

# 플립드 러닝 기반 소프트웨어 교육에서 초등학생의 흥미도와 컴퓨팅 사고력에 영향을 미치는 요인에 관한 질적 연구\*

임경희\*\* · 신종호\*\*\*

성균관대학교\*\* · 아주대학교\*\*\*

## 요약

본 연구의 목적은 플립드 러닝 기반 소프트웨어 교육이 초등학생의 흥미도와 컴퓨팅 사고력에 미치는 영향 요인을 찾아 효과적인 교수학습 전략을 제안하는 것이다. 이를 위해 초등학교 학생들을 대상으로 플립드 러닝 기반 소프트웨어 교육프로그램을 개발하여 수업에 적용하고, 학습과정에서 발생한 산출물과 관찰 및 인터뷰 자료를 바탕으로 질적 연구를 수행하였다. 연구 결과, 초등학생의 플립드 러닝 기반 소프트웨어 교육에서 흥미도에 영향을 미치는 요인은 '수업에 대한 기대', '실제적 과제', '과제에 대한 성취', '동료와의 상호작용'으로 나타났다. 컴퓨팅 사고력을 높이는 요인으로는 주어진 수업시간 안에서의 '과제 완료로 인한 성취감', '유미미한 동료와의 상호작용', '교사의 핵심적인 피드백'으로 나타났다. 이러한 요인을 바탕으로 소프트웨어 교육에서 흥미를 높이고 컴퓨팅 사고력을 증진시키기 위하여 수업단계별로 적용할 수 있는 교수학습 전략을 제안하였다.

키워드 : 플립드 러닝, 소프트웨어 교육, 학습 흥미도, 컴퓨팅 사고력, 질적연구

## A Qualitative Research on Influential Factors of Software Education based Flipped Learning on Elementary Students' Interest and Computational Thinking\*

Kyunghee Lim\*\* · Jongho Shin\*\*\*

SungKyunKwan University\*\* · Ajou University\*\*\*

## ABSTRACT

The purpose of this study was to propose an effective teaching and learning model by investigating the influencing factors of elementary school students' interest and computational thinking in software education based on flipped learning. To accomplish the purpose of the study, we developed a software education program based on flipped learning for elementary school students and applied qualitative research based on the interviews with the students and outputs from the learning process. The results of this study, First, factors affecting interest in the elementary school students' software education based on flipped learning were 'the expectation of class', 'authentic task', 'the accomplishment of task' and 'interaction with peers'. Second, the factor of enhancing computational thinking was 'the accomplishment of task', 'interaction with peers', and 'the teacher's meaningful feedback'.

Keywords : Flipped Learning, Software Education, Academic interest, Computational Thinking, Qualitative research

\* 이 연구는 임경희(2019)의 석사학위 논문을 수정·보완한 것임.

교신저자 : 신종호(아주대학교)

논문투고 : 2019-07-03

논문심사 : 2019-07-11

심사완료 : 2019-08-14

## 1. 서론

소프트웨어의 가치와 중요성이 높아짐에 따라 우리나라에서도 2015개정 교육과정을 마련하여 초등학교에서 고등학교까지 소프트웨어 교육을 의무적으로 이수하도록 하고 있다. 그러나 소프트웨어 교육을 위한 여건이나 기반이 충분히 마련되지 않은 상황에서 소프트웨어 교육이 학교 현장에 확산되면서 여러 가지 문제점도 함께 발생하고 있다[3][9][18].

소프트웨어 교육의 성취 목표로 내세운 핵심 역량은 정보문화소양, 컴퓨팅 사고력, 협력적 문제해결력이며, 이 중에서 컴퓨팅 사고력(Computational Thinking)은 국내 SW교육의 목표로 '컴퓨팅 사고력을 가진 창의·융합 인재' 양성으로 지정될 정도로 관심을 받아 왔다[15]. 그러나 현재 소프트웨어 교육을 위해 이루어지는 대부분 수업은 교사 중심의 전달식 강의로 진행되고 있으며, 이러한 수업 형태는 정보를 전달하는 다른 교과에서는 효과적일 수 있으나 컴퓨팅 사고력의 증진을 주요 목표로 하는 소프트웨어 교육에는 도움이 되지 못한다. 특히, 초등교육 현장에서 소프트웨어 교육은 단순히 코딩이나 프로그래밍 방법, 프로그래밍의 개념이나 문제 해결과정의 반복 위주로 진행되고 있어 컴퓨팅 사고력을 길러 주기에는 미흡하며, 컴퓨터 프로그래밍 자체에 싫증을 내거나 흥미를 잃는 학생들이 발생하고 있다는 문제점이 제기되고 있다[18][22].

또한, 현재 우리나라에서 진행되고 있는 2015개정 교육과정의 초·중등 소프트웨어 교육 강화 정책에서 공통적으로 제기되고 있는 문제점 중의 하나는 소프트웨어 교육에 대한 수업 시수 부족이다[3][9][22]. 컴퓨팅 사고력은 컴퓨터과학을 탐구하는 영역이 아니고, 생활에서 부딪히는 문제를 컴퓨터과학자처럼 해결하기 위한 기본적인 능력으로[4], 컴퓨팅 사고력 성취는 오랜 기간에 걸쳐 많은 양의 학습시간을 필요로 하고 학생들에게 인내심을 요구하는 과정이다. 그러나 초등학교의 경우 소프트웨어 교육을 필수로 이수하도록 하고 이전보다 단 5시간이 늘어나 17시간 이상 이수하도록 규정되어 있지만, 현장의 교사들은 이 시수로는 체계적이고 충실한 소프트웨어 교육이 이루어질 수 없다고 주장하고 있다[3].

이와 같이 교사의 지식전달 중심의 강의식 수업, 프로그래밍 개념이나 코딩 위주의 수업, 부족한 수업 시수 등 소프트웨어 교육에서 발생하고 있는 문제 해결 방안으로

논의되고 있는 것이 플립드 러닝(flipped learning)이다[9]. 플립드 러닝은 강의나 직접적인 교수법을 없애는 것이 아니라 교사와 개별 학생간의 중재활동과 학생들 간의 협력 활동 시간을 최대화하는 것이며, 다루고자 하는 학습 내용의 희생 없이 학생들의 적극적이고 능동적인 수업참여를 이끌어 낼 수 있기 때문이다[5]. 이에 따라 초등학생을 대상으로 플립드 러닝을 적용한 소프트웨어 교수학습 모델을 개발하고, 이를 바탕으로 소프트웨어 교육에 대한 학습자 태도 및 만족도, 컴퓨팅 사고력 등 효과성을 검증하는 연구들이 다수 수행되어 왔다[6][10][20][26].

그러나 초등학교에서의 소프트웨어 교육의 역사가 짧은 것을 감안할 때 소프트웨어 교육의 효과성을 종합적으로 논의하기에는 시기상조이며, 그 효과 또한 학습자 수준에 따라 일관된 결과가 도출되지 않았다[22][23]. 또한 플립드 러닝을 적용한 소프트웨어 교육의 효과성과 관련하여 이루어진 연구들은 대부분 사전·사후 검증을 통한 양적인 결과만 강조할 뿐, 플립드 러닝에 참여한 개인 학습자들의 흥미가 왜 높아지는지, 어떤 과정을 통해 컴퓨팅 사고력이 향상되는지 등 소프트웨어 교육이 이루어지는 학습과정 측면에 대한 분석은 매우 제한적이다. 이에 국내 소프트웨어 교육 연구 동향을 분석한 연구에서는 SW교육 관련 교육 상황에서의 사례분석이나 질적 경험을 드러내는 질적 연구나 혼합연구는 미흡함을 지적하고 있다[17]. 이러한 측면에서 소프트웨어 교육이 이루어지고 있는 교수학습 상황에서 실제 어떤 활동들이 일어나고 있는지를 질적 접근을 통해 심층적으로 살펴보는 것이 필요하다[22].

