

CPS에 기반한 스크래치 EPL이 문제해결력과 프로그래밍 태도에 미치는 효과

조성환*, 송정범*, 김성식*, 이경화**

한국교원대학교 컴퓨터교육과*, 한국교원대학교 수학교육과**

요약

프로그래밍 교육은 학생들의 창의력, 문제해결력, 논리적 사고력 향상 등에 긍정적인 영향을 끼친다. 하지만, 프로그래밍을 하기 위해서는 기본 문법과 구조를 이해하고 프로그램을 사용하는 방법을 익히는데 너무 많은 노력을 요구하는 등 문제점을 갖고 있으며, 이로 인해 학습자에게 프로그래밍에 대한 부정적인 인식을 갖게 한다. 따라서 본 연구에서는 배우기 쉽고, 직관적이며, 학습자의 문제해결력 향상을 도모할 수 있는 교육용 프로그래밍 언어(EPL)인 '스크래치'를 CPS(Creative Problem Solving) 모형을 활용해 적용해봄으로써 중학교 남녀 학습자에게 있어 문제해결력 향상과 프로그래밍 교육에 대한 태도에 어떠한 영향을 미치는지 단일집단 사전사후 검사를 통해 검증하였다.

키워드 : EPL, 프로그래밍 교육

The Effect of CPS-based Scratch EPL on Problem Solving Ability and Programming Attitude

Seong-Hwan Cho*, Jeong-Beom Song*, Seong-Sik Kim*, Kyung-Hwa Lee**

Korea National University of Education, Dept. of Computer Education*

Korea National University of Education, Dept. of Mathematics Education**

ABSTRACT

A programming education has favorable influence on creative / logical thinking and problem solving ability of students. However, students typically have to spend too much effort in learning basic grammar and the usage model of programming languages, which negatively affect their eagerness in learning. In this respect, we proposed to apply the 'Scratch' using the Creative Problem Solving(CPS) Teaching Model; Scratch is an easy-to-learn and intuitive Educational Programming Language(EPL) that helps improving the problem solving ability of the class. Then we verified the effect of Scratch EPL through the design of both pretest and posttest for a subject group. In summary, the CPS based Scratch EPL was shown to significantly improve the problem solving ability and also help them develop favorable attitude in programming.

Keywords : Educational Programming Language(EPL), Programming Education

본 연구는 한국학술진흥재단 2007년도 WISE 사업(KRF-2007- WISE 009)에 의해 지원되었음

1. 서론

21세기 지식-정보화 사회에서는 정보가 유력한 자원이 되고, 정보의 처리·가공에 의한 가치의 생산을 중요하게 여긴다. 따라서 이러한 사회에서 강조해야 할 교육내용은 스스로 문제를 해결하는 능력, 협동심, 비판적 사고력 및 창의적 사고력 등이 될 것이다[18]. 그러므로 학교교육의 패러다임도 학습자의 창의성과 문제해결력 향상 방향으로 변화되어야 할 것이 요구되어 진다. 이러한 실정에서 컴퓨터 프로그래밍 교육을 통해 학습자의 유창성, 독창성, 융통성과 같은 창의적 인지능력을 신장시킬 수 있는 시도는 바람직하다 할 수 있다.

한편, 교육부에서도 이러한 흐름을 반영하기 위해 2007년 2월 28일 새로운 교육과정을 고시하여 중학교 컴퓨터 교과에 있어서는 기존의 소프트웨어 활용 중심이 아니라, 교과명 자체를 ‘정보’로 변경하면서 컴퓨터의 과학적인 측면을 강조하였고, 초등학교에서는 2005년 개정된 ICT 교육운영 지침안을 통해 본질적인 과학으로서의 컴퓨터교육을 강조하고 있다. 특히, 학습자의 논리적인 사고력과 창의적 문제해결 능력을 신장시킬 수 있는 프로그래밍 이해와 활용에 관련된 내용을 초등학교에서는 ‘정보처리의 이해’ 영역에, 중학교에서는 ‘문제해결 방법과 절차’ 영역에 할당하고 있다[8][13].

또한, ACM(Association for Computing Machinery)에서 제시한 “A Model Curriculum for K-12 Computer Science”에서는 컴퓨터 교육이 활용 중심이 아닌 알고리즘적 사고 형성을 통한 컴퓨터 과학에 있어서의 기본적인 기술 습득과 프로그래밍 언어를 통한 교육의 필요성을 강조하고 있다[18].

한편, 프로그래밍 교육이 학습자의 논리적 사고력 증진, 문제해결 능력 신장, 완성을 통한 성취감과 자신감의 획득, 토론을 통한 협동심과 상호작용 효과와 같은 고등인지 능력을 향상시킨다는 사실은 많은 연구에서 들어난 사실이지만[1][18][19][21][23][5], 프로그래밍을 하기 위해서는 기본 문법과 구조를 이해하고 프로그램을 사용하는 방법을 익히는데 너무 많은 노력이 요구되어 진다. 따라서 학습자는 프로그램 학습 초기에 가지고 있던 의욕이나 동기를

상실하고 프로그래밍에 대한 부정적인 인식만 남게 된다[19]. 또한, 대부분의 사람들은 컴퓨터 프로그래밍은 오직 고도의 테크니컬 훈련을 받은 사람만이 가능한 특별하고 지루한 과정으로 보고 있다[6].

