

문제 중심 학습을 적용한 스크래치 프로그래밍 교수 학습 모형

배학진[†] · 이은경^{††} · 이영준^{†††}

요 약

스크래치는 교육용 프로그래밍 언어 중 하나로 풍부한 멀티미디어 프로그래밍 환경과 쉽고 흥미로운 사용자 인터페이스를 제공한다. 또한 한국어를 지원하고 있어 초·중등학교 학습자들의 프로그래밍 교육에 유용하게 활용되고 있다. 그러나 프로그래밍 과정은 복잡한 문제해결과정으로 논리적 사고 및 추상적 사고와 같은 고차원적인 인지 능력을 요구하기 때문에 여전히 어린 학습자들에게 인지적 부담을 유발할 수 있다. 따라서 본 연구에서는 프로그래밍 학습에 관한 학습자의 흥미와 내적 동기를 유발하고 교육용 프로그래밍 언어인 스크래치 활용의 효과를 최대화하기 위한 교수 학습 방법으로 문제 중심 스크래치 프로그래밍 교수 학습 모형을 개발하였다. 해당 모형은 기존 문제 중심 학습을 기반으로 프로그래밍 활동에서 다루어지는 문제와 문제해결의 특성을 반영하고, 초등학교 학습자라는 학습자 특성을 반영하였다. 개발한 프로그램을 실제 초등학교 5학년 학습자들에게 적용한 결과, 해당 모형을 적용한 집단이 일반적인 스크래치 프로그래밍 교수 학습을 진행한 집단에 비해 논리적 사고력 및 문제해결력이 유의하게 높게 나타났다.

주제어 : 교육용 프로그래밍 언어, 스크래치, 문제 중심 학습, 프로그래밍 교육

A Problem Based Teaching and Learning Model for Scratch Programming Education

HakJin Bae[†] · EunKyoung Lee^{††} · YoungJun Lee^{†††}

ABSTRACT

Scratch, one of the educational programming languages, provides a media-rich programming environment and easy interface to users. It supports Korean language and is utilized usefully in programming classes in elementary and middle schools. However, programming causes cognitive loads to young students. Because the programming process is a complex problem solving procedure that requires logical and abstract thinking abilities. Therefore, we developed a problem based scratch programming teaching and learning model to enhance intrinsic motivation of learners and to maximize the effects of using the scratch, educational programming language. The developed problem based teaching and learning model considered elementary students' characteristics. It was implemented in fifth grade elementary school classes and the educational effects of the model was analysed. The developed model was helpful in enhancing students' problem solving potential and logical thinking abilities.

Keywords : Educational Programming Language, Scratch, Problem Based Learning, Programming Education

†정 회원: 서울무현초등학교 교사
 ††중신회원: 청운중학교 교사
 †††중신회원: 한국교원대학교 컴퓨터교육과 교수(교신전자)
 논문접수: 2009년 04월 02일, 심사완료: 2009년 05월 15일

1. 서 론

프로그래밍 교육은 학습자의 논리적 사고력, 문제해결력과 같은 고등 사고 능력 향상을 위해 꼭 필요하지만, 국내의 경우 전문계 고등학교 또는 대학교육에서 제한적으로 이루어지고 있다. 그러나 최근 초등학교 단계에서부터 체계적인 프로그래밍 교육의 필요성이 대두됨에 따라 어린 학습자를 대상으로 한 프로그래밍 교육 방법에 대한 연구와 논의가 활발히 이루어지고 있으며, 이에 대한 대안으로 교육용 프로그래밍 언어의 활용을 적극 권장하고 있다.

국내의 경우, 2005년 개정된 ‘초·중등학교 정보통신기술교육 운영지침’에서 ‘정보 처리의 이해’ 영역을 통해 초등학교 1학년부터 알고리즘과 프로그래밍 학습에 관한 내용을 체계적으로 제시하고 있으며, 다양한 교육용 프로그래밍 언어의 활용을 권장하고 있다[1]. 2007년 개정된 중등학교 ‘정보(Informatics)’ 교육과정 또한 프로그래밍 교육을 위한 도구로 교육용 프로그래밍 언어를 사용할 것을 권장하고 있다[2][3].

미국의 경우 ACM(Association for Computing Machinery)에서 제시한 ‘K-12 컴퓨터 과학 모델 교육과정’에서 “알고리즘적 사고를 위해서는 로고와 같은 교육용 프로그래밍 언어를 통한 프로그래밍 교육이 필요하다.”고 제시하고 있으며, 인도의 경우 정규 교육과정에서 초·중학교의 프로그래밍 교육을 위해 로고를 사용할 것을 명시하고 있다[4][5].

교육용 프로그래밍 언어는 기존의 텍스트 형식에서 벗어나 시각적인 화면과 쉬운 인터페이스를 제공해 준다. 특히 명령어가 시각적인 블록 형태로 이루어져 있어 순서에 따라 해당 블록을 끌어다 놓기 활동을 통해 조합함으로써 프로그램을 쉽게 완성할 수 있다. 이와 같은 특성을 지닌 교육용 프로그래밍 언어를 활용함으로써 언어 자체에 대한 학습이나 문법적 오류에 대한 학습 부담을 없애고 논리적으로 자신의 생각을 어떻게 표현할 것인가에만 집중하게 할 수 있다.

따라서 이러한 프로그래밍 학습을 위한 도구의 전환은 기존 프로그래밍 교육의 한계를 극복하고

초등학교 단계에서부터의 프로그래밍 교육의 활성화를 가져오기 위해 의미 있는 선택이라고 할 수 있다.

그러나 프로그래밍의 과정은 학습자들에게 논리적인 사고와 추상적인 추론 과정 등의 복잡한 인지 능력을 요구하기 때문에 어린 학습자들에게 여전히 과도한 인지적 부담을 유발할 수 있다[6]. 따라서 프로그래밍 학습에 대한 어린 학습자들의 인지적 부담을 감소시키고 학습 자체에 대한 내적 동기와 흥미를 유발하기 위한 교수 학습 방법에 대한 연구가 필요하다.