그동안 양적연구에서 제시한 효과성에 비추어 플립드 러닝 기반의 소프트웨어 교육의 학습 과정에 참가하는 초등학생들의 흥미가 높아지는 요인은 무엇이며, 무엇이 컴퓨팅사고력을 향상시키는지에 대한 분석은 보다 효과적인 소프트웨어 교육을 위한 교수학습 전략 마련에 있어 무엇보다도 중요하다. 이에 본 연구에서는 질적연구를 통하여 플립드 러닝 기반 소프트웨어 교육에서 이루어지는 학습자들의 학습경험을 탐색함으로써 양적연구에서 놓치거나 간과할 수 있는 현상을 보다 심도 깊게 이해하고 그 요인들의 관계나 작용도 이해하고자 하였다. 이를 통해 구체적으로 초등 소프트웨어 교육을 위한 플립드 러닝 기반의 교수학습 과정에 내재되어 있는 흥미 유발 요인과 컴퓨팅 사고력 향상 요인을 찾아보고 이를 통해 보다 효과적인 소프트웨어 교육을 위한 교수

학습전략을 제안하고자 한다.

## 2. 이론적 배경

### 2.1. 플립드 러닝

플립드 러닝은 교육 테크놀로지를 활용하여 지식 전달 중심의 강의식 수업 방식을 ‘배움’중심의 수업으로 수업의 형식을 바꾼 ‘교육모형’이다[3]. 플립드 러닝은 2000년대 초반부터 전 세계적으로 매우 활발하게 활용되고 있는 교실 수업 방식으로, 수업시간의 유연한 활용 가능성과 다양한 테크놀로지의 효과적 적용 측면에서 플립드 러닝의 교육적 유용성이 강조되어왔으며, 구체적으로 학생들의 학업 성취 및 학습자와 교수자의 의욕적 측면에서의 긍정적인 결과들이 보고되고 있다[27].

플립드 러닝의 일반적인 특징은 크게 네 가지로 나눌 수 있다[1][20]. 첫째, 플립드 러닝에서 학생들은 사전 제작된 동영상을 통해 핵심 개념을 미리 학습하게 되므로 교실에서의 교사는 핵심 내용의 전달이 아닌 그 내용을 바탕으로 하는 심화·응용학습을 촉진하고 도와주는 역할을 하게 된다. 둘째, 강의실 안에서는 학습의 개념을 이해한 학습자들이 프로젝트 기반 학습, 문제 기반 학습, 사례연구, 실험, 게임, 시뮬레이션, 퀴즈 등의 학습 활동을 수행함으로써 이해를 넘어 활용과 적용을 통한 심화학습에 참여한다. 셋째, 선수지식을 통해 학습의 개념을 습득한 학습자는 다양한 학습활동을 통해 능동적인 학습을 수행하게 되고 교수자의 조언 아래 학습자의 수준에 따라 조정된 개인적, 맞춤형 학습이 가능하게 된다. 넷째, 교수자와 학습자, 학습자와 학습자간의 상호작용이 강조되며, 교실 수업은 서로의 의견을 나누고 공동의 과제를 해결하는 장으로 활용된다. 이러한 특징에 비추어 여러 연구에서 소프트웨어 교육방법은 소프트웨어 관련 주요 내용들을 동영상 자료로 학습하고 교실 수업에서 다양한 활동을 통해 관련 내용을 이해하는 플립드 러닝 방법이 유용하다고 제안되어 왔다[8][12].

### 2.2. 학습 흥미도

소프트웨어 교육 이수가 의무화되면서 모든 학교에서

소프트웨어 교육이 실현되고 있지만 수업내용 및 수업방법에 따라 소프트웨어 교육에 대한 학생들의 흥미도가 부족하다는 문제가 제기되고 있다. 이에 학생들의 흥미와 관심을 높이기 위하여 교육용 로봇 등의 교구를 활용하거나[11][13][19][29], 협동학습이나 플립드러닝 등 새로운 교육방법을 적용하고[6][7][20][30], 그에 대한 효과성을 분석한 연구들이 수행되고 있다. 연구결과 이러한 새로운 교육적 시도들은 대체적으로 학생들의 학습 흥미도에 긍정적인 영향을 미치는 것으로 나타났다. 그러나 소프트웨어 교육에서 교육용 로봇 등 교구를 활용하여 학생들의 흥미도를 높이고자 한 연구들은 대부분 사전·사후 검증을 통해 흥미도에 유의미한 차이가 났다는 통계적인 결과만을 강조할 뿐 왜 그러한 효과가 발생했는지에 대한 구체적이고 깊이 있는 설명은 제한적이다. 또한, 플립드 러닝을 적용한 연구의 경우에는 일방적인 강의식 수업에 비해 흥미도가 높아졌다는 결과를 제시하는 연구[20]도 있지만 오히려 흥미도와 지속력이 떨어지거나[30], 수업 흥미도 요인에 대해서는 플립드 러닝의 효과가 두드러지지 않는다는 결과를 제시한 연구도[6] 있다.

이러한 선행연구 결과와 초등학교 소프트웨어 교육의 짧은 역사를 고려할 때 학습자의 흥미에 영향을 미치는 구체적인 요인을 분석하고 이를 통해 구체적인 교수학습 전략의 개발을 위한 기초적인 연구가 필요하다고 할 수 있다.

### 2.3. 컴퓨팅 사고력

컴퓨팅 사고력이란 “문제를 접했을 때, 컴퓨터 과학자처럼 문제를 해석하고, 문제해결 과정을 설계하며, 이를 바탕으로 인간의 행동을 이해하는 접근방식”으로 [23], 21세기를 살아가는 대부분의 사람들이 갖추어야 할 기본 능력이라고 지적한다[32]. 컴퓨팅 사고력의 중요성이 증대됨에 따라 컴퓨팅 사고력 신장을 위한 소프트웨어교육에 대한 관심이 높아지고 있으며, 교육 효과성에 대한 논의도 많이 이루어지고 있다.

국내 문헌 분석에서 나타난 컴퓨팅 사고력 증진을 위하여 소프트웨어 교육에서 활용되고 있는 교수 접근은 문제해결적 접근, 협력적 상호작용 접근, 성찰기반 접근, 놀이중심 발견 학습 및 디자인 접근이 활용되고 있는 것으로 분석되었다[2]. 동시에 소프트웨어 교육이 컴퓨팅 사고력 향상에 미치는 효과에 대한 연구도 꾸준히 수행

되고 있다. 컴퓨팅 사고력 향상을 위한 국내 소프트웨어 교육에 대한 메타 분석에 의하면 SW 교육을 실시한 집단이 실시하지 않은 통제 집단에 비하여 컴퓨팅 사고력이 높은 것으로 나타났다[23]. 하지만 교수학습 과정에서 어떠한 요인이 컴퓨팅 사고력을 향상시켰는지 명확한 요인에 대한 설명은 매우 제한적이다. 컴퓨팅 사고력은 구성요소가 다면성을 가지고 있으므로 단일 방법의 평가로 컴퓨팅 사고력을 모두 설명하기 어렵다는 점을 인식하고 이를 측정하기 위한 다양한 방법들이 활용될 필요가 있다[2]. 특히, Shute 등은 컴퓨팅 사고력 측정을 위하여 구인의 속성, 학습자의 인지적·정서적 특성, 교수-학습적인 맥락을 복합적으로 고려해야 한다고 주장하였다[23][28].