따라서, 이러한 문제점을 극복하고 프로그래밍 교육이 가지고 있는 문제해결력 신장, 창의력 신장, 반성적 사고력 신장 등의 장점을 얻고, 프로그래밍에 대한 태도에 긍정적인 영향을 주고자 ‘교육용 프로그래밍 언어(Educational Programming Language)인 스크래치를 활용하고자 한다. 스크래치는 미국 MIT와 UCLA의 연구자가 공동으로 개발한 EPL로서 색깔별로 구분되는 블록들을 마치 레고 블록을 짜맞추는 과정을 통해 프로그래밍을 구성하는 방식(Building-block programming)으로[34][29] 초보자도 이해하기 쉽고, 배우기 쉬우며, 매우 직관적인 프로그래밍 언어이다. 따라서 학습자들은 동기유발은 물론이고, 프로그래밍 교육에서 근본적으로 이루고자 하는 목적인 창조적으로 생각하고 문제를 체계적으로 해결하며, 다양한 방식으로 새로운 형태의 문제를 접근할 수 있는 알고리즘 사고방식을 기를 수 있다.

따라서, 본 연구에서는 스크래치 EPL을 통한 프로그래밍 교육이 문제해결력 신장과 학습자의 긍정적인 프로그래밍 태도 형성에 있어 중학교 남녀 학습자에게 어떠한 영향을 미치는지 검증하고자 한다.

2. 관련 연구

2.1 EPL과 스크래치

기존의 프로그래밍 교육은 학습하기 어렵고, 기초적인 이론 부분의 지식을 학습하기에 시간과 노력이 상당히 소요된다는 부정적인 인식으로 선호되지 않았었다[39][19][3]. 또한 프로그래밍 교육이 단순히 올림피아드 대회 준비나 실업계 고등학교의 경우는 교육과정에 편성되어 있기 때문에 개인의 요구나 흥미보다는 수동적이고 의무적으로 진행되어 온 것이 사실이다[10][25]. 이는 프로그래밍 교육이 가지고 있는 교육적 가치를 상쇄시키는 가장 큰 원인이라 말할 수 있다. 따라서 프로그래밍

교육이 갖는 장점을 살리면서 학습자의 동기를 자극해 흥미와 관심을 불러일으킬 수 있는 대안적인 프로그래밍 언어인 EPL이 등장하게 되었다. 스크래치(Scratch) EPL은 2007년 5월 매사추세츠 공과대학(MIT)의 미디어랩과 UCLA의 연구자가 미국 과학재단, 인텔의 지원하에 공동으로 개발한 교육용 프로그램 언어로, 객체와 이벤트를 이용하여 게임이나 움직이는 스토리 등을 작성할 수 있는 그래픽 프로그래밍 툴이다[37].

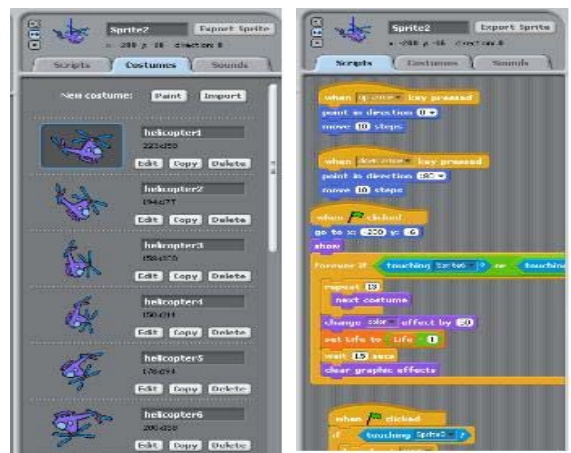
물론, 스크래치 이외에도 두리틀(Dolittle), 스킵이토이(Squeak E-toy)와 같은 EPL들이 있다. 두리틀은 2001년 일본의 가네무네가 개발한 EPL로 텍스트 기반 언어이다[2]. 특히, 한국어와 한국어 어순을 지원하여 가독성이 높다. 하지만 어려운 수준의 예약어(한자 예약어)를 알아야 하고, 동사형-명령형 예약어의 혼용으로 모호하게 해석되는 등의 문제점을 갖고 있다[20].

또한, 스킵이토이는 Smalltalk-80으로부터 발전된 EPL로서 직접 코딩을 하지 않고 아이콘을 누르거나 조작하는 drag and drop 방식이다[11][19]. 하지만, 멀티미디어 지원과 코딩 부분에 어려움이 있어 그 활용에 제한을 받는다[35].

한편, 스크래치는 아직 한국어를 지원하지 않는다는 점을 제외하면 두리틀이나 스킵같은 EPL이 제공하는 기능을 뛰어넘는 기능들을 제공한다.



(그림 1) 스크래치 화면 구성

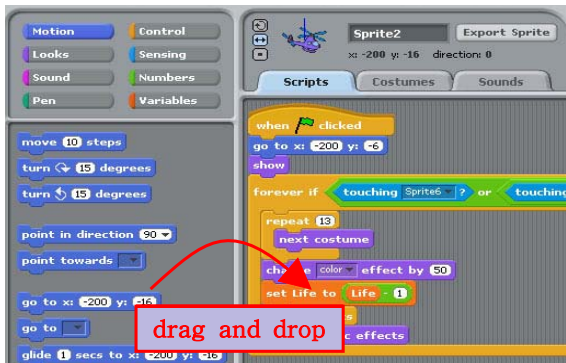


(그림 2) 스크래치 미리보기, 커스텀, 스크립트

일단, 스크래치에서는 여러 객체들을 쉽고 편리하게 정의할 수 있다. 이때의 객체를 스프라이트(sprite)라고 부른다. 스크래치에서의 프로그래밍 방식은 각각의 스프라이트에 색으로 구분되는 8개의 스크래치 블록 카테고리에서 원하는 블록 요소들을 스크립트 영역에 drag and drop 방식으로 구성한다. 각각의 블록들은 서로 끼워 맞출 수 있게 홈을 갖거나 안쪽에 또 다른 블록들을 넣을 수 있도록 구성되어 있어, 구문 에러가 발생하지 않으며 스테이지 영역을 통해 스프라이트에 프로그래밍한 내용을 즉시 확인할 수 있어서 매우 직관적이다.