본 연구에서는 문제 중심 학습에 기반한 프로그래밍 교수 학습 모형을 개발하였다. 실생활과 관련된 실제적 과제의 제시를 학습자가 학습 활동에 몰입하게 하고 흥미를 유발할 수 있다. 또한 학습자들은 단편적이고 수동적인 지식을 습득하는 것이 아니라 교수 학습 상황에서 주도적으로 문제를 분석하고 해결하는 경험을 통해 자기 주도적 학습 능력과 문제해결능력을 기를 수 있을 뿐만 아니라, 소그룹 활동을 통한 문제 해결 과정은 학습자들의 인지적 부하를 감소시켜 주고 협력적 사고를 촉진시킬 수 있기 때문이다[7][8].

2. 관련 연구

2.1 스크래치 활용 프로그래밍 교육의 가치와 방향

교육용 프로그래밍 언어는 컴퓨터 과학뿐만 아니라 수학이나 과학과 같은 다양한 교과에서 활용되고 있는 교수 학습 도구이다. 대표적인 교육용 프로그래밍 언어는 로고(Logo), 스퀸 이토이(Squeak Etoys), 앨리스(Alice), 스크래치(Scratch) 등이 있으며, 이러한 언어들을 활용한 연구 결과들은 학습자들의 동기 유발 및 문제해결력 향상에 긍정적인 영향을 주었음을 보여주고 있다[9][10][11].

스크래치는 스퀸 이토이의 영향을 받아 개발된 언어로 풍부한 멀티미디어 기반 프로그래밍 환경을 제공하며, 전 세계 다양한 국가의 언어를 지원한다. 이러한 특성은 프로그래밍 과정에서 모국어

가 아닌 영어 사용으로 인해 부가될 수 있는 인지적 부담을 감소시킴으로서 학습 외적인 내용이 아닌 근본적인 학습 활동에 집중하게 할 수 있다.

스크래치 활용 교육과 관련된 선행 연구 결과들을 종합하여 프로그래밍 교육을 위한 스크래치 활용의 가치를 제시하면 다음과 같다.

첫째, 프로그래밍 교육에서 부가될 수 있는 학습 외적 인지 부하를 감소시킬 수 있다. 기존의 프로그래밍 교육의 가장 큰 문제점은 도구 사용법 자체의 어려움으로 인해 프로그래밍을 통한 추상적 개념의 습득 및 논리적 사고력의 향상보다 도구 사용법을 학습하는데 많은 시간과 노력을 투자함으로써 근본적인 학습 목표에 도달하지 못하고 있다는 점이다[9]. 스크래치의 경우 쉬운 사용자 인터페이스와 모국어 지원 등의 특성으로 인해 도구 자체의 어려움으로 인한 외적 인지부하를 감소시킬 수 있다. 특히 블록들의 조합을 통한 프로그래밍 방식은 구문 오류를 사전에 예방함으로써 문법적 오류 디버깅에 대한 노력을 최소화할 수 있다는 점에서 다른 교육용 프로그래밍 언어와 차이를 지닌다[9]. 실제로 Malan 과 Leitner(2007)의 연구에서는 대학의 프로그래밍 코스에서 프로그래밍과 관련된 사전 지식이 전혀 없는 초보 학습자들이 스크래치를 통해 미리 프로그래밍과 관련된 기본 개념을 익힘으로써, 이후 진행된 자바 프로그래밍 학습에 긍정적인 영향을 끼쳤음을 보여주었다[12].

둘째, 즉각적이고 구체적인 피드백을 제공함으로써 학습자의 흥미와 내적 동기를 유발할 수 있다. 이은경과 이영준(2008)의 연구에서는 중학생을 대상으로 한 프로그래밍 교육에 스크래치를 활용함으로써 학습에 대한 내적 동기를 설명하는 몰입수준과 프로그래밍 수행 능력 측면에서 유의한 영향을 주었음을 확인하였다[9].

셋째, 스크래치 활용 프로그래밍 교육은 협력적 문제 해결의 기회를 제공하고 상호작용을 촉진할 수 있다. Papert의 구성주의 철학에 의하면 학습자들은 구체물이나 공유 가능한 생산품을 협력적으로 구성해나가는 과정을 통해 이해를 증진시킨다. 스크래치의 경우 학습자가 스스로 구성한 작품을 웹상에 쉽게 게시하고 공유할 수 있는 기능을 제공함으로써 협력적 문제 해결의 기회를 제

공하기 위한 학습 환경을 제공한다. 그러나 단지 새로운 프로그래밍 도구의 투입이 완전한 학습효과를 보장할 수 없다. 따라서 새로운 도구 사용을 통한 신기효과를 넘어 프로그래밍 학습에서의 스크래치 활용의 효과를 최대화하고 도구가 아닌 프로그래밍 학습에 관한 내적 동기를 유발하기 위한 교수 학습 설계에 관한 노력이 요구된다.

2.2 문제 중심 학습

문제 중심 학습은 구성주의에 기반한 교수 학습 모형으로 학습 전 과정에서 학습자들의 주도적 역할을 강조하며 학습을 위해 주어지는 구체적 상황에 기반한 ‘문제’ 혹은 ‘과제’를 해결하는 과정이다[7]. 문제 중심 학습에서 학습자들은 정보를 수집하고, 가설을 설정하고, 자료를 분석하고, 가설을 검증하고 해결책을 찾기 위해 과학적인 방법을 사용한다. 여기서의 ‘문제’는 학습자가 자신의 지식영역에서 지식과 기능을 획득하고 촉진할 수 있도록 설계되어진다. 또한 학습자들은 문제를 해결하기 위해 이용될 수 있는 새로운 정보를 습득하는 과정에서 학습동기가 유발되며 이러한 접근방법을 통해 내적 보상에 기초한 자기 주도적 학습 능력을 기를 수 있다[13]. 즉, 문제 중심 학습은 비구조적이고 현실세계가 반영된 문제 상황을 중심으로 교육과정과 수업을 구조화한 교육적 접근으로, 학습자가 스스로 또는 협력적으로 문제를 해결하는 과정에서 고등 사고 능력과 협력적 사고 능력을 신장시킬 수 있도록 하는 교수 학습 모형이라고 할 수 있다.