### 3. 연구방법

#### 3.1. 질적 연구의 필요성

선행연구에서 살펴본 바와 같이 소프트웨어 교육에 대한 문제해결 및 효과성을 높이기 위하여 플립드 러닝 기반 소프트웨어 교육을 실행하고 이를 바탕으로 소프트웨어에 대한 인식 조사와 흥미도, 학습동기 및 만족도, 컴퓨팅 사고력 등 효과성에 대하여 분석하고 있다 [4][6][7][24][25][26][31]. 이러한 선행연구들은 소프트웨어 중심 사회로 발전해 감에 따라 새롭게 시작된 소프트웨어 교육에 대한 교육당사자들의 관심을 불러일으키는데 기여한 바 있다. 그러나 이러한 선행연구들은 대부분 실험연구를 통해 소프트웨어 교육에서 학습자의 흥미 및 컴퓨팅 사고력에 대한 효과성 관련 통계적 자료를 제공하고는 있지만, 실제적으로 플립드 러닝 기반 소프트웨어 교육이 어떻게 학생들의 흥미와 컴퓨팅 사고력 향상에 도움을 주는지에 대한 실제적이고 심층적인 이해와 설명을 제공해 주지 못하고 있다. 학습 흥미도는 소프트웨어 교육과정에서 학습자의 주의집중과 학습 지속에 영향을 미치며, 컴퓨팅 사고력은 인지 과정이기 때문에 문제를 찾고, 이해하고, 분석하여 해결하는 절차적 과정 단계에서 학생들의 인지적 처리와 정서적 반응을 바탕으로 피드백하기 위한 미시적 도움 자료가 요구된다. 하지만 이와 같은 교실에서의 활동 양상에 대한 탐색은 양적 연구로는 불가능하므로 이를 위해서는 관찰 연구나 질적

연구 등의 방법이 활용될 필요가 있다[17][22].

이에 본 연구에서는 플립드 러닝 기반의 소프트웨어 교육과정에서 학생들의 학습경험을 심층적으로 기술하여 이해하고, 그러한 총체적 맥락 속에 나타나는 학생들의 흥미를 증가시키고, 컴퓨팅 사고력을 향상시키는 요인들이 무엇인지 분석하는 질적 연구를 수행하고자 하였다. 또한 이를 통해 효과적인 소프트웨어 교육을 위한 보다 구체적인 교수학습 전략을 도출해 내고자 하였다.

#### 3.2. 플립드 러닝 기반 소프트웨어 교육 프로그램 개발 및 적용

##### 3.2.1. 수업 모형 설계

본 연구에 적용된 플립드 러닝 기반 소프트웨어 교육 수업 모형은 선행연구에서 제안된 플립드 러닝 모형을 참조하여 <Table 1>과 같이 개발하였다[14][16].

<Table 1> Instructional model of Software Education based on Flipped learning

단계	활동	내용
사전 학습	디딤 수업 시청	- 주요 개념과 기능을 습득을 위한 디딤 수업 동영상 시청
	맷글 달기와 퀴즈 풀기	- 질문하기, 딸림 문제 풀기
준비도 확인 및 학습 과제 안내	수업 준비도 확인	- 퀴즈에 대한 정답 확인 및 부족 부분에 대한 피드백
	학습 과제 안내	- 해결해야 할 과제 안내
과제 수행 및 해결	과제 분석	- 과제 해결을 위한 문제 분석(모둠 활동)
	과제 수행	- 과제 해결을 위한 코딩 및 디버깅 (개별활동)
	상호 검토	- 상호 검토를 통한 정교화(모둠활동)
정리 및 평가	공유 및 과제	- 서로의 결과를 공유 및 핵심 개념 과제
	활동 평가	- 활동 결과물을 평가하기

‘사전 학습’ 단계는 가정에서 학생이 개별적 자율학습 방법으로 디딤 동영상을 시청하는 단계이다. 학생들은 디딤 수업 동영상을 시청함으로써 동영상에 포함된 학습내용 및 주요 개념을 인지학습 한다. 디딤 영상 수업을 시청한 후 학생들 간의 상호작용과 교사와의 소통을

위하여 웹 페이지에 댓글 달기를 기능을 이용하여 학생의 느낌이나 질문 사항, 의견 제시 등을 할 수 있다. 또 디딤 영상 시청에 따르는 간단한 테스트 문제를 풀어야 한다. 이 문제의 정답률을 참고하여 학습자는 스스로 자신의 학습을 점검하여 반복적으로 영상을 시청할 수 있고 교사도 학생들의 학습 정도를 파악하여 수업에서 피드백 자료로 활용할 수 있다.

‘준비도 확인 및 학습 과제 안내’ 단계는 교실에서의 활동 수업 단계로서 교사가 학생들에게 디딤 수업 영상을 통해 학습한 주요 개념의 인지를 확인하는 질문과 피드백을 한다. 또한 수업의 목표와 목표 성취를 위한 활동 과제가 제시되고 과제 수행 방법이 안내된다.

‘과제 수행 및 해결’ 단계에서는 우선 과제로 주어진 문제를 분석하기 위해 모둠끼리 자료를 수집하고 탐구하여 분석하는 활동을 한다. 문제 분석이 끝나면 문제 해결을 위해 스크래치를 이용하여 사전 디딤 동영상 학습을 통해 익힌 기능을 활용한 프로그래밍을 한다. 프로그래밍 과정은 계속적인 테스트와 디버깅 활동으로 이루어진다. 학생들은 이 활동 안에서 동료와의 상호작용 및 모둠원 간의 활발한 토의를 하며 과제의 완성도를 높여 간다.

‘정리 및 평가’ 단계에서는 서로의 다양한 산출물을 발표하고 평가하여 다른 사람의 아이디어나 코딩 방법 등을 공유하는 기회를 가지므로써 소프트웨어에 대한 높은 안목을 갖출 수 있도록 한다.

이와 같이 플립드 러닝 기반 소프트웨어교육 수업 모형은 총 4단계로 구성되어 있으며 수업 상황 및 교육 여건에 따라 조금씩 변형하거나 보충하여 본 수업에 적용하였다.

### 3.2.2 교육 프로그램 설계

플립드 러닝을 활용한 스크래치 프로그래밍 교육을 위한 교육 내용을 <Table 2>와 같이 선정하였다. 초등학생들의 인지수준에 맞도록 실제적 생활과 밀접한 주제로 가상 물체와 실제적 사물 로봇의 동작을 제어하며 프로그래밍 학습을 할 수 있도록 구성하였다. 두 차시분 에 걸친 학습에서 성취 항목이 같도록 주제를 선정하여 홀수 차시에는 스크래치 블록 코딩 소프트웨어만 사용하고, 짝수 차시에는 로봇을 추가하여 코딩 명령블록이 실제 사물에 적용되어 실행되는 모습을 직접 관찰할 수 있도록 했다.