또한, 각 스프라이트는 카스텀(costume)이라는 표현 형식(비트맵)과 외부 이벤트에 대한 반응과 동작을 정의한 스크립트(script)로 구성되어 있으며, 각각의 스프라이트는 다른 스프라이트와 독립적으로 행동한다. 이벤트 처리의 핵심인 스크립트는 이벤트와 그에 따른 스프라이트의 동작을 직관

적으로 정의할 수 있다. 직관성이야말로 스크래치의 최대 장점이다. 즉, 스크래치가 돋보이는 점은 객체와 이벤트 프로그래밍 과정의 직관성을 더욱 높였다는 점이다[33].



(그림 3) 스크래치 프로그래밍 방식

이외에도 스크래치가 갖는 특징으로는

첫째, 블록을 쌓는 놀이를 통하여 학습하는 프로그래밍 언어(Building-block Programming)로 어렵게 코드를 작성하지 않고 8개의 블록 카테고리에서 원하는 블록을 골라 쌓고 조립하는 것만으로 프로그래밍이 가능하다.

둘째, 풍부한 미디어의 투입이 가능하여 재미있게 학습하는 프로그래밍이 가능하다.

셋째, 뛰어난 공유성(Shareability)인해 프로그래밍 과정에서 다른 사람과 제작내용을 공유할 수 있고, 데스크탑·모바일·태블릿 등 모든 형태의 장치에서 공유가 가능하다.

넷째, 물리적 센서(모터, 전구등)와 장치를 활용하면 조작을 하면서 학습하는 프로그래밍이 가능하다.

다섯째, 다양한 언어를 지원하여 공유성을 증대시키고 사용의 편의성을 증대시킬 수 있다 [29][31][36]. 하지만, 아직 한글은 지원하지 않는다.

2.2 문제해결력과 프로그래밍

2.2.1 문제해결력의 의미

문제해결력은 학교 교육과정의 주요 교육목표

중의 하나이며, 체계적 분석, 효과적인 협력기술 등과 함께 21세기에서 반드시 알아야 할 9가지 학습 기술 중의 하나이다[32].

문제해결력의 정의에 대해서는 학자마다 다소 차이가 있지만, PISA의 문제해결 영역 전문가 위원회(PEG)에서는 “실제적이고 범교과적 문제 상황을 직면했을 때, 이를 해결하기 위하여 인지적 과정들을 활용할 수 있는 개인의 능력”이라고 정의하고 있다[14]. 또한 Mayer(1999)는 “문제해결자가 명백하게 해결 방법을 알 수 없을 때 목적 상황 속에서 현재 상황을 유도하는 인지적 처리과정”이라고 하였다[30].

결국 문제해결력은 문제 상황속에서 제시된 정보를 이해하고, 중요한 특징이나 관련성을 파악하며, 외적 표상을 구성하거나 적용하고, 문제를 해결하고 문제에 대한 해결책을 평가, 정당화하고 의사소통을 하는 등의 과정을 거치는 학습자의 인지적 활동인 것이다[14].

2.2.2 프로그래밍 학습과 문제해결력

프로그래밍은 즉각적인 피드백이 가능한 상징적인 체계 속에서, 구체적이고 엄격한 사고와 정확한 표현을 요구하며, 가설을 세우고 검증할 수 있는 학습 환경을 제공한다. 이를 Polya가 제시한 문제해결 4단계에 비추어 보면, 학습자는 프로그래밍의 과정에서 자신이 학습의 주체로서 주어진 과제를 이해·분석하고, 과제 수행을 위한 방법을 찾으며, 그 방법을 목적인대로 실행하고 수정하는 작업을 거치는 가운데 보다 논리적인 사고를 할 수 있다 [24]. 또한, 더 나아가서는 자기 자신이 어떻게 사고하는지를 탐색할 수 있기도 하다. 특히 프로그래밍과 관련된 문제해결 과정에서 학습자가 주체가 되어서 행하는 디버깅과 같은 오류수정 과정은 사고력 향상의 기회를 제공할 수 있고, 학생 자신의 행동에 대한 통찰이 가능하도록 할 수 있다. 프로그래밍에서의 오류란 틀린 것이라기보다는 수정될 수 있는 것이라고 인식되어야 하며, 오류수정의 과정이 중요한 학습의 기회로 인정되어야 한다[16]. 한편, 2003년 10월에 ACM(Association for Computing

Machinery)에서 제시한 컴퓨터 과학 교육과정에 관한 최종 보고서에 따르면 초·중등 각 학년을 4단계로 나누고, 알고리즘, 컴퓨터 구조, 운영체제 등의 원리를 이해하기 위한 컴퓨터 과학교육 과목을 제시하고 있으며 특히 프로그래밍 언어 과목을 강조하고 있다. 또한 ACM이 제시한 교육 과정에서는 레벨 1에 해당하는 8학년(우리나라 중학교 2학년에 해당) 학생들에게 컴퓨터 과학교육의 기본적인 목적인 알고리즘적 사고 형성과 문제해결력 습득을 위하여 교육용 프로그래밍언어(EPL)를 사용할 것을 제안하고 있다[26].

이상을 정리하면, 일반적으로 프로그래밍은 컴퓨터를 사용하는 학생들에게 인지적 사고력과 문제해결력을 향상시켜 줄 수 있는 효과적인 학습 환경을 제공해 주며, 나아가 문제해결력을 기르며 논리적 사고력과 창의력을 개발할 수 있는 도구로서의 가치를 가진다[15]고 할 수 있다.

2.3 프로그래밍에 대한 태도

태도란 특정의 사물, 사람, 또는 관념 등에 대하여 개인이 가지고 있는 긍정적이거나 부정적인 느낌을 말한다. 학습자의 태도는 타고난 것이 아니라 경험과 학습에 의하여 바뀌어 질 수 있는 것이기는 하지만, 일반적으로 한 번 습득된 태도는 비교적 지속성이 있는 것으로 간주된다. 따라서 프로그래밍에 대한 태도가 긍정적일수록 프로그래밍을 이용한 학습의 학습성취도가 높고, 이러한 태도는 경험에 의하여 학습되어지는 것이다. 그러므로 프로그래밍 학습의 효과를 높이기 위해서는 학습자들에게 프로그래밍에 대한 긍정적 태도를 형성할 수 있는 학습 환경을 제공하여야 한다.