<표 1>은 기존 연구에서 제시된 다양한 문제 중심 학습 모형을 비교 분석한 것으로 다음과 같은 공통 요소를 지닌다.

첫째, 다양한 방식으로 문제를 제시하고 내면화하기 위한 전략을 사용한다.

둘째, 결과를 공유하기 위한 기회를 제공한다.

셋째, 해결책을 탐색하고 숙고하는 과정을 중요시한다.

넷째, 자유롭고 활발한 의사소통을 유도한다.

이러한 문제 중심 학습 모형이 프로그래밍 교육에 적용된다면 프로그래밍 교육의 궁극적인 목표인 논리적 사고력, 문제해결력과 같은 고등 사

고 능력을 향상시키는 데 도움이 될 수 있지만, 프로그래밍 학습 내용 및 학습 환경의 특성에 맞게 수정될 필요가 있다.

즉, 일반적인 문제 중심 학습에서 다루는 문제는 단지 실생활 속의 비구조화된 문제들이지만 프로그래밍 학습에서의 문제는 컴퓨터를 통해 이러한 문제를 해결하기 위한 구체적인 산물인 프로그램의 구현을 요구한다는 점에서 차이가 있다. 또한 이러한 문제를 해결하기 위한 알고리즘을 구체적으로 개발하고 이를 적용한 프로그램을 구현하는 과정이 문제해결의 과정이 된다.

<표 1> 문제 중심 학습 전개과정 비교

모형	전개 과정
Barrows 와 Myers	수업소개 ▶ 분위기조성 ▶ 문제제시 및 내면화 ▶ 역할 분담 ▶ 해결안 숙고(생각, 사실, 학습과제, 실행 계획) ▶ 해결안 정비 ▶ 학습과제 규명 및 분담 ▶ 학습자료 선정, 선택 ▶ 의견교환 ▶ 새로운 접근 시도(생각, 사실, 학습과제, 실행계획)[14]
IMSA	수업준비 ▶ 문제제시 ▶ 알고 있는 것, 알아야 할 내용 기록 ▶ 문제 정의 ▶ 정보의 수집과 공유 ▶ 가능한 해결안 고안 ▶ 적절한 해결안 확정 ▶ 평가 ▶ 결과물 발표[15]
Delisle	문제제시 ▶ 전체계획(아이디어, 사실, 학습과제, 활동계획) ▶ 문제해결 찾아가기 ▶ 문제해결 다시 찾아가기 ▶ 평가하기[16]
Wood	용어 정의 ▶ 문제정의/토론문제탐색 ▶ 브레인스토밍 ▶ 잠정적 해결안 ▶ 학습목표설정 ▶ 개별학습 ▶ 결과 공유[17]
조연순과 우재경	문제발견과 정의 ▶ 탐구와 정보탐색의 과정 ▶ 해결책의 산출과 평가[18]

3. 문제 중심 스크래치 프로그래밍 수업 모형 개발

3.1 모형 설계 원리

프로그래밍 수업을 통한 학습자의 논리적 사고력 및 문제해결력 향상을 위한 수업 모형은 다음과 같은 설계 원리에 따라 개발하였다.

첫째, 초등학생이라는 학습자의 발달 특성을 고려한다. 프로그래밍의 과정 및 문제 중심 학습의 참여는 어린 학습자들에게 과도한 인지적 부담을 일으킬 수 있다. 따라서 학습자들이 본격적인 문

제 중심 학습을 시작하기 이전에 기본적인 도구 사용법 및 프로그래밍 관련 개념 습득을 위한 수업이 선행될 수 있도록 교육 내용을 설계하였다.

둘째, 실생활과 관련된 문제 상황 및 문제를 설계하여 프로그래밍을 통한 문제 해결 활동이 학습자들에게 유의미한 경험이 될 수 있도록 하였다. 학습자들에게 실생활의 문제 해결과 가장 유사한 경험을 제공함으로써 학습자들의 문제해결력을 향상시킬 수 있기 때문이다.

셋째, 초등학교 정보통신기술교육의 실질적인 운영 상황을 고려하여 재량활동 시간에 교수 학습 진행이 가능하도록 교육 내용을 구성하였다. ‘초·중등학교 정보통신기술교육 운영지침’에 따르면 초등학교 5~6학년의 경우 재량활동이나 특별활동, 실과 교과와 컴퓨터 관련 영역을 활용하여 연간 34시간 이상의 시간을 확보하여 정보통신기술 관련 교육을 진행하도록 하고 있다[1]. 따라서 주당 1차시에 해당하는 수업 시수를 고려하여 교육 내용을 구성하였다.

넷째, 교수 학습의 형태는 수업 단계별 특성에 따라 적합한 형태로 설계하였다. 즉, 기본적인 도구 사용법 및 개념 이해를 위한 학습 단계에서는 개별학습을, 프로그래밍을 통한 실제적인 문제 해결 학습 단계에서는 협력 및 경쟁학습 형태로 구성하였다.

다섯째, 프로그래밍을 통한 문제 해결에 집중할 수 있도록 문제를 설계한다. 기존의 문제 중심 학습에서 다루는 문제들이 범교과적이고 실제적인 문제인데 반해 본 연구에서 설계한 문제들은 프로그래밍 과정을 통해 해결할 수 있는 실제적 문제들로 제한하여 설계하였다.

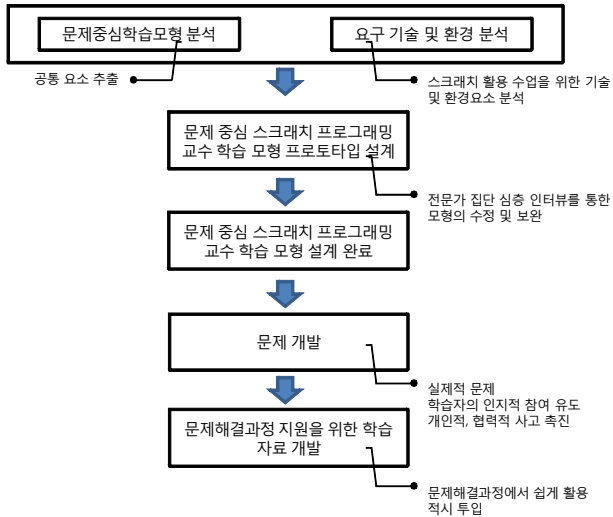
여섯째, 문제해결을 위한 프로그래밍 도구는 풍부한 멀티미디어 환경을 제공하는 교육용 프로그래밍 도구인 스크래치를 활용하였다.