<Table 2> Education program design

순서	주제	관련 교과	교육 내용
1차시	애니메이션 만들기	국어	- 순차구조의 이해 - 멀티미디어 요소
2차시	햄스터 로봇 탐구	과학	- 순차구조의 활용 - 직진, 회전, 바퀴 제어
3차시	피코와 얼음 땀 놀이	정보	- 조건 제어문 이해
4차시	사거리 통과하기	정보	- 조건 제어문 활용
5차시	패턴 만들기	미술	- 반복문 이해 - 암호와 보안의 이해
6차시	조난 구조 보내기	미술	- 반복문 활용 - LED 빛의 원리
7차시	악기 없이 연주하기	음악	- 디지털 음악 이해
8차시	햄스터 그룹사운드	체육	- 디지털 음악 활용
9차시	춤추기 대회	체육	- 함수의 이해 - 메시지 전달
10차시	햄스터 릴레이 달리기	과학	- 함수의 이해 - 센서에 의한 로봇간의 통신
11차시	벤저민의 옷장 정리하기	수학	- 변수의 이해 - 기억장치의 원리
12차시	로봇 축구대회	사회 과학	- 추상화 하기 - 로봇 동작 모듈 만들기

### 3.2.3. 교수 개발과 적용

수업실행을 위한 교수자료는 교수·학습 과정안과 수업활동지인 학생용 학습 교재로 구분되며 교수·학습 과정안은 초등학교 수업 1차시 분량으로 40분 수업에 적용할 수 있도록 구성하였다. 6주 동안 홀수 차시에는 스크래치만을 이용하고 짝수 차시에는 스크래치와 햄스터 로봇을 결합하여 프로그래밍 수업을 조직하고, 플립드 러닝 방식으로 수업을 진행하였다.

개발된 교육 프로그램은 경기도 소재 A초등학교에서 방학 중 진행되는 방과 후 소프트웨어 교육 과정에 적용하였다. 해당 교육프로그램에 5학년 학생 8명이 참여하였으며, 이들을 대상으로 연구를 수행하였다. 연구참여자의 특성은 <Table 3>과 같다.

<Table 3> Characteristics of participants

참여자	성별	소프트웨어 학습경험 및 특성
A	여	방과후학교의 코딩교육 경험이 있음.
B	여	컴퓨터학원에서 코딩교육 경험 있음
C	남	EBS 코딩교육 프로그램 참여 경험 있음
D	남	코딩교육 경험 없음.
E	여	개인지도로 코딩교육 수강 경험 있음
F	남	코딩교육 경험 없음.
G	여	유튜브 코딩학습 경험 있음
H	남	방과후학교의 코딩교육 경험 있음.

한편, 플립드 러닝 기반의 소프트웨어 교육이 흥미도와 컴퓨팅 사고력 향상에 관한 요인을 추출하기 위해서는 수업 적용을 통해 실제 학생들의 흥미도와 컴퓨팅 사고력이 향상되었을 때 가능하다. 이에 컴퓨팅 사고력의 향상정도를 파악하기 위하여 학생들을 대상으로 컴퓨팅 사고력을 측정하였다. 먼저 컴퓨팅 사고력을 측정하기 위한 측정도구는 비버첼린지[33]에서 제공하는 과제 샘플과 정보올림피아드[34]의 초등부 기출문제(2007년-2016년)에서 초등학생 연령에 해당하는 문제를 추출하여 개발하였다. 검사 도구의 신뢰도를 확보하기 위하여 컴퓨터공학 전문가 2명, 영재교육 전문가 1명에게 검사 도구에 대한 검증을 받았다.

컴퓨팅 사고력 측정은 수업시작 1주일 전과 수업 종료 1주일 후에 지필시험 방식으로 2회 실시하였다. 측정된 자료를 분석한 결과 <Table 4>와 같이 수업에 참여하기 전보다 수업에 참여한 후에 학생들의 컴퓨팅 사고력이 향상되었음을 확인할 수 있었다.

<Table 4> Measurement result of computational thinking

분류 참여자	상		중		하		합계	
	사전	사후	사전	사후	사전	사후	사전	사후
A	0	24	9	18	12	18	21	60
B	24	24	36	18	6	18	66	60
C	24	24	18	27	6	18	48	69
D	0	12	9	9	0	12	9	33

### 3.3. 자료 수집

플립드 러닝 기반의 소프트웨어 교육에서 학생들의 흥미도와 컴퓨팅 사고를 향상시키는 요인이 무엇인지 규명하기 위해서는 수업 과정에서의 학생들의 학습경험을 면밀하게 관찰하고, 겉으로 드러나지 않는 학생들의 생각과 느낌을 표현할 수 있도록 해야 한다. 이에 연구자는 교수자로서 전통적 수업에서와 같이 지식전달자가 아닌 학생의 학습활동을 돕는 조력자 및 촉진자가 되어 학생들의 학습활동에 직간접적으로 관여하고 관찰하는 한편, 학생들의 의견을 청취하기 위하여 지속적인 면담을 진행하였다. 연구 문제에 대한 구체적인 답을 얻기 위하여 수업 중에 주제를 바탕으로 대화할 수 있도록 과정을 구성하였으며, 연구자는 참여자들이 문제에 대하

여 자신의 생각과 느낌을, 그러한 이유에 대하여 자연스럽게 말할 수 있는 분위기를 제공하였다. 관찰 결과들은 수업이 끝난 후 관찰기록으로 상세히 기술하였으며, 학생들과의 대화 결과는 키워드를 바탕으로 전사하였다.

### 3.4. 자료의 분석 및 타당성 검토

본 연구에서는 질적 접근 중 질적 사례연구를 기반으로 연구를 수행하였다. 질적 사례연구는 하나의 사례를 심층적으로 이해하고 검토하고 분석하고 해석함으로써 그 현상에 대한 총체적인 이해가 가능하다. 수집된 자료들은 ‘원자료의 전사’, ‘주제별 약화화’, ‘주제의 발견’이라는 일반적인 질적 연구 분석 단계를 거쳐 정리되었다 [15]. 첫째, 수업과정에서 수집하거나 기록한 자료들에 대한 철저한 전사작업을 실시하였다. 둘째, 심층면담과 관찰 기록 등 자료를 여러 번 정독하면서 전체적으로 파악한 후 자료에서 나타나는 주제를 찾아 기술하였다. 셋째, 가장 빈도가 높게 나타나는 주제들을 학생들의 흥미도와 컴퓨팅 사고력을 향상시키는 요인에 초점을 맞추어 주요