이는 곧 프로그래밍 태도가 사용하는 프로그래밍 언어의 질과 가장 밀접한 관계가 있음을 뜻하며, 스크래치 EPL과 같이 프로그래밍에 대한 긍정적 태도를 형성할 수 있는 우수하고 질이 높은 프로그래밍 언어를 사용함으로써 프로그래밍에 대한 태도가 향상 될 수도 있음을 알 수 있다[4].

3. 스크래치 프로그래밍 학습을 위한 교수 설계

3.1 CPS 교수-학습 모형

창의적 문제해결력 신장을 위한 교수-학습 모형으로 가장 널리 알려진 것은 창의적 문제해결(Creative Problem Solving) 모형이다. 60년대 Osborn과 Parnes에 의해 개발되었으며, 특히 영재 학생들 대상의 창의성 신장에 대한 연구로부터 형성되기 시작하였다[17]. 이를 다시 Isaksen과 Treffinger가 6단계로 구성된 새로운 CPS 모형을 제시하였고, 각 단계는 확산적 사고(여러 가지 선택 사항 생각해내기)와 수렴적 사고(한 가지 가능성에 초점 맞추기)로 구성되어 있다[6].

Isaksen과 Treffinger가 제시한 CPS 모형 6단계를 정리하면 <표 1>과 같다.

CPS 수업 모형의 특징은 6단계로 이루어져 있으나 매 단계를 반드시 거쳐야 되는 것은 아니다. 필요에 따라서는 생략을 할 수도 있는 것이다. 그리고 각 단계마다 다양한 사고 기법을 적용하여 확산적 사고와 수렴적 사고를 거치기 때문에 상호작용이 유용하고 독창적인 해결안을 이끌게 한다[7]. 따라서, 컴퓨터 프로그래밍 학습이 다음과 같은 특징을 가지고 있기 때문에 CPS 수업 모형을 적용함은 타당하다고 본다.

첫째, 프로그래밍 학습은 문제해결을 위한 체계적이고 논리적인 접근을 요구하게 된다. 따라서, 학습자는 문제해결에 앞서 다양한 가능성을 탐색하고 그 중 가장 효율적인 방법을 선택, 실행하게 된다.

둘째, 프로그래밍 학습은 학생과 학생간, 교사와 학생간의 상호작용을 활발하게 해준다[33]. 이 과정을 통해 학습자는 개인 및 상호간에 다양한 아이디어를 공유하여 유용한 아이디어를 찾아낼 수 있는 가능성을 가진다.

셋째, 프로그래밍 학습을 통해 학습자는 계획한 대로 결과물이 산출되지 않으면, 프로그래밍 단계를 재분석해 보고, 오류를 검토해 수정하는 등의 끊임없는 인지체계를 재조정해 나간다.

마지막으로 프로그래밍은 현실세계와 유리된 교육이 아닌 통합적인 사고를 바탕으로 한 현실세계의 맥락을 중요시하며, 프로그래밍 과정 중 부딪히게 되는 문제들에 대한 해결의지와 해결에 대한 만

<표 1> 창의적 문제해결(Creative Problem Solving) 모형

CPS 단계	확산적 단계	수렴적 단계
혼란발견	문제 해결의 기회 탐색하기	문제해결을 위한 광범위하고 일반적인 목표 설정하기
자료발견	다양한 관점으로 문제를 보고, 상세한 세부사항 검토하기	문제해결을 위한 가장 중요한 자료 결정하기
문제발견	여러 방법으로 문제를 진술해 보기	진술한 것들 중 특정한 것을 고르기
아이디어 발견	많고 다양한, 그리고 독특한 아이디어 생성하기	흥미 있고, 유용하고, 그리고 잠재성을 지닌 아이디어 선정하기
해결안 발견	가장 가능성 있는 아이디어에 대한 평가 준거를 만들고, 대안을 정교화하기	가능성 있는 아이디어를 준거에 따라 선택, 촉진, 지원하기
수용안 발견	아이디어 실행에 있어 지지/저항을 할만한 것을 찾아보고, 실천을 위한 계획 짜기	구체적인 실천 계획 짜기

족감을 얻을 수 있다[18].

3.2 교수-학습 내용

본 연구에서 스크래치를 활용하여 프로그래밍 학습을 하는데 필요한 학습 내용은 미국의 K-12 컴퓨터과학 교육과정 모델과 초·중등학교 정보통신 기술 교육 운영지침 개정안 및 해설서, 개정된 중학교 “정보” 교육과정을 바탕으로 교육 내용을 재구성하였다.

<표 2> K-12 프로그래밍 언어 관련 학습내용[19]

프 로 그 래 밍	Written 활동
	<ul style="list-style-type: none"> 수학적이고 논리적인 표현의 결과 (정수형, 문자형, 소수점 등)를 평가하고 우선순위를 안다. 소스부터 실행 가능한 형태까지 구문적, 의미적 에러들을 포함한 플로우 차트를 그릴 수 있다. 주어진 문제를 하향식(top-down)을 이용해서 코드로 바꿀 수 있다. 객체지향 프로그래밍의 장점을 안다.
	Lab 활동

<ul style="list-style-type: none"> 다양한 데이터 타입을 표현할 수 있다. 프로그램을 코딩하고 실행할 수 있다. 연산자를 수행하고 출력하는 프로그램을 작성한다. 프로그램에서 구조화된 프로그램 (순서, 선택문, 반복문)의 구조를 알 수 있다.

