 확률이 0.01 미만으로 매우 높은 향상도를 보였다. 종합적으로 보면 CT-CPS 수업모형을 기반으로 설계된 본 수업의 적용이 지식적인 내용의 전달보다는 문제를 스스로 찾고 확산적인 사고를 통해 문제를 창의적으로 해결하는 방법을 찾아가는 수업이었음을 알 수 있었다. 또한 이러한 과정을 자기 주도적으로 진행해 가면서 내가 알고 있는 것이 무엇이고 어떻게 과정을 진행해야 하는지를 생각할 기회가 부여 되었고, 이를 통해 이것을 왜 공부해야하는지에 대한 학습동기를 향상할 수 있었음을 알 수 있었다. 또한 과제 해결과정에 대한 성공 경험을 통해 자기 효능감이 높게 향상되었음을 알 수 있었다.

또한 수업에 대한 학생의 의견에 대해 조사한 결과를 해석해보면 인지적 측면과 정의적 측면에서 학생들이 긍정적으로 변화한 것을 관찰할 수 있었으며, 나아가 본 연구에서 적용된 형태의 수업이 정보교과 시간에 적용될 수 있는 가능성도 확인할 수 있었다.

6. 결론 및 제언

본 연구는 중학생의 소프트웨어 교육에서 학생들의 인지적, 정의적 역량을 균형 있게 함양하기 위해 개발된 CT-CPS 수업모형의 효과성을 분석하고자 수행된 연구로서, CT-CPS 수업모형을 토대로 수업을 디자인하여 I시의 S 중학교 방과 후 소프트웨어 수업 과정에 적용하였고, 이 후 학생들의 인지적 측면과 정의적 측면에서의 효과성을 분석하였다.

적용 및 분석 결과 검사의 인지적 영역과 정의적 영역의 대부분 영역에서 통계적으로 유의한 향상을 관찰할 수 있었으며 본 수업 모형을 적용한 수업이 학생들의 인지적 정의적 역량 향상에 효과가 있음을 알 수 있었다. 그러나 결과의 해석 및 논의 부분에서 논한 바와 같이 본 수업모형은 기초 지식을 함양하는 소프트웨어 수업보다는 기초적인 지식과 내용에 대한 학습이 이루어진 이후, 내가 다룰 수 있는 소프트웨어를 활용하여 창의적으로 문제를 해결해가는 수업에 적용하는 것이 더 효과적이다.

본 연구는 소집단(20명, 1개 집단)을 대상으로 4차시의 수업을 적용한 후 연구 결과를 도출하였으므로 이를 전체 중학생들을 대상으로 일반화하여 적용하는데 한계점이 존재할 수 있다.

2015 개정 교육과정에서 추구하는 소프트웨어 교육의 올바른 목적과 취지를 올바르게 현실화하기 위해서는 소프트웨어 교육이 올바른 방향과 방법을 취해야 하며, 이를 위한 다양한 연구와 노력이 필요하다. 본 연구자는 이 연구가 다양한 분야에서 이루어지는 소프트웨어 교육의 방법 탐색에 관한 연구의 기초자료로 활용되길 바라며, 또한 많은 연구자와 교사가 소프트웨어 교육을 위한 올바른 방법을 제안하는 다양한 연구를 수행하길 기대한다.

참고 문헌

[1] Nobel Prizes and Laureates(2014.2.12). The Nobel Prize in Chemistry 2013. [http://www.nobelprize.org/nobel\\_prizes/chemistry/laureates/2013/](http://www.nobelprize.org/nobel_prizes/chemistry/laureates/2013/)