<Table 5> Influential factors on Students' Interest

주제	하위 범주
- 디딤 동영상 시청으로 수업 참여 전에 충분히 동기화 됨.	수업에 대한 기대
- 누적된 학습활동 경험이 새로운 학습 기대를 이끔.	
- 수업을 재미있는 놀이처럼 느낌.	
- 수업활동에 적극적 참여함으로써 다른 부정적 요소는 약화시킴.	
- 자신의 수업참여에 대한 방해를 적극적으로 방어함.	과제에 대한 성취
- 동료학습자의 과제를 공유하여 감상하는 것에 재미를 느낌	
- 자신의 작품을 꼭 완성하려는 의지를 보임	
- 단순한 과제의 완성보다 수준 높은 과제 완성을 위해 노력함.	
- 완성한 자신의 과제를 자랑스러워 함.	동료와 상호 작용
- 작품 감상시간을 좋아하며 자신의 작품을 발표하기 원함	
- 협동학습의 역할에 대해 자신에게 맞는 위치를 찾으려고 함.	
- 과제 해결을 위한 정보 습득에 매우 적극적인.	
- 컴퓨터 문제 해결에 자신감이 생김	과제
- 자신의 작품 발표에서 동료학습들의 호의적 반응에 자신감을 얻음	
- 작품으로 자신의 능력을 나타내 보이고 싶어 함,	
- 동료학습자의 수업활동 과정에 대해 관심을 가짐.	
- 협력을 위한 역할 나눔이 자율적으로 이루어짐.	실제적 과제
- 동료학습자와 함께 하는 작업에 더욱 흥미를 느낌.	
- 동료학습자의 우수한 활동 참관이 학습 활동에 동기부 작용함	
- 자신들이 만든 경기를 즐겁게 함.	
- 새로운 문물에 대해 감동을 받음	실제적 과제
- 미래에 필요한 기술이 무엇인지 인지함	
- 학습의 결과로 나타난 로봇의 실제적 반응 결과에 대해 만족함.	
- 실제 생활에서 사용하는 것과 학습을 연결시킴	
- 로봇 특성에 따라 나타나는 문제점을 찾고 적극적으로 해결함.	
- 로봇을 서로 연결하여 영향을 주고받는 활동에 재미를 느낌	

주제를 추출하였다. 그리고 학습 흥미도와 컴퓨팅 사고력에 영향을 미치는 요인에 맞추어 이를 다시 구체적으로 검토, 분류하여 각각 4개와 3개의 범주로 유목화 하였다.

본 연구의 신뢰도와 타당도를 확보하기 위해서 첫째, 심층면담, 관찰, 내용분석과 같은 자료수집방법의 삼각법(triangulation)이 활용되어 졌으며, 둘째, 연구자의 작업에만 전적으로 의존할 경우 나타날 수 있는 문제들을 보완하기 위하여 동료 연구자들과 자료 분석과정을 공유하고 검토하는 작업(peer review)이 이루어졌다. 셋째, 자료의 수집 방법, 수집 자료에 대한 해석, 자료의 처리 절차에 대한 오류를 검증하기 위하여 교육학 박사 1인에게 전체적인 과정과 분석결과를 제공하고 과정의 정확성과 분석의 타당성 검증을 요청하였다. 검토 결과, 전체적으로 적절하다는 의견이 제시되었다. 마지막으로, 연구과정에 참여한 참여자 중 2명을 따로 선발하여 그들에게 연구자가 도출한 연구결과를 설명해 주고 그들의 경험을 제대로 반영하였는지 점검하는 참여자 점검 과정(member check)을 거쳤다.

## 4. 연구 결과

### 4.1. 흥미도를 높이는 요인

플립드러닝 기반 소프트웨어 교육에 나타난 학습자의 흥미도는 학습효과를 높이는 중요 요인이다. 자료 분석을 통하여 학생들의 흥미도 향상과 관련하여 의미 있는 주제를 도출하였으며, 도출된 주제를 추상화하여 4개의 범주로 유목화 하였다<Table 5>. 분석을 통하여 발견한 흥미 유발 요인은 ‘수업에 대한 기대’, ‘과제에 대한 성취’, ‘동료와의 상호작용’, ‘실제적 과제’였다.

#### 4.1.1. 수업에 대한 기대

학습자들은 인터뷰 과정에서 수업에 참여하는 자신의 기대를 표현했다. 이것은 학습자의 수업에 대한 긍정적 기대가 플립드 러닝 기반 소프트웨어 수업에서 흥미를 높이는 요인이었음을 말하는 것이다. 학습자들은 디딤 동영상을 통해 사전 학습의 내용을 숙지하고, 자신들이 실제 수업에서 하게 될 활동에 대해 기대를 가지고 있

었기 때문에, 과제가 주어지면 바로 활동을 시작하게 된다. 이것은 학습 흥미를 지속적으로 유지할 수 있게 할 뿐만 아니라, 새로운 학습에 긍정적인 영향을 미친다.

“나는 마음이 즐겁고 생각이 신나서 열심히 참여하겠다는 생각을 하고 와요. 미리 동영상 수업으로 간단한 것은 해 보았기 때문에 자신도 있고요.”(교사 심층인터뷰 중, 연구참여자 C)

#### 4.1.2. 과제에 대한 성취

전달식 강의인 소프트웨어 수업에서 학생들은 수업활동 안에서 학습해야 할 내용을 인지하고 활동에 즉각적으로 활용하여야 한다. 빠른 수업활동 전환으로 학생들은 과제에 대한 완성도보다는 과제 완료에 더 집중하게 된다. 그러나 플립드 러닝 기반 수업에서는 활동에 사용할 핵심 내용을 개인의 수준에 맞춰 미리 학습한 상태이므로 수업에서 시간적 여백이 존재하여 학습활동에서 충분히 과제 해결이 가능할 뿐만 아니라 완성도까지 높일 수 있다. 이를 통해 학생들은 과제 성취로 인한 만족감을 드러냈다.

“그동안 다른 수업에서 내가 하려고 한 것을 끝까지 다 못할 때가 있었고요. 또 어떤 때는 겨우 다 만들었는데 수업이 끝나서 가지고 놀지 못하니 까 실망도 되고 그랬어요. 그런데 이번 수업에서는 충분히 여유 시간이 되서 내가 원하는 것을 만들 수 있었어요.”(교사 심층인터뷰 중, 연구참여자 C)

#### 4.1.3. 동료와의 상호작용

플립드 러닝 기반 소프트웨어 교육에서의 수업활동은 주로 협동학습형태를 사용한다. 모둠 형태로 수행할 수 있도록 과제가 주어지기 때문에 이를 해결하기 위해 과제를 분담하여 역할을 정한 후 협력하여야 한다. 이러한 활동에서 문제해결을 위해 서로 방안을 모색하는 과정에서 긍정적 동기를 부여받게 된다. 학생들은 실제 동료와의 상호작용에 대한 감동을 다음과 같이 표현하였다.

“이 수업에서는 친구들이 있어서 더 좋았어요. 음악 연주회 만들 때 모둠에 친구들이 모두 열심히 해 참여했어요. 특히 ○○이가 멜로디 블록 쌓는 것을 혼자서 다해줘서 그게 되게 할 게 많은 건데...”

코딩이 다 완성돼서 실행시켜서 함께 연주하고 감상할 때 정말 즐거웠고, 우리가 하나가 되는 느낌이었어요.”(교사 심층인터뷰 중, 연구참여자 A)

**4.1.4. 실제적 과제**

수업 진도를 나가야 한다는 이유로 학생들에게 자신이 배우고 있는 것이 어떻게 세상과 연계되어 있는지 더 깊은 의미를 알아볼 기회를 제공하지 못하는 것이 전달식 강의 수업의 어려움이었다. 그러나 플립드 러닝에서는 디딤 동영상에 의한 사전학습으로 실제 수업활동 시간이 늘어나는 효과가 있어서 이러한 문제점을 보완하여 수업할 수 있다. 과제로 주어진 내용이 추상적이거나 가상적인 것으로 이루어져 있다면 초등학교 5학년의 학습 경험으로 문제를 분석하여 해결하는 데 어려움이 클 것이다. 그러므로 학령에 따른 발달 수준에 맞추어 과제를 제시하는 것은 무엇보다도 중요하다. 연구에 참여한 학습자는 생활 속에서 일어났던 자신의 경험을 다음과 같이 유쾌하게 말했다.