<표 2>에서 제시하고 있는 K-12 프로그래밍 언어 관련 학습내용은 Written 활동과 Lab 활동으로 구성되어 있다. Written 활동은 학습자가 실습을 해보기 전에 알아야할 내용과 실습이 끝난 후 학습이 잘 되었는지를 체크하는 활동으로 구성되어 있고, Lab 활동은 학습자가 실습을 해봄으로써 직접 습득하게 되는 내용들로 구성되어 있다[19].

또한, 2005년 12월에 개정된 초·중등학교 정보통신기술 교육운영 지침안에 소개된 프로그래밍 관련 교육내용과 2007년 교육인적자원부가 고시한 중학교 교육과정에서 프로그래밍 관련 교육내용을 정리하면 <표 3>, <표 4>와 같다[1][22].

<표 3> 초-중등학교 정보통신기술 교육운영지침

영역	정보처리의 이해
3단계	● 프로그래밍의 이해와 기초
4단계	● 입·출력 프로그래밍
5단계	● 프로그램 제작 과정의 이해

<표 4> 중학교 “정보”교과 교육과정 내용 체계

영역	문제해결 방법과 절차
1단계	○ 문제와 문제 해결 과정 <ul style="list-style-type: none"> ● 문제의 분석과 표현 ● 문제 해결과정 ○ 프로그래밍의 기초 <ul style="list-style-type: none"> ● 변수의 개념과 활용 ● 자료의 입력과 출력 ● 제어문의 이해

한편, 송기상 외(2000)가 중학생의 논리적 사고력과 컴퓨터 학습의 관계에 관한 연구에서 중학교 1학년 상당수의 학생들이 형식적 조작기에 이르지 못하고 있어 추상화된 내용을 쉽고 구체적인 실제 사물이나 상황을 들어 전달하는 것이 학생들의 이해를 도울 수 있다[9]고 밝힌 점을 고려하여 학습자가 가장 관심 있어 하는 게임 제작과 관련한 내용으로 교육 내용을 구성하였다.

세부 학습주제 및 내용 요소는 다음과 같다.

<표 5> 스크래치 EPL 학습 내용

주제	교육과정 관련 내용요소
헬리콥터 움직이기	<ul style="list-style-type: none"> 문제와 문제 해결 과정 프로그래밍의 기초(반복문)
미사일 발사하기	<ul style="list-style-type: none"> 알고리즘의 실제 정보와 자료구조 멀티미디어 정보의 표현
미사일 사라지기	<ul style="list-style-type: none"> 프로그래밍의 기초 (조건문과 반복문) 문제 해결 방법과 절차
괴물 나타나고/ 움직이기	<ul style="list-style-type: none"> 정보와 자료구조 알고리즘의 실제 프로그래밍의 기초 (변수 사용, 자료의 입출력)
미사일이 괴물에 닿았을 경우	<ul style="list-style-type: none"> 네트워크의 이해 멀티미디어 정보의 표현 자료의 탐색
점수 설정 및 레벨 설정	<ul style="list-style-type: none"> 자료의 정렬 알고리즘의 실제 문제와 문제 해결 과정
나만의 게임 만들기	<ul style="list-style-type: none"> 문제와 문제 해결 과정 알고리즘의 개요 알고리즘의 실제 프로그래밍의 기초 멀티미디어 정보의 표현 네트워크의 이해

학습 활동은 하나의 학습주제를 완성하기 위해 학습자가 다양한 문제해결 전략과 방법을 활용하고 비교할 수 있도록 허용하였다. 이는 학습자가 다양한 방법을 통해 문제를 분석하고 이해하여, 효율적이고 창의적인 알고리즘 해결방법을 설계 및 구현할 수 있도록 한 것이다.



(그림 4) 실험수업 및 학습자 활동

4. 연구 방법

4.1 연구대상

본 연구의 대상은 한국교원대학교 WISE 충북지역 센터의 도움을 받아 방학 중 실시하는 IT캠프에 참여를 희망하는 청원군 지역 소재 O 중학교 1학년 28명으로 구성되었고, 남녀 비율은 남학생이 13명, 여학생이 15명 이었다. 실험 수업은 CPS 교수 모형을 적용하여 스크래치 EPL로 16차시 분량의 프로그래밍 수업을 진행하였다.

4.2 연구 설계

본 연구는 CPS에 기반한 스크래치 EPL 수업이 학습자의 창의적 문제해결 능력과 프로그래밍에 대한 태도에 어떠한 영향을 미치는지 또한, 남녀 성별에 있어 창의적 문제해결력과 프로그래밍에 대한 태도는 차이가 있는지 알아보기 위해, 단일집단 사전 사후 검사 설계 방법을 적용하였으며 구체적인 연구의 실험 설계는 다음과 같다.

실험집단	남	O ₁	X ₁	O ₂
	여			

O₁ : 사전 검사 (문제해결력 검사, 프로그래밍에 대한 태도 검사)

X₁ : CPS 학습 모형을 적용한 스크래치 EPL

O₂ : 사후 검사 (문제해결력 검사, 프로그래밍에 대한 태도 검사)

4.3 연구 도구 및 절차

4.3.1 문제해결력 검사도구

문제해결력 검사를 위한 도구로는 2003년에 OECD가 실시한 PISA(Programme for International Student Assessment) 의 문제해결력 소양 영역의 공개문항 19문항을 본 연구의 목적에 맞게 수정·보완하여 12문항을 선정하였다. 선정된 문항은 컴퓨터 전공 중등 교사(전문가 집단)의 타당도를 검증받았고, Pilot Test를 통해 신뢰도를 분석한 결과 Cronbach $\alpha = .723$ 이었다. 문항별 채점은 정답의 경우 10점, 오답의 경우 0점을 부여하였으며, PISA에 제시된 채점기준에 따라 부분점수 5점을 부여하였다. PISA의 문제해결 소양 영역의 문항 평가들은 <표 6>과 같고[14], 본 연구에서 사용한 문제해결력 측정 문항은 <표 7>과 같다.