3.2 모형 개발 절차

모형 개발은 <그림 1>과 같이 분석, 설계, 개발의 단계로 진행되었다.

분석 단계에서는 기존의 문제 중심 학습 모형들의 비교 분석을 통해 공통 요소를 추출하고 스크래치를 활용한 프로그래밍 수업을 위해 요구되

는 기술 및 환경 요소들을 분석하였다.



<그림 1> 모형 개발 절차

설계 단계에서는 분석 결과들을 토대로 모형 프로토타입을 설계하고 교수설계 및 문제해결교육, 프로그래밍교육 전문가 집단의 심층 인터뷰를 통해 모형을 수정·보완하였다.

개발 단계에서는 분석 및 모형 설계 결과들을 토대로 교수 학습에 활용할 실제적인 문제들을 개발하였다. 문제는 문제해결활동으로의 학습자의 적극적인 참여를 유도할 수 있도록 하였으며, 개인적인 사고와 집단 내, 집단 간의 협력적인 사고를 촉진할 수 있는 형태로 개발하였다. 문제가 개발된 이후, 해당 문제해결과정에서 학습자들이 활

용할 수 있는 다양한 학습 자료들을 개발하였으며, 모든 학습 자료는 학습자들이 문제해결과정에서 쉽게 활용 가능하고 적시에 투입될 수 있도록 개발하였다.

이러한 절차에 따라 개발된 최종 모형의 세부 내용은 <표 2>와 같다.

3.2.1 문제 인식하기

문제의 해결목표와 상황을 명확히 인식하는 단계로, 문제해결에 대한 동기 유발을 위해 다양한 발문과 자료를 활용해야 한다.

문제 제시는 프로그래밍을 통해 해결 가능한 실제적 문제를 제시하고, 문제와 관련된 다양한 프로그래밍 자료를 제시하고 해당 과제 해결을 위해 반드시 알아야 할 프로그래밍 코드를 제시한다.

3.2.2 문제해결계획 세우기

문제 중심 학습에서는 이미 알고 있는 지식만을 사용하는 것이 아니라 새로운 정보와 지식을 추가해야 해결할 수 있는 문제를 제공하기 때문에 문제해결계획을 세우는 과정은 매우 중요하다.

따라서 이 단계에서는 제시된 문제해결을 위해 프로그래밍으로 이미 구현되어 있는 것, 새롭게 구현해야 하는 것, 새로운 것을 구현하기 위한 전

<표 2> 문제 중심 스크래치 프로그래밍 수업 모형

단계	활동 내용	집단구성
문제 인식하기	<ul style="list-style-type: none"> • 동기유발 • 문제 제시 및 파악 • 기본 프로그래밍 자료 소개 	전체/ 모둠
문제해결계획 세우기	<ul style="list-style-type: none"> • 현재 알고 있는 내용 정리 • 문제를 해결하기 위해 알아야 할 내용 파악 • 문제 해결을 위한 일정, 역할 분담 계획 	모둠
탐색하기	<ul style="list-style-type: none"> • 문제 해결을 위한 관련 지식, 정보 탐색 • 문제해결계획의 검토 • 문제해결계획의 수정 	모둠
해결책 발견하기	<ul style="list-style-type: none"> • 다양한 문제 해결책 고안하기 • 최적의 프로그래밍 결과물(문제 해결책) 결정하기 	개인/ 모둠
발표 및 평가하기	<ul style="list-style-type: none"> • 해결책을 모둠별로 발표 • 자기평가와 상호평가, 과정평가와 결과평가 • 문제해결과정에 대한 자신의 참여도 및 태도에 대한 평가 	개인/ 모둠

략 등을 검토하기 위한 충분한 시간을 제공하여야 한다.

3.2.3 탐색하기

문제를 인식하고 해결계획을 세운 후, 문제해결을 위해 요구되는 지식과 정보들을 탐색하는 과정이 필요하다. 이 과정을 통해 프로그래밍과 관련된 기본 개념을 익힐 수 있을 뿐 아니라 논리적 사고력, 문제해결력과 같은 고차원적인 사고력의 신장이 가능하다.

이 단계에서 학습자들은 문제해결계획의 검토 및 수정과정을 반복한다. 이 때 교사는 탐색활동의 특성에 따라 학습집단을 구성하고 시간을 배분하여야 한다.

3.2.4 해결책 발견하기

탐색과정을 통해 발견한 지식과 정보들을 활용하여 문제 해결책을 만드는 과정이 필요하다. 이 단계에서는 문제해결을 위한 다양한 해결책을 고안한다. 여기서 해결책은 특정 문제의 해결을 위해 구현된 프로그램이나 결과물을 의미한다. 이는 프레젠테이션, 발표, 상황극, 실험 등의 다양한 형태 속에 프로그램으로 구현한 결과가 포함되어질 수 있다.

학생들이 다양한 형태의 해결책을 고안하는 과정에서 프로그래밍 활동이 포함될 수 있도록 함으로써 학습자들의 자유롭고 창의적인 사고를 제한하지 않도록 해야 한다.

3.2.5 발표 및 평가하기

학생들이 고안한 해결책을 발표하고 평가하는 과정은 다양한 해결책을 공유하고 평가하기 위한 마무리 단계로 반드시 필요하다. 해결책을 제안함과 동시에 동료들의 해결 방안 가운데 최선의 것을 선정하고 평가하는 것은 실세계에서 일어나는 의사결정 과정과 매우 유사하다. 이 단계에서는 학생들이 발표한 해결책에 대해 자기평가와 상호평가, 과정평가와 결과평가를 진행한다. 이러한 과정을 통해 학습자들은 스스로 문제해결의 전 과정을 경험하면서 어떤 점에서 어려움을 겪었으며 어느 부분에서 만족스러웠는지 등을 반성하게 된다. 이러한 활동을 통해 학생들은 문제해결과정 전체를 스스로 평가하게 되며, 이후의 학습활동에 대한 피드백을 제공받게 된다.