“제가 어떤 곳을 지날 때면 (소프트웨어 수업이) 나도 모르게 노래가 나오도록 만들었어요. 할머니 집 가는 길에 어떤 곳을 지날 때 마다 꼭 바퀴에서 ‘학교 종이 땡땡땡’ 노래가 나오는 곳이 있거든요.”(교사 심층인터뷰 중, 연구참여자 A).

**4.2. 컴퓨팅 사고력 향상에 영향을 미치는 요인**

플립드 러닝 기반의 소프트웨어 교육프로그램을 모두 종료한 후 컴퓨팅 사고력이 향상되었는지 측정한 결과 선행연구의 결과와 유사하게 앞서 제시한 <Table 4>와 같이 학생들의 컴퓨팅 사고력은 사전 검사에 비해 향상된 것으로 나타났다. 이에 학생들의 컴퓨팅 사고력이 어떤 요인에 따라 향상되었는지를 분석하였다. 분석을 통해 컴퓨팅사고력 향상과 의미 있는 주제를 도출하였으며, 도출된 주제를 추상화하여 3개의 범주로 유목화 하였다<Table 6>. 컴퓨팅 사고력을 향상시키는 요인으로 분석된 것은 ‘과제에 대한 성취’, ‘동료와의 상호작용’, ‘교사의 피드백’으로 나타났다.

<Table 6> Influential factors on computational thinking

주제	하위 범주
- 협동학습에서 자신이 맡은 역할에 대해 책임감이 높아짐.	
- 미해결 과제에 대한 해결 욕구가 높아짐.	과제에 대한 성취
- 기계 종속적인 과제에 대한 이해도가 높아짐.	
- 언플러그드 학습에서의 컴퓨팅 사고력 적용이 나타남.	
- 문제풀이에서 집중력을 보임.	
- 어려운 과제에서도 쉽게 포기하지 않고 끝까지 시도하여 완수함.	
- 동료학습자와의 심적 경계가 무너지고 과제수행동안 적극적으로 협력함.	
- 값의 변화에 따라 로봇의 행동이 바뀌는 것을 동료학습자와의 의견교환을 통해 점검함.	동료와의
- 자신의 의견을 동료학습자와 나누고 싶어함.	의
- 문제 상황에서의 동료와의 의견교환으로 문제점을 상호	상호 작용
추론함.	
- 협력적으로 경기를 함.	
- 동료학습자와의 토의로 문제점 해결	
- 자율적 토의와 토론으로 경기규칙을 정함.	
- 학습능력의 차이로 인한 동료학습자와의 의사소통이 원활하지 못함.	
- 의문에 대한 질문내용에 적극적으로 자신의 생각을 반영.	
- 과제에 대한 이해부족으로 긴장함.	
- 문제에 대한 부담을 줄여주는 교사의 개입에 안도하여 작업을 시작함.	교사의
- 자신이 알게 된 것에 대해 말하며 성취한 학습의 피드백	결과를
표출하고 싶어 함.	
- 학습활동 중에 쌓은 신뢰가 다른 분야에서도 그대로 적용됨.	
- 문제에 논리적으로 반응함.	
- 문제해결의 알고리즘을 찾는 속도가 빨라짐.	
- 기계 종속적인 문제해결에 어려움을 보임	

**4.2.1. 과제에 대한 성취**

플립드 러닝 기반 소프트웨어 교육에서는 수업 활동에 사용할 기본적인 지식을 동영상 사전학습을 통해 학생들은 이미 숙지한 상태이기 때문에 과제를 완성하는 속도가 빠르다. 따라서 수업시간의 10~20%를 학생들이 자신의 흥미를 바탕으로 프로젝트와 연구를 할 수 있도록 활용하여 스스로 배운 것들을 다양한 방식으로 실험해 볼 수 있다. 이런 심화 학습으로 학생들은 과제 수행 활동에 더욱 자신감을 가질 뿐만 아니라, 여러 측면에서 과제를 세밀히 분석할 시간을 가질 수 있다. 따라서 미해결 과제를 남기지 않고 해결함으로써 과제 속에 포함



된 원리인 컴퓨팅 사고력 습득할 수 있다.

“아이들끼리 서로 경쟁하듯이 좀 더 다양하게 신호 체계를 만들어 보려고 시도한다. 또 서로 친구가 하는 것에도 관심을 보이면서 다른 친구의 아이디어도 자연스럽게 차용하고 자신의 생각을 보태어 발전시켰다.” {연구자 관찰일지 중}

**4.2.2. 동료와의 상호작용**

플립드 러닝 기반의 소프트웨어 수업 활동은 주로 협동학습의 형태로 제시된다. 과제 수행에 대한 각자의 역할을 나누어 수행하면서 자신의 역할과 함께 동료의 역할도 참고해서 연결해야 할 때가 많다. 이 때 학생들은 동료의 역할 과제 수행에 대해 질문하고 대답하며 상호작용을 함으로써 능력이 낮은 학생도 동료 학습자의 우수한 컴퓨팅 사고력 성취에 영향 받을 수 있다.

“중복 숫자 찾기 놀이에서 ○○이는 두 묶음의 번호표를 받아두고 잠시 그냥 있었다. 그러다가 △△이가 번호표를 정리하는 모습을 보더니 △△보다 빨리 중복되지 않은 번호표 묶음을 으뜸차순으로 정렬하였다. 그리고 중복된 번호표 묶음에서 번호표 하나씩을 빼서 이분법을 이용하여 중복 번호를 찾았다.” {연구자 관찰일지 중}

**4.2.3. 교사의 피드백**

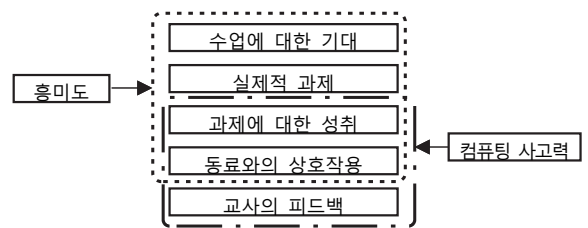
교사와 학생의 관계를 최고 수준으로 만드는 것은 서로 간의 활발한 피드백이다. 플립드 러닝 기반 소프트웨어 교육에서는 교사가 1차적으로 동영상 시청 후 퀴즈를 통해 학생들의 이해정도를 파악했기 때문에 구체적인 피드백을 할 수 있다. 또, 수업활동 시간에도 지식 전달과 같은 강의식 수업을 하지 않기 때문에 학생들을 심도 있게 관찰하는 시간이 확보된다. 따라서 각 학생 개인의 특성과 수준에 맞는 질문과 설명을 하여 학습을 돕거나 촉진하는 개별화 된 맞춤형 교육이 가능해진다.

“선생님이 항상 제 이야기를 들어주시고, 또 내가 모르는 것을 이미 알고서 알려주려 하시고, 제가 질문한 것에 대해서 귀찮아하지 않고 잘 말씀해 주시고 하니까 모르던 것이 잘 이해돼서 코

딩하면서 별로 어렵다고 안 느꼈어요.”(교사 심층 인터뷰 중, 연구참여자 B)

**5. 결론 및 제언**

본 연구는 초등학교에서 플립드 러닝 기반 소프트웨어 교육 프로그램을 개발하고 적용하여 소프트웨어에 대한 흥미도와 컴퓨팅 사고력을 향상시키는 요인이 무엇인지 분석하고자 하였다.