<표 6> PISA 2003 문제해결 영역 평가틀

구성요소	하위요소
문제 유형	의사결정, 체제 분석 및 설계, 문제점 해결
문제 상황	교실 상황이나 학교 교육과정과 무관한 상황을 선정 (개인, 학교, 일과 여가, 지역사회와 학교)
학문 영역	폭넓은 교과 영역을 포함 (수학, 과학, 문학, 기술, 사회학 등등)
문제해결 과정	문제 이해, 문제 특성화, 문제 표상화, 문제 해결, 문제 숙고, 문제 해결책 의사소통
추론 기술	분석적, 정량적, 유추적, 종합적 추론

<표 7> 문제해결력 측정 문항

문제형태	문항번호(유형)
의사결정	7번(OC), 8번(M), 9번(M)
체제 분석 및 설계	1번(OC), 2번(OC), 3번(M), 4번(M) 5번(OC), 6번(OC)
문제점 해결	10번(OC), 11번(M), 12번(OC)

*유형: 선택형(M), 개방형 서술형(OC), 폐쇄형 서술형(CC)

4.3.2 프로그래밍 태도 검사도구

프로그래밍 태도 측정을 위한 검사도구는 우영애(2007)가 개발한 프로그래밍 교과에 정의적 태도 영역을 검사하는 검사지를 본 연구의 목적에 맞게 수정 및 보완하여 사용하였다. 문제해결력 검사 문항과 마찬가지로 전문가 집단의 타당도를 검증 받았고, Pilot Test 결과 신뢰도는 Cronbach $\alpha = .871$ 이었다. 검사문항은 수행의지, 학습 습관, 흥미, 도구가치 4개의 하위 영역별로 구성되어 있으며, 각 영역 당 5 문항씩 모두 20문항으로 구성하였고, 각 문항은 Likert 방식에 따라 5점 평점척도를 부여하도록 하였다[12].

4.3.3 자료 분석

본 연구의 통계적 분석을 위해 한글 SPSS 12.0 for windows를 사용하였으며, 구체적인 분석 방법으로 CPS에 기반한 스크래치 EPL 교육이 남학생, 여학생 실험집단의 문제해결력과 프로그래밍 태도에 어떠한 영향을 미치는지 알아보기 위해 사전-사후 검사 평균 및 표준편차를 산출하고 이것이 통계적으로 유의한 차이가 있는지 독립표본 t 검증과 대응표본 t 검증을 각각 실시하였다.

5. 연구 결과

5.1 사전 검사 결과

실험집단의 문제해결력과 프로그래밍 태도에 대한 사전 검사를 실시한 결과 <표 8>에서처럼 문제해결력에 있어서 전체 평균은 83.75점이고 성별에 따른 차이는 남자 집단은 평균이 82.31점, 여자 집단은 평균이 85.00점으로 두 집단간에 통계적으로 유의미한 차이를 나타나지 않았다($p > .05$). 또한, 프로그래밍 태도에 대한 사전 검사에 있어서, <표 9>에서와 같이 전체 평균 점수는 56.50점이고, 남자 집단은 62.00점, 여자 집단은 51.73점으로 두 집단간에 유의미한 차이를 나타내었다($p < .05$). 이는 여자집단이 남자집단에 비해 프로그래밍에 대한 태도가 다소 약한 것으로 해석할 수 있다.

<표 8> 문제해결력 사전검사 결과

집단별	N	M	SD	t	df	P
남	13	82.31	17.783	-.450	26	.657
여	15	85.00	13.887			

<표 9> 프로그래밍 태도 사전검사 결과

집단별	N	M	SD	t	df	P
남	13	62.00	6.455	3.055	26	.005
여	15	51.73	10.505			

5.2 사후 검사 결과

실험 처치 후, 문제해결력과 프로그래밍에 태도 사후 검사 결과는 <표 10>에서처럼 문제해결력 전체 평균점수가 91.50점이고, 성별에 따라서는 남자 집단이 89.38점, 여자 집단이 93.33점으로 두 집단간에 유의미한 차이를 나타내지 않았다($p>.05$). 프로그래밍에 대한 태도에 있어서도 <표 11>에서와 같이 전체 평균점수가 65.71점이고, 남자 집단과 여자 집단의 평균 점수는 각각 69.54점과 62.40점으로 두 집단간 유의미한 차이를 나타내지 않았다($p>.05$).

<표 10> 문제해결력 사후검사 결과

집단별	N	M	SD	t	df	P
남	13	89.38	16.641	-.585	26	.563
여	15	93.33	18.741			

<표 11> 프로그래밍 태도 사후검사 결과

집단별	N	M	SD	t	df	P
남	13	69.54	8.048	1.938	26	.064
여	15	62.40	10.953			

한편, 문제해결력과 프로그래밍 태도에 대한 사전-사후 차이를 검증해본 결과 <표 12>에서처럼 문제해결력에 있어 실험집단 전체적으로 유의미한 차이를 나타냈고($p<.05$), 남자 집단과 여자 집단 각각에 있어서도 통계적 유의수준에서 문제해결력의 향상을 보이고 있다. 프로그래밍 태도에 있어서도 <표 13>에서와 같이 실험집단 전체적으로 프로그래밍에 대한 태도가 긍정적으로 향상되고 있음을 보이고 있고($p<.05$), 남자 집단과 여자 집단 각각에 있어서도 프로그래밍에 대한 태도의 향상이 통계적으로 유의미한 차이를 나타내고 있다($p<.05$). 특히, 여자집단의 경우 프로그래밍 태도에 있어서 남자집단보다 사전-사후의 폭이 크다.