3.3 모형에 따른 수업 설계

해당 모형을 적용한 수업을 진행하기 위한 수업 설계는 크게 문제 설계 및 개발, 학습 목표의 확인 및 설정, 자료의 선정 및 전달, 교수 학습 계획 작성, 평가의 단계로 구분된다.

<표 3>은 수업을 위해 설계된 문제를 구체적으로 나타낸 것이다. 수업은 크게 기본 교육 및 문제 중심 학습으로 구분되며, 기본 교육은 프로그래밍의 기본 개념 및 스크래치 사용법을 익히기 위해 6차시 분량으로 구성하였으며 구체적인 교육내용은 <표 4>와 같다.

<표 3> 구체적 문제 상황

구분	주제	역할	구체적 문제 상황
문제 1	체험학습 발표자료 만들기	발표자	긴 여름 방학을 끝내고 개학을 했습니다. 방학 중에 가족과 함께 다녀온 체험학습에서 보고 들은 것을 친구들에게 소개해 주려고 합니다. 사진, 동영상, 소리, 그림 등 다양한 자료가 모아졌습니다. 어떻게 하면 보고 들은 것을 반친구들에게 효과적으로 전달할 수 있을지 생각해 봅시다.
문제 2	제 소개를 하겠습니다	소개자	새로운 학년이 시작되어 새로운 선생님과 친구들을 만났습니다. 내일은 선생님께서 반 친구들 앞에서 각자 소개하는 시간을 갖기로 하였습니다. 주어진 시간 안에 자신의 특기나 취미, 가족사항, 존경하는 인물, 혈액형 등 가능한 자신과 관련된 많은 자료를 효과적으로 설명할 수 있어야 합니다. 친구들 앞에서 자신의 이름을 가장 인상적이고 독창적으로 알릴 수 있을지 고민해 봅시다.
문제 3	작가가 되어봅시다	작가	나는 훌륭한 극본을 만들어 무대에 작품을 올려보는 것이 소원입니다. 그 꿈을 이루기 위해 다양한 이야기를 즐겨 써오고 있습니다. 원하는 무대를 배경으로 만들고 등장인물을 정해 간단한 대화가 오가는 장면을 만들어 봅시다.

<표 4> 기본 프로그래밍 교육 내용

차시	주제	하위 주제	내용 및 활동
1-2	프로그래밍의 이해 및 기초	프로그래밍의 이해	<ul style="list-style-type: none"> 프로그래밍의 개념 프로그래밍의 역할 및 도구
		문제해결전략 및 순서도	<ul style="list-style-type: none"> 다양한 문제해결 전략 탐구 순서도 필요성 및 작성 방법
		알고리즘의 이해	<ul style="list-style-type: none"> 알고리즘의 개념 알고리즘의 중요성 스크래치의 성격
3-4	스크래치 기본	스크래치 기본 사용법	<ul style="list-style-type: none"> 화면 구성 및 메뉴 사용법 스프라이트 추가 및 만들기
		스크래치 블록의 이해	<ul style="list-style-type: none"> 각 블록의 기본 역할 이해 블록 연결로 간단한 프로그램 작성
5-6	스크래치 활용	기본 움직임 및 활용	<ul style="list-style-type: none"> 기본 이동 코스튬 추가로 애니메이션 만들기 다양한 효과 주기
		스크래치 보드의 이해	<ul style="list-style-type: none"> 스크래치 보드의 구성 및 연결 각 센서의 역할 및 예제 프로젝트 분석

<표 5> 문제 1의 교수 학습 과정

단계	활동 내용	차시
문제 인식	<ul style="list-style-type: none"> 동기유발: 다양한 그래픽 효과를 준 프리젠테이션을 보여준다. 문제상황 제시: '현장학습 후 발표자료 만들기'라는 문제상황을 다양한 방법으로 제시한다. 기본적 프로그래밍 예제: 발표자료를 만드는데 필요한 기본적 알고리즘 제시하고 익힐 수 있도록 한다. 	1
문제해결 계획 세우기	<ul style="list-style-type: none"> 문제를 해결하기 위해 우리가 알고 있는 것: 이미 구현되어 활용 가능한 프로그래밍 방법을 확인 한다. 문제를 해결하기 위해 더 알아야 할 것: 새롭게 구현해야 할 프로그래밍 내용을 확인한다. 알아내는 방법 살펴보기: 역할 분담하여 새로운 알고리즘 탐구하기 	2
탐색하기	<ul style="list-style-type: none"> 다양한 프로그래밍 시도하기: 체험학습 내용을 효과적으로 발표하기 위한 다양한 프로그래밍을 실시한다. 더 알아야 할 것 추가: 추가하여 개발할 내용을 확인하며 수정해 나간다. 디버깅: 만들어진 프로그래밍 내용을 실시해보고 디버깅을 통해 최적화 한다. 	3-4
해결책 발견	<ul style="list-style-type: none"> 다양한 프로그래밍 해결책 고안하기: 개발된 해결책들을 정리한다. 최적의 프로그래밍 구현 결과 정하기: 구현된 결과물 가운데 최적의 작품을 결정한다. 	5-7
발표 및 평가	<ul style="list-style-type: none"> 해결책 발표하기: 개발된 해결책을 다양한 방법으로 발표한다. 해결책 평가하기: 발표된 해결책을 자기평가, 상호평가를 실시한다. 	8-10

<표 5>는 문제 1에 대한 상세 교수 학습 과정을 제시한 것이다.

이러한 연구 목적 달성을 위해 설정한 가설은 다음과 같다.

4. 적용 및 결과 분석

4.1 연구가설

본 연구의 목적은 초등학생을 위한 문제 중심 스크래치 프로그래밍 교수 학습 모형을 개발하고 해당 모형이 학습자의 논리적 사고력 및 문제해결성향에 미치는 영향을 분석하고자 하는 것이다.