[Figure 1] Influential Factors on Elementary Students' Interest and computational thinking

연구결과 [Figure 1]과 같이 플립드 러닝 기반 초등 소프트웨어교육에서 학습 흥미와 만족도를 높이는 요인은 ‘수업에 대한 기대’, ‘과제에 대한 성취’, ‘동료와의 상호작용’, ‘실제적 과제’로 확인되었다. 또, 컴퓨팅 사고력을 증진시키는 요인은 ‘과제에 대한 성취’, ‘동료와의 상호작용’, ‘교사의 피드백’으로 나타났다 또 이러한 요인들 중 “과제에 대한 성취”와 “동료와의 상호작용”은 흥미도와 컴퓨팅사고력을 높이는 공통적 요인으로 작용하는 상호의존적 관계를 갖는다고 할 수 있다.

이러한 연구결과를 바탕으로 플립드 러닝 기반 초등학교 소프트웨어 교육에서 흥미를 높이고 컴퓨팅 사고력을 증진시키는 요인을 바탕으로 수업단계에 따른 교수학습 전략을 <Table 7>과 같이 제시할 수 있다.

첫째, 사전 학습 단계에서는 디딤 동영상 시청, 소통 과정인 댓글 달기, 학습 확인 과정인 퀴즈 풀기 항목으로 구성한다. 디딤 동영상은 학생들의 관심, 경험, 인지수준을 파악하여 수업 담당교사가 학생들이 긍정적 기대를 가지도록 동기화하는 요소를 동영상에 포함하여 학생들이 직접 제작해 볼 수 있는 기회를 제공하는 것을 원칙으로 하였다. 이를 통해 학생들과 교사와의 친밀

<Table 7> Teaching and learning strategies for software education based on flipped learning

단계	활동	교수학습 전략	흥미도 및 컴퓨팅 사고력 요인
사전 학습	디딤 동영상 시청	- 학생들의 관심, 경험, 인지수준을 파악하여 수업 담당교사가 직접 디딤 영상을 제작한다. - 디딤 동영상과 연결되어 참여하는 수업활동에 대해 학생들이 긍정적 기대를 가지도록 동기화하는 요소를 동영상에 포함한다.	수업에 대한 기대(흥미도)
	넷글 달기와 퀴즈 풀기	- 교사는 디딤 영상 시청 후 퀴즈를 통해 학생들의 사전학습 정도를 파악하여 본 수업 활동에서 피드백에 반영한다. - 교사는 넷글달기를 통해 학생들이 가진 수업 활동에 대한 의견을 수렴하여 분석한 후 다시 질문하기, 수업 활동의 간단한 안내 등을 함으로써 학생들이 수업참여에 충분히 동기화 되도록 자극한다.	
준비도 확인 및 학습 과제 안내	수업 준비도 확인	- 퀴즈에 대한 정답 확인 및 학생들의 이해 부족 부분에 대해 해당 학생에게 구체적으로 개별 질문을 하고 생각을 말하도록 한다.	교사의 피드백(컴퓨팅 사고력)
	학습 과제 안내	- 학습과제를 안내할 때는 이론적이거나 추상적인 사실들을 배제하고, 실생활에서 학생들의 관심이 높거나, 밀접하게 관련된 상황 또는 사건을 찾아 학생들에게 소개하고, 그와 관련하여 과제를 제시한다.	수업에 대한 기대(흥미도) 실제적 과제(흥미도)
과제 수행 및 해결	과제 분석	- 과제는 모듈 단위의 협동학습 형태로 제시하여 학생들이 함께 문제를 분석하고 토의, 토론하도록 한다.	실제적 과제(흥미도) 동료와의 상호작용(흥미도, 컴퓨팅 사고력)
	과제 수행	- 과제 수행 활동에서는 개별학습을 충분히 할 수 있도록 수업의 여백을 둔다.	교사의 피드백(컴퓨팅 사고력)
	상호 검토	- 개별 활동이 끝난 후에 모듈끼리 협의하여 과제에 대한 완성도를 높이도록 한다.	
정리 및 평가	활동 평가 및 공유	- 완성과제를 전체 학생 앞에 실연한다. 자신들의 작품에 대해 설명하고, 다른 모듈으로부터 좋은 점과 아쉬운 점에 대한 의견을 수렴하는 시간을 충분히 가진다. - 반론의 시간을 주어 토의할 수 있도록 한다.	과제에 대한 성취(흥미도, 컴퓨팅 사고력) 교사의 피드백(컴퓨팅 사고력)
	정리	- 교사가 전체 수업 활동을 통해 꼭 알아야 할 개념과 원리를 다시 한번 학생들에게 질문하여 강조하고 지식이 구조화 되도록 돕는다.	

도를 높이고, 수업에 대한 기대를 향상시킴으로써 학생들의 흥미를 높일 수 있다.

둘째, 준비도 확인 및 학습 과제 안내 단계에서는 수업준비도 확인, 학습 과제 안내 항목으로 구성한다. 학생들의 수업 준비 정도를 확인하는 과정으로 퀴즈에 대한 정답 및 학생들의 이해 부족 부분에 대해 해당 학생에게 구체적으로 개별 질문을 하고 생각을 말하도록 하여 지도한다. 학생들의 수업 준비가 확인되면 교사는 학습과제를 안내한다. 이 때 학습과제는 이론적이거나 추상적인 것들이 아닌, 실생활에서 학생들의 관심이 높고 밀접하게 관련된 상황 또는 사건을 찾아 학생들에게 소개하고, 그와 관련된 과제를 제시하도록 한다.

셋째, 과제 수행 및 해결 단계에서는 과제 분석, 과제 수행, 상호 검토 과정을 거친다. 과제는 모듈 단위의 협동학습 형태로 제시하여 학생들이 함께 문제를 분석하고 토의·토론할 수 있도록 한다. 이 후에 과제 수행 활

동에서는 개별학습을 충분히 할 수 있도록 수업의 여백을 둔다. 수업의 여백은 학생들의 고차원적인 사고를 촉진할 수 있는 시간이 될 것이므로 플립드 러닝 수업 활동에 있어 매우 긍정적 요소로 작용하는 부분이다. 이러한 개별 활동이 끝나면 모듈끼리 협의하여 과제에 대한 완성도를 더 높이게 된다.

넷째, 정리 및 평가 단계에서는 학습 활동을 평가하고 공유하며 정리한다. 완성과제를 전체 학생 앞에 실연함으로써, 학생들은 자신들의 작품에 대해 설명하고, 다른 모듈으로부터 좋은 점과 아쉬운 점에 대한 의견을 수렴하는 시간을 가진다. 또 서로의 작품에 대하여 토의·토론을 함으로써 안목을 넓힐 수 있다. 이러한 과정이 모두 끝났을 때 교사가 전체 수업 활동을 통해 꼭 알아야 할 개념과 원리를 다시 한번 학생들에게 질문하여 강조하고 지식이 구조화 되도록 도울 수 있다.