프로그래밍에 대한 태도가 긍정적으로 향상되고 있음을 보이고 있고($p<.05$), 남자 집단과 여자 집단 각각에 있어서도 프로그래밍에 대한 태도의 향상이 통계적으로 유의미한 차이를 나타내고 있다($p<.05$). 특히, 여자집단의 경우 프로그래밍 태도에 있어서 남자집단보다 사전-사후의 폭이 크다.

<표 12> 문제해결력 사전-사후 차이 검증

집단별	N	M	SD	t	df	P
전체	28	-7.750	8.864	-4.627	27	.000
남	13	-7.077	6.409	-3.981	12	.002
여	15	-8.333	10.748	-3.003	14	.009

<표 13> 프로그래밍 태도 사전-사후 차이 검증

집단별	N	M	SD	t	df	P
전체	28	-9.214	4.795	-10.169	27	.000
남	13	-7.538	4.075	-6.671	12	.000
여	15	-10.667	5.024	-8.223	14	.000

6. 결론 및 제언

프로그래밍 교육은 학습자의 창의력, 문제해결력, 논리적 사고력 등에 긍정적인 영향을 끼친다. 하지만, 기존의 프로그래밍 교육과 로봇 프로그래밍, 두리틀 같은 EPL 등은 학습의 난해함과 경제적인 부담감, 사용 인터페이스의 부적절함 등의 문제점들을 갖고 있다. 따라서 이러한 문제점들을 해결할 수 있고 학습자의 알고리즘 사고능력과 문제해결 능력을 신장시킬 수 있는 스크래치 EPL은 프로그래밍 교육의 적절한 접근법이라 할 수 있다.

본 연구에서는 CPS에 기반한 스크래치 EPL이 중학교 학습자의 문제해결력 신장과 프로그래밍에 대한 긍정적 태도 형성에 기여하는 효과를 검증하였다. 또한 남녀 학습자에 있어서 스크래치 EPL이 어떠한 효과차이를 나타내는지 살펴보았다.

연구 결과 첫째, CPS에 기반한 스크래치 EPL은 중학교 학습자의 문제해결력에 긍정적 영향을 미치고 있음을 알 수 있었다.

둘째, 프로그래밍에 대한 태도에 있어서도 스크래치 EPL이 기존의 부정적인 이미지를 상쇄시키고 프로그래밍에 대한 긍정적 태도를 형성할 수 있는 학습 환경을 제공함을 알 수 있었다.

셋째, 남학생과 여학생은 문제에 접근하는 방식이나 선호하는 활동 방식 등의 차이가 있지만, 본 연구에서는 그러한 성별 차이가 사전 프로그래밍에 대한 차이를 제외한 스크래치를 통한 문제해결력과 프로그래밍 태도에 있어서는 유의미한 차이를 나타내지 않았다. 이러한 점은 많은 선행연구에서 지적한 내용과 같다[27][28][38].

단, 본 연구의 실험 처치가 짧은 기간에 집중적으로 이루어졌으며, 실험 대상의 수가 소수에 불과하고 단일 집단에 대한 실험을 통해 얻은 결과이므로 본 연구에서 얻은 연구 결과를 일반화하기에는 다소 무리가 있을 것으로 사료되고, 향후 장기적인 기간과 적절한 실험 대상을 통해 다양한 관점에서 스크래치 EPL의 효과성을 검증할 필요가 있을 것이다.

참고문헌

[1] 교육인적자원부(2007), 중학교 교육과정, 교육인적자원부 고시 제2007-79호 [별책 3].
 [2] 길혜민(2004), 중등교육에 있어서 객체지향형 EPL ‘두리틀’의 적용, 고려대학교 교육대학원 석사학위논문.
 [3] 김경아(2006), 두리틀과 비주얼베이식의 학습효과 비교, 고려대학교 교육대학원 석사학위논문.
 [4] 박동호(1997), 저작시스템 프로그래밍 학습이 문제해결력 및 컴퓨터에 대한 태도에 미치는 영향, 한양대학교 교육대학원 석사학위논문.
 [5] 박미화(2006), 목표모형에 기반한 중등 일반계 프로그래밍 교육과정 개발, 한국교원대학교 석사학위논문.
 [6] 박영임(2006), 창의적 문제해결 수업모형이 아동의 창의성 신장에 미치는 효과, 부산교육대학교 교육대학원 석사학위논문.
 [7] 박현주(1999), Treffinger의 창의적 문제해결 수업모형이 아동의 창의성 및 자기존중감에 미

치는 효과, 한국교원대학교 석사학위논문.
 [8] 백영균,우인상(1994), LOGO 프로그래밍의 수업방법이 문제해결력에 미치는 효과에 관한 연구, 교육공학연구 논문지, 9(1).
 [9] 송기상 외(2000), 중학생의 논리적 사고력과 컴퓨터 학습과의 관계, 정보교육학회논문지, 5(2).
 [10] 엄수빈(2003), 컴퓨터 프로그래밍 교육의 실효성 분석 및 개선방안에 관한 연구, 상명대학교 교육대학원 석사학위논문.
 [11] 오세인(2007), Squeak 언어를 적용한 실업계 고등학교 프로그래밍 수업이 논리적 사고력 향상에 미치는 영향. 한국교원대학교 석사학위논문.
 [12] 우영애(2007), PERC 적용 수업이 수학 및 프로그래밍 학습태도에 미치는 영향 : Visual Basic 중심으로, 강원대학교 석사학위논문.
 [13] 유인환(2005), 창의적 문제해결력 신장을 위한 로봇 프로그래밍의 가능성 탐색, 교육과학연구 논문지, 36(2).
 [14] 이미경(2004), OECD/PISA 2003 평가틀 및 예시문항 : 수학/과학/문제해결력 소양 영역, 한국교육과정평가원 연구자료 ORM 2004-25-1.
 [15] 이성근(2003), 문제해결력 향상을 위한 웹 기반 프로그래밍 학습 시스템, 대구교육대학교 교육대학원 석사학위논문.
 [16] 이유순(1995), 논리적 사고력 및 문제해결력 신장을 위한 컴퓨터 프로그래밍 교육 : 베이직(Basic), 로고(Logo) 프로그래밍 비교 연구, 이화여자대학교 교육대학원 석사학위논문.
 [17] 이종연 외(2007), 창의적 문제해결 모형 기반 초등학교 사회과 수업의 효과성 분석, 교육공학연구 논문지, 23(2).
 [18] 이좌택(2004), 문제기반학습에 터한 로봇제어 프로그래밍 수업이 중학생의 논리적 사고력에 미치는 효과, 한국교원대학교 박사학위논문.
 [19] 채유진(2006), 컴퓨팅 교육을 위한 교육용 프로그래밍언어 두리틀, 스크의 비교분석, 고려대학교 교육대학원 석사학위 논문.
 [20] 최혜심(2004), 객체지향형 교육용 프로그래밍언어 두리틀의 다중 예약어 지원 체계, 고려대학교 석사학위논문.