연구 가설 1: 문제 중심 스크래치 프로그래밍 교육은 초등학교 학습자의 논리적 사고력에 유의한 영향을 미친다.

연구 가설 2: 문제 중심 스크래치 프로그래밍 교육은 초등학교 학습자의 문제해결성향에 유의한 영향을 미친다.

4.2 연구대상

본 연구의 대상은 서울에 위치한 초등학교 5학년 2개 반을 각각 통제집단(27명)과 실험집단(28명)으로 구성하였다. 모든 연구 대상은 순서도와 프로그래밍에 관한 사전지식이나 기초적 소양이 없는 집단이다.

4.3 연구설계

본 연구에서는 실험집단과 통제집단을 임의로 선정하여 실시하는 이질 통제집단 전후검사 설계를 사용하였다. 실험처치 전, 문제해결성향과 논리적 사고력에 관한 두 집단의 동질성 여부를 분석하기 위해 사전검사를 실시하였다.

이후 실험처치 단계에서는 두 집단 모두 기본적인 스크래치 사용법에 관한 사전교육을 6차시에 걸쳐 실시하였다.

이후, 실험집단을 대상으로 본 연구를 통해 개발한 문제 중심 프로그래밍 교수 학습 모형을 적용한 수업을 실시하고 통제집단을 대상으로 전통적인 프로그래밍 수업을 10차시에 걸쳐 실시하였다. 전통적인 프로그래밍 교수 학습은 문제 중심 교수법이 제거되고 단지 학습자에게 학습 내용을 전달하고 실습할 수 있는 환경을 제공하는 형태로 진행하였다.

실험처치 이후, 사후검사를 통해 문제 중심 프로그래밍 교수 학습 모형의 효과를 분석하였다.

구체적인 연구의 실험설계는 <표 6>과 같다.

<표 6> 실험설계

실험집단	O ₁	X ₁	O ₂
통제집단	O ₃	X ₂	O ₄

O₁, O₃ : 사전 검사(문제해결성향 검사, 논리적 사고력 검사)

X₁ : 스크래치 활용 문제 중심 프로그래밍 수업

X₂ : 스크래치 활용 전통적인 프로그래밍 수업

O₂, O₄ : 사후 검사(문제해결성향 검사, 논리적 사고력 검사)

4.4 연구도구

4.4.1 문제해결성향 검사

문제해결성향 검사는 Hepper와 Petersen(1982)

이 개발한 PSI(Problem Solving Inventory)를 번안하여 사용하였다[19].

이는 개인의 문제해결 행동과 태도 등에 대한 자신의 인지력을 측정하는 32개의 문항으로 각 문항은 5점 평정 척도로 구성된다. 문제해결성향의 하위요소는 ‘자신감’, ‘접근회피양식’, ‘자신의 통제’로 구성된다.

‘자신감’은 자신의 문제해결성향에 대해 얼마나 긍정적으로 생각하는지를 측정하기 위한 문항으로 구성된다. ‘접근회피 양식’의 경우, 자신의 문제를 해결할 수 있다는 자신감의 정도를 측정하기 위한 문항으로 구성되며 ‘자신의 통제’는 문제를 해결해 나가는 데에 있어서 정서적으로 얼마나 잘 통제하는지를 측정하기 위한 문항들로 구성된다. 가능한 점수 분포는 최고 160에서 최저 32이다. 본 연구의 검사지 신뢰도는 Cronbach α 값이 .90으로 양호하게 나타났으며, 구체적인 문항 구성은 <표 7>과 같다.

<표 7> 문제해결성향 검사지 문항 구성

하위요인	문항수
자신감	11
접근회피 양식	16
자신의 통제	5
계	32

4.4.2 논리적 사고력 검사

논리적 사고력 검사 도구로 Roadrangka 등(1983)이 개발한 GALT(Group Assessment of Logical Thinking) 검사지를 수정·보완하여 사용하였다 [20].

해당 검사지의 문항 수는 총 15문항으로 구체적인 문항 구성은 <표 8>과 같다.

각 문항은 답과 이유를 질문하는 2개의 하위검사문항으로 구성되며, 답과 이유 모두 맞을 경우 1점을 부여하였다. 따라서 가능한 점수 분포는 최고 15점에서 최저 0점이다. 본 연구의 검사지 신뢰도는 Cronbach α 값이 .94로 양호하게 나타났

<표 8> 논리적 사고력 검사지 문항 구성

논리 유형	문항수
보존 논리	2
비례 논리	4
변인 통제 논리	2
확률 논리	2
상관 논리	2
조합 논리	3
계	15

5. 연구결과 및 논의

5.1 문제해결성향

실험처치 전, 실험집단과 통제집단의 문제해결성향에 대한 동질성 여부 검증을 위해 실시한 사전검사 결과는 <표 9>와 같으며, 실험집단과 통제집단 사이에는 유의한 차이가 없는 것으로 나타나($p>.05$), 동질집단임이 확인되었다. 사후 문제해결성향의 경우, <표 10>에서와 같이, 실험집단의 평균($M=105.61$)이 통제집단($M=96.70$)보다 높게 나타났으며, 통계적으로 유의한 차이를 보이고 있다($p<.05$). 즉, 문제 중심 프로그래밍 수업이 전통적인 프로그래밍 수업보다 학습자의 문제해결성향에 긍정적인 영향을 주는 것으로 볼 수 있다.

<표 9> 집단별 사전 문제해결성향 분석 결과

집단	사례수	평균	표준편차	t	p
실험집단	28	95.82	11.867	.583	.563
통제집단	27	93.96	11.788		

<표 10> 집단별 사후 문제해결성향 분석 결과

집단	사례수	평균	표준편차	t	p
실험집단	28	105.61	13.847	2.512	.015
통제집단	27	96.70	12.369		

또한 각 집단별 사전·사후 문제해결성향의 향상 정도가 통계적으로 유의한지 알아보기 위해 대응표본 t 검정을 실시하였으며 그 결과는 <표 11>과 같다.