본 연구는 플립드 러닝 기반의 소프트웨어 교육에서

의 학습과정을 질적으로 분석하여 소프트웨어 교육에 대한 흥미와 컴퓨팅 사고에 영향을 미치는 요인들을 도출했다는 점에서 의의가 있다. 본 연구의 한계점을 바탕으로 후속연구를 제안하면 다음과 같다. 첫째, 본 연구는 소수의 초등학교 학생을 대상으로 수행된 연구이므로 중·고등학생부터 대학생까지 연구대상을 확대할 필요가 있다. 둘째, 본 연구에서는 컴퓨팅 사고력과 정보문화소양능력 요소 중에 몇 가지에 한정하여 프로그램을 구성한 것이기 때문에 향후 연구에서는 소프트웨어 교육과 관련된 더 많은 요소가 포함된 프로그램을 개발하고 그에 대한 효과성 검증을 기대한다.

### 참고문헌

- [1] Bergmann, J., & Sams, A. (2014). *Flipped learning: Gateway to student engagement*. International Society for Technology in Education.
- [2] Choi, H. S. (2018). Domestic Literature Review on Computational Thinking Development through Software Programming Education. *Journal of Educational Technology, 34*(3), 743-774.
- [3] Choi, S. Y. (2015). A Study of Problems and their Solving Strategies Consequent upon Software Education Reinforcement in Primary and Secondary Schools. *The Journal of Korean Association of Computer Education, 18*(3), 93-104.
- [4] Davenport, C. E. (2018). Evolution in Student Perceptions of a Flipped Classroom in a Computer Programming Course. *Journal of College Science Teaching, 47*(4), 30-35.
- [5] Hamdan, N., McKnight, P., McKnight, K., & Arfstrom, K. M. (2013). *The flipped learning model: A white paper based on the literature review titled a review of flipped learning*. Flipped Learning Network/ Pearson/George Mason University.
- [6] Han, T. I. (2017). The Effects of Flipped-Learning on Learning Motivation and Class Satisfaction in Software Education. *Journal of The Korean Association of Information Education, 21*(6), 665-673.
- [7] Jeon, S. J. (2016). The Effect of Flipped Learning on Learning Motivation in Software Education. *Journal of The Korean Association of Information Education, 20*(5), 433-442.
- [8] Jung, Y. S. (2017). *Software education of 4th industry revolution age*. National Information Society Agency.
- [9] Jeong, Y. S. (2018). The Problems and Improvement of the SW Education Policy in Elementary School. *The Korean Association of Information Education Research Journal, 9*(1), 91-97.
- [10] Kim, K. M., & Kim, H. S. (2017). A Study on Customized Software Education method using Flipped Learning in the Digital Age. *Journal of Digital Convergence, 15*(7), 55-64.
- [11] Kim, K. Y., & Yoo, I. H. (2017). Effects of SW Education using app inventor on computational thinking and attitude towards computer of elementary school students. *Journal of The Korean Association of Information Education, 21*(4), 371-380.
- [12] Kim, J. S., Han, S. K., Kim, S. W., Jung, S. W., Yang, J. M., Jang, E. D., & Kim, J. N. (2015). *Development of teaching and learning model for SW education*. CR 2015-35. Korea Educational Development Institute · Korea Education and Research Information Service.
- [13] Kim, T. Y., & Kim, H. J. (2019). Effects of education program using ICT based SW education robot system on academic interest. *Asia-pacific Journal of Multimedia Services Convergent with Art, Humanities, and Sociology, 9*(5), 833-844.
- [14] Kim, Y. B. (2015). *Development of the design principles on flipped learning support system*. Unpublished Doctoral Dissertation, Pusan University.
- [15] Kim, Y. C. (2006). *Qualitative research methodology 1*. Seoul: Moonumsa.
- [16] Kim, Y. J. (2016). *Development and Application of Elementary School Software Education Programs*. Master's thesis, Seoul National University of Education.
- [17] Lee, A. H. (2018). Domestic Research Trends Analysis of Software Education. *The Journal of*

- Educational Information and Media*, 24(2), 277-301.
- [18] Lee, J. M., Jung, Y. J., & Park, H. K. (2017). Gender Differences in Computational Thinking, Creativity, and Academic Interest on Elementary SW Education. *Journal of The Korean Association of Information Education*, 21(4), 381-380.
- [19] Lee, J. M., Park, H. K., & Choi, H. S. (2018). Effects of SW Education Using Robots on Computational Thinking, Creativity, Academic Interest and Collaborative Skill. *Journal of The Korean Association of Information Education*, 22(1), 9-21.
- [20] Lee, M. H., & Ham, S. B. (2018). The Development and Application of a Teaching and Learning Model Based on Flipped Learning for Convergence Software Education in Elementary Schools. *Journal of The Korean Association of Information Education*, 22(2), 213-222.
- [21] Lee, M. K. (2016). *Understanding and Practice of Flipped Learning*. Paju:Kyoyookbook
- [22] Leem, J. H. (2018). Main Issues of Software Education and Tasks of Educational Technology for improving Software Education. *Journal of Educational Technology*, 34(3), 679-709.
- [23] Na, C. S., Joo, H., Lee, J. J., & Kim, D. S. (2018). Inducing Computational Thinking in Korean SW Education: Synthesizing Standardized Mean Changes through Meta-analysis. *Journal of Educational Technology*, 34(3), 775-815.
- [24] Oh, I. S. (2018). *A Study on Enhancing Computational Thinking through Flipped Learning-based Software Education*. Master's thesis, Konkuk University.
- [25] Senske, N. (2017, March). Evaluation and Impact of a Required Computational Thinking Course for Architecture Students. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education* (pp. 525-530). ACM.
- [26] Seo, E. K. (2017). Development of Creative Thinking and Coding Course method on Design Thinking using Flipped Learning. *Journal of Learner-Centered Curriculum and Instruction*, 17(16), 173-199.
- [27] Shin, J. H., & Park, S. Y. (2017). A study on faculty perception of flipped learning implementation in higher education : The case of A university. *Journal of Learner-Centered Curriculum and Instruction*, 17(15), 347-371.
- [28] Shute, V. J., & Ventura, M. (2013). *Stealth assessment: Measuring and supporting learning in video games*. Cambridge: MIT Press.
- [29] Son, K. J., & Han, J. H. (2018). Comparison on effectiveness of SW education using robots based on narrative-paper art activities. *Journal of The Korean Association of Information Education*, 22(4), 419-425.
- [30] Song, H. N., Ryu, M. Y., & Han, S. K. (2018). The flipped learning-based SW-STEAM education program for learning motivation. *Journal of The Korean Association of Information Education*, 22(3), 325-333.
- [31] Tugun, V., Uzunboylu, H., & Ozdamli, F. (2017). Coding education in a flipped classroom. *TEM Journal*, 6(3), 599-606.
- [32] Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.
- [33] Bebras Challenge 2015, 2016. [www.bebas.uk](http://www.bebas.uk)
- [34] Korea Olympiad in Informatics. 2016. <https://www.digitalculture.or.kr/koi>

### 저자소개



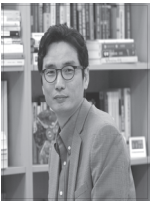
#### 임 경 희

2019 아주대학교 소프트웨어교육학(석사)

2019~ 현재 성균관대학교 컴퓨터교육학과 박사과정

관심분야 : 소프트웨어 교육, 교수 학습방법, 소프트웨어교육 평가방법

E-mail : khlim626@g.skku.edu



#### 신 종 호

2011 한양대학교 교육공학과(박사)

2015~ 현재 아주대학교 다산학부 대학 조교수

관심분야 : 혁신 교수방법, 학습분석, 교수개발

E-mail : jayshin@ajou.ac.kr