- [21] 한건우(2007), 프로그래밍 교육에서 문제해결력 신장을 위한 동료 에이전트 시스템 개발, 한국교원대학교 박사학위논문.
- [22] 한국교육학술정보원(2005), 창의적 문제해결력 증진을 위한 Creative Thinker 프로그램 연구, 연구보고서 KR 2005-33.
- [23] 한국교육학술정보원(2005), 초·중등학교 정보통신기술 교육 운영지침 개정안 및 해설서, 연구보고서 RM2005-51.
- [24] 허영주(1991), 컴퓨터 프로그래밍 학습이 수학적 문제해결력에 미치는 효과에 대한 연구, 동국대학교 석사학위논문.
- [25] 황진욱(2006), 컴퓨터 프로그래밍 교육의 중요성 및 효과적인 교육방법 연구, 아주대학교 석사학위논문.
- [26] ACM(2003). A Model Curriculum for K-12 Computer Science. [On-line], <http://www.acm.org/education/curric_vols/k12final1022.pdf>
- [27] CHEOL HYUN JEONG(2001), Gender Differences in Computer Attitudes: New Evidence From Korea, International Review of Public Administration, 6(2).
- [28] Heidi Schelhowe(2006), Gender Questions and Computing Science, Proceedings of the international symposium on Women and ICT: creating global transformation CWIT '05
- [29] Maloney, J., Burd, L., Kafai, Y., Rusk, N., Silverman, B., and Resnick, M. (2004). Scratch: A Sneak Preview. Second International Conference on Creating, Connecting, and Collaborating through Computing. Kyoto, Japan, pp. 104-109.
- [30] Mayer(1999), Problem Solving in Encyclopedia fo CREATIVITY(vol. 2, 437-447). San Diego: Academic Press.
- [31] MIT-Media Lab(2007), Programming with Scratch, [online], <<http://scratch.mit.edu/files/Programming-with-Scratch.pdf>>
- [32] Partnership for 21st Century Skills(2003). Learning for the 21st Century. [on-line], <http://www.21stcenturyskills.org/downloads/P21_Report.pdf>
- [33] Peppler, K., & Kafai, Y. (2005). Creative coding: The role of art and programming in the K-12 educational context. [online], <<http://scratch.mit.edu/pages/research>>
- [34] Resnick, M. (2007). All I Really Need to Know (About Creative Thinking) I Learned (By Studying How Children Learn) in Kindergarten. Proceedings of the SIGCHI Conference on Creativity and Cognition, Washington, D.C.
- [35] Resnick, M., and Silverman, B. (2005). Some Reflections on Designing Construction Kits for Kids. Proceedings of Interaction Design and Children conference, Boulder, CO.
- [36] Scratch Homepage, [online], <<http://scratch.mit.edu>>
- [37] Stefanie Olsen(2007), MIT-UCLA develop programming language for kids, [online], <http://www.news.com/8301-10784_3-9719468-7.html>
- [38] Sylvia Beyer, Kristina Rynes, Julie Perrault, Kelly Hay, Susan Haller(2003), Gender Differences in Computer Science Students, Proceedings of the 34th SIGCSE technical symposium on Computer science education SIGCSE '03.
- [39] Tony Jenkins(2002), On the Difficulty of Learning to Program, [online], <<http://www.psy.gla.ac.uk/~steve/localed/jenkins.html>>

저자소개



조 성 환

1997 춘천교육대학교
(교육학학사)
2007 한국교원대학교
컴퓨터교육과(교육학석사)

2007~현재 한국교원대학교 컴퓨터교육과
박사과정

관심분야 : 컴퓨터교육, 정보통신윤리교육,
프로그래밍교육

E-Mail : 74csh@hanmail.net



이 경 화

1990 서울대학교 수학교육과
(이학사)
1993 서울대학교 수학교육과
(교육학석사)

1996 서울대학교 수학교육과(교육학박사)

1997~2003 청주교육대학교 수학교육과 교수

2003~현재 한국교원대학교 수학교육과 교수

관심분야 : 확률교육, 통계교육, gender difference

E-Mail : khmath@knue.ac.kr



송 정 범

1998 공주교육대학교
(교육학학사)
2007 공주교육대학교
컴퓨터교육과(교육학석사)

2007~현재 한국교원대학교 컴퓨터교육과
박사과정

관심분야 : 컴퓨터교육, 프로그래밍 교육,
교수·학습 설계

E-Mail : qjatns@paran.com



김 성 식

1977 고려대학교 경영학과
(경영학사)
1986 미국 카톨릭대학교
전산학과(이학사)

1988 오리곤 주립대학교 전산학과(이학석사)

1992 고려대학교 컴퓨터과학과(이학박사)

1992~현재 한국교원대학교 컴퓨터교육과 교수

관심분야 : 컴퓨터교육, 원격교육, 정보통신윤리교육

E-Mail : seongkim@knue.ac.kr