<표 11> 집단별 사전·사후 문제해결성향의 대응표본 t 검정 결과

집단	사례수	사전 문제해결성향-사후 문제해결성향			
		평균	표준편차	t	p
실험집단	28	-9.786	7.390	-7.006	.000
통제집단	27	-2.741	6.648	-2.142	.042

문제해결성향의 경우, 두 집단 모두 사전검사 결과에 비해 사후검사 결과, 문제해결성향이 향상되었지만, 통제집단의 경우 실험집단에 비해 향상 정도가 낮게 나타났다. 이러한 결과는 스크래치 활용 프로그래밍 수업이 학습자의 문제해결성향 향상에 긍정적인 영향을 준다는 것을 의미하며, 특히 문제 중심 교수 학습 모형을 적용할 경우 전통적인 교수 학습 방식에 비해 더 나은 향상을 가져올 수 있다는 것을 보여준다.

5.2 논리적 사고력

논리적 사고력의 경우, 문제해결성향과 마찬가지로 실험집단과 통제집단 사이에는 유의한 차이가 없는 것으로 밝혀졌으며($p>.05$), 그 결과는 <표 12>와 같다.

사후 논리적 사고력의 경우, <표 13>에서와 같이, 실험집단의 평균($M=6.79$)이 통제집단($M=5.04$)보다 높게 나타났으며, 통계적으로 유의한 차이를 보이고 있다($p<.05$). 즉, 문제 중심 프로그래밍 수업이 전통적인 프로그래밍 수업보다 학습자의 논리적 사고력에 긍정적인 영향을 주는 것으로 볼 수 있다.

<표 12> 집단별 사전 논리적 사고력 분석 결과

집단	사례수	평균	표준편차	t	p
실험집단	28	4.39	3.304	-.106	.916
통제집단	27	4.48	2.860		

<표 13> 집단별 사후 논리적 사고력 분석 결과

집단	사례수	평균	표준편차	t	p
실험집단	28	6.79	3.119	2.182	.034
통제집단	27	5.04	2.808		

또한 각 집단별 사전·사후 논리적 사고력의 향상 정도가 통계적으로 유의한지 알아보기 위해 대응표본 t 검정을 실시하였으며 그 결과는 <표 14>와 같다.

<표 14> 집단별 사전·사후 논리적 사고력의 대응표본 t 검정 결과

집단	사례수	사전 논리적 사고력-사후 논리적 사고력			
		평균	표준편차	t	p
실험집단	28	-2.393	1.227	-10.316	.000
통제집단	27	-.556	1.050	-2.749	.011

논리적 사고력의 경우, 문제해결성향과 마찬가지로 두 집단 모두 사전검사 결과에 비해 사후검사 결과, 논리적 사고력이 향상된 것으로 드러났다. 그러나 통제집단의 경우 실험집단에 비해 향상 정도가 낮은 편이다. 이러한 결과는 스크래치 활용 프로그래밍 수업이 학습자의 논리적 사고력 향상에 긍정적인 영향을 준다는 것을 의미하며, 특히 문제 중심 교수 학습 모형을 적용할 경우 전통적인 교수 학습 방식에 비해 더 나은 향상을 가져올 수 있다는 것을 보여준다.

6. 결 론

본 연구에서는 초등학교 학습자들의 프로그래밍 교육을 위해 교육용 프로그래밍 언어를 활용할 경우, 해당 도구의 효과성을 최대화하고 프로그래밍 학습에 대한 내적 동기를 유발하기 위한 교수 학습 모형을 제시하고자 하였다.

교육용 프로그래밍 언어의 경우, 어린 학습자나 초보 학습자들의 알고리즘 및 프로그래밍 교육을 목적으로 개발된 도구로, 보다 쉬운 사용자 인터페이스와 풍부한 멀티미디어 학습 환경을 제공함으로써 프로그래밍 학습에 관한 흥미와 동기를 유발할 수 있다. 또한 프로그래밍 언어 사용법 학습으로 인해 부가될 수 있는 과도한 인지적 부담을 감소시킴으로써 학습자가 본질적인 학습 내용에 집중할 수 있도록 도울 수 있다.

특히 스크래치는 가장 최근에 개발된 교육용 프로그래밍 언어로 초·중학생의 프로그래밍 교

육을 통한 정보유창성 신장을 목적으로 개발되었다. 해당 언어의 경우, 풍부한 멀티미디어 학습 환경을 제공하며, 프로그래밍 실행 환경 및 코딩 환경 모두 시각적인 환경을 지원할 뿐 아니라, 한국어를 지원한다. 따라서 국내의 초등학교 학습자들의 프로그래밍 교육을 위한 도구로 유용하게 활용될 수 있으며, 실제로 초등학교, 중학교 현장에서의 프로그래밍 교육에 스크래치를 활용한 경우, 학습 동기 및 문제해결력, 창의성 등에 긍정적인 영향을 보고한 연구 결과들이 제시되었다.

그러나 프로그래밍의 과정은 복잡한 문제해결 과정으로 논리적 사고 및 추상적 추론과 같은 고차원적인 인지 능력을 요구하기 때문에 여전히 초등학생과 같은 어린 학습자들에게 인지적 부담을 유발할 수 있다.

따라서 스크래치와 같은 도구 활용의 효과를 최대화하고 학습자들의 인지적 부담을 감소시키기 위한 교수 학습 방법에 관한 연구가 요구된다.

본 연구에서는 프로그래밍 학습에 대한 초등학교 학습자들의 외적 인지 부하를 감소시키기 위한 전략으로 교육용 프로그래밍 언어 중 하나인 스크래치를 활용하되 해당 도구 활용 효과를 최대화하고 프로그래밍 학습 과정을 효과적으로 지원할 수 있는 교수 학습 모형을 개발하고자 하였다. 따라서 본 연구에서는 초등학생을 위한 문제 중심 스크래치 프로그래밍 교수 학습 모형을 개발하고 해당 모형을 적용한 교수 학습 과정을 개발한 뒤, 실제 초등학교 현장에 적용하여 그 교육적 효과를 검증하였다.

실험처치 후 실시한 문제해결성향 및 논리적 사고력의 경우, 문제 중심 프로그래밍 교수 학습 모형을 적용한 실험집단이 강의 및 시범실습, 따라하기 등의 전통적인 프로그래밍 교수 학습 모형을 적용한 통제집단에 비해 유의하게 높은 것을 확인하였다.