- [ 2 ] 아시아경제(2012.11.26). ‘위기는 기회’ 강조하는 BMW 노르베르트 라이트호퍼 CEO. <http://www.asiae.co.kr/news/view.htm?idxno=2012112517345422670>
- [ 3 ] MK 뉴스(20140624). 고급SW, 정부주도 개발로 경쟁력 키워야. <http://news.mk.co.kr/newsRead.php?year=2014&no=917922>
- [ 4 ] J. M. Wing.(2006). Computational Thinking. *Communication of the ACM*, 49(3), 33-35.
- [ 5 ] J. M. Wing.(2008). Computational Thinking and thinking about computing. *Philosophical transactions of the Royal Society A*, 366, 3717-3725.
- [ 6 ] V. Barr and C. Stephenson.(2011). Bringing Computational Thinking to K-12: What is Involved and What is the Role of the Computer Science Education Community?. *ACM Inroads*, 2, 48-54.
- [ 7 ] Philip Guo(2011). What is Computer Science? Efficiently Implementing Automated Abstractions.<http://www.pgbovine.net/what-is-computer-science.htm>
- [ 8 ] The CSTA Standard Task Force(2011). *CSTA K-12 Computer Science Standards*. Revised 2011, 1-73.
- [ 9 ] Department for Education(2013). *The national curriculum in England*. Framework document.
- [ 10 ] 교육부(2015). 소프트웨어 교육 운영 지침.
- [ 11 ] 교육부(2015). 2015 개정 정보 교육과정.
- [ 12 ] Google for Education. (2015). Exploring Computational Thinking. <https://www.google.com/edu/resources/programs/exploring-computational-thinking/index.html#!ct-overview>
- [ 13 ] Microsoft Education. (2015). Learn to code.<https://www.microsoft.com/en-us/education/students/code/default.aspx>
- [ 14 ] 삼성전자 주니어 소프트웨어 아카데미(2014). <http://www.juniorsw.org/>
- [ 15 ] 네이버 소프트웨어야 놀자 (2017). <http://www.playsw.or.kr/main>
- [ 16 ] 교육부 보도자료 (2015. 7. 22). 교육부·미래부, SW중심사회를 위한 인재양성 추진계획 발표.
- [ 17 ] 교육부 보도자료(2016. 2. 11). 선도학교 확대를 통한 소프트웨어 교육 본격 시작.
- [ 18 ] 전용주, 김태영(2014). 국내 외 동향 분석을 통한 SW교육의 이해. **한국컴퓨터교육학회 학술발표대회 논문집**. 18(2), 137-142.
- [ 19 ] 전용주, 김태영(2015). 창의,융합적 SW교육을 위한 CT-CPS 프레임워크 개발 기초 연구. **한국컴퓨터교육학회 학술발표대회 논문집**. 19(1), 37-42.
- [ 20 ] Jeon & Kim(2015), The Development of the CT-CPS Framework for Creative and Integrative Software Education. *Korean Journal of Teacher Education*. 31(3), 67-88.
- [ 21 ] 전용주(2017). 새로운 교육과정의 소프트웨어 교육을 위한 컴퓨팅 사고력 기반 창의적 문제해결(CT-CPS) 수업모형의 개발 및 적용. 박사학위논문, 한국교원대학교 대학원.
- [ 22 ] 김지현, 김태영(2016). 중등 수학과학 영역을 위한 피지컬컴퓨팅 교육이 융합적 역량 향상에 미치는 영향. **한국컴퓨터교육학회 논문지**. 19(2), 87-98.
- [ 23 ] 김병조, 전용주, 김지현, 홍창의, 김태영(2014). Computational Thinking에 기반한 IT융합형 인재상에 관한 연구. **한국컴퓨터교육학회 학술발표대회 논문집**. 18(1), 27-33.
- [ 24 ] 한국교육개발원, 한국교육학술정보원(2015). **SW교육 교수학습 모형 개발 연구**.
- [ 25 ] 구재훈(2016). 초등 정보과학영재의 창의적 문제해결력 향상을 위한 융합형 CT-CPS 프레임워크 기반 수업 콘텐츠 개발 및 적용. 석사학위논문, 한국교원대학교 대학원.
- [ 26 ] 안상진(2013). 피코보드를 활용한 프로그래밍 학습이 중학생의 프로그래밍 흥미와 메타인지에 미치는 영향. 석사학위논문, 한국교원대학교 대학원.
- [ 27 ] 박도인(2007). 창의적 문제해결모형 기반 초등학교 사회과 수업에서 학습동기, 메타인지, 창의적 문제해결력의 관계. 석사학위논문, 이화여자대학교 대학원.



## 전 용 주

2005 한국교원대학교  
컴퓨터교육과(교육학학사)

2014 한국교원대학교  
컴퓨터교육과(교육학석사)

2017 한국교원대학교 컴퓨터교육과(교육학박사)

2017 청주 사직초등학교 교사

관심분야: 소프트웨어 교육, 정보영재 교육, 컴퓨  
팅 사고력, 웹 프로그래밍

E-Mail: yyongju@naver.com



## 김 태 영

1985 한양대학교  
산업공학과(이학사)

1990 Texas A&M University  
컴퓨터과학과(석사)

1994 Texas A&M University 컴퓨터과학과(박사)

1994~ 현재 한국교원대학교 컴퓨터교육과 교수

관심분야: 컴퓨터교육, 데이터베이스, 프로그래밍

E-Mail: tykim@knue.ac.kr