또한, 문제해결성향 및 논리적 사고력의 사전·사후 차이 검증 결과, 두 집단 모두 유의한 향상을 보였으나, 실험집단의 향상이 더 높은 것으로 확인되었다. 이러한 연구 결과는 스크래치를 활용한 프로그래밍 학습이 초등학교 학습자의 문제해결성향 및 논리적 사고력 향상에 긍정적인 영향을 준다는 것을 의미한다. 그러나 실험집단의 향

상 정도가 통제집단의 향상 정도에 비해 높게 나타났으며, 사후검사 결과 두 집단의 각 점수가 유의한 차이를 나타낸 점을 종합해보면, 문제 중심 프로그래밍 교수 학습 방법이 전통적인 프로그래밍 교수 학습 방법에 비해 유의한 향상을 이끈 것으로 해석할 수 있다.

이러한 연구 결과들을 종합해 보면, 본 연구에서 개발한 문제 중심 스크래치 프로그래밍 교수 학습 모형은 초등학교 학습자의 논리적 사고력 및 문제해결력 향상에 긍정적인 요인으로 작용할 수 있음을 의미한다. 단, 문제해결력의 경우, 문제해결성향과 더불어 실제적인 문제해결능력을 종합적으로 검토할 필요가 있으며, 학습자의 내적 동기 수준에 미치는 영향을 면밀히 검토할 필요가 있다.

참 고 문 헌

- [1] 교육인적자원부 (2005). **초·중등학교 정보통신기술교육 운영지침**. 교육인적자원부.
- [2] 교육인적자원부 (2007). **초·중등학교 교육과정**. 제2007-79호. 교육인적자원부.
- [3] 김경훈·이원규·김성식·강신천·강의성·김영식·유현창 (2007). **중학교 교과재량활동 I (한문, 정보, 환경) 교육과정해설 연구개발**. 연구보고 CRC 2007-24. 한국교육과정평가원.
- [4] Tucker, A., Deek, F., Jones, J., McCowan, D., Stephenson, C., & Verno, A. (2003). *A Model Curriculum for K-12 Computer Science: Report of the ACM K-12 Education Task Force Computer Science Curriculum Committee*. NY: Association for Computing Machinery.
- [5] NCERT (2007). *National Council of Educational Research and Training*. <http://ncert.nic.in/welcome.html>
- [6] White, G. L. & Sivitanides, M, P. (2002). A Theory of the Relationships between Cognitive Requirements of Computer Programming Languages and Programmer's Cognitive Characteristics. *Journal of Information Systems Education*, 13(1), 59-66.
- [7] 장인에 (1997). **왜 구성주의인가?**. 서울: 문음사.
- [8] Hmelo-Silver, C. E. (2004). Problem-based learning: What and how do students learn?. *Educational Psychology Review*, 16(3), 235-266.
- [9] 이은경·이영준 (2008). Scratch 활용 프로그래밍 교육이 중학생의 몰입수준과 프로그래밍 능력에 미치는 영향. **중등교육연구**, 56(2), 359-382.
- [10] 오세인·박정호·이태욱 (2007). squeak 언어를 적용한 실업계 고등학교 프로그래밍 수업이 논리적 사고력 향상에 미치는 영향. **한국컴퓨터교육학회학술발표논문집**. 99-103.
- [11] 정미연·이은경·이영준 (2008). Squeak Etoys 활용 알고리즘 학습이 중학생의 문제해결력에 미치는 영향. **대한공업교육학회지**. 33(2), 170-191.
- [12] Malan, D. J., & Leitner, H. H. (2007). Scratch for Budding Computer Scientists. *Proceedings of the 38th SIGCSE technical symposium on Computer science education, Covington, Kentucky, USA*, 223-227.
- [13] Morrison, G. R., Lowther, D. L., and DeMeulle. L. (1999). *Integrating Computer Technology Into The Classroom*. New Jersey: Prentice Hall.
- [14] Barrows, H. S., and Myers, A. C. (1993). *Problem-based learning in secondary schools*. Unpublished monograph. Springfield, IL: Problem-based learning institute. Lanphier high school and southern illinois university medical school.
- [15] IMSA (2001). What is the relationship between problem-based learning and other instructional approaches? <http://www.imsa.edu/team/cpbl/whatis/design/slide34.html>.
- [16] Barrow, H. S., and Kelson, A. M. (1996).

Problem-based learning: total approach to education. Unpublished monograph. Springfield, IL: Southern Illinois school of Medicine.

- [17] Wood, E. J. (1993). Resource programs for the biology curriculum. *Biochemistry Education*, 21, 159-160.
- [18] 조연순 · 우재경(2003). 문제중심학습(PBL)의 이론적 기초: 지식관과 교육적 가치. **교육공학연구**. 41(3), 571-600.
- [19] Hepper, P. P., and Petersen. C. H. (1982). The development and implications of a personal problem-solving inventory. *Journal of Counseling Psychology*, 29, 66-75.
- [20] Roadranka, V., Yeany, R. H., and Padilla, M. J. (1983). *The Construction and Validation of Group Assessment of Logical Thinking (GALT)*. Paper presented at the annual meeting of the National Association for Research in Science Teaching, Dallas, Texas.

현재 청운중학교 교사

관심분야: 정보교육, 로봇프로그래밍, 학습과학

E-Mail: soph76@hitel.net



이 영 준

1988 고려대학교 전산학과
(이학사)

1994 미국 미네소타대학교
(전산학 Ph.D.)

현재 한국교원대학교 컴퓨터교육과 교수

관심분야: 정보통신교육, 지능형시스템, 학습과학

E-Mail: yjlee@knue.ac.kr



배 학 진

2001 서울교육대학교
초등교육과(교육학학사)

2009 한국교원대학교
컴퓨터교육과(교육학석사)

현재 서울묵현초등학교 교사

관심분야: 컴퓨터교육, 로봇교육

E-Mail: samhj@naver.com



이 은 경

1998 한국교원대학교
컴퓨터교육과(교육학학사)

2009 한국교원대학교
컴퓨터교육과(교육학박사)