

스캐폴딩 기반 학습 전략이 스크래치 프로그래밍 학습태도에 미치는 영향

김승연*, 정인기**

원주일산초등학교*, 춘천교육대학교 컴퓨터교육과**

요약

프로그래밍 교육은 컴퓨터과학의 중요성이 부각됨에 따라 필요성이 증대되고 있다. 그러나 기존의 컴퓨터 교육은 문제 해결 과정보다는 상용 프로그램의 사용법에 치우쳐 있었다. 또한 초등학교에서의 프로그래밍 교육은 학생들의 수준에 맞지 않거나 흥미를 유발하지 못하였다. 본 연구에서는 초등학교 학생들에게 스크래치 프로그래밍 언어로 프로그래밍을 교육할 때 스캐폴딩 기반 학습 전략을 적용한 후 학생들의 프로그래밍 학습 태도를 관찰하였다. 스캐폴딩 기반 학습 전략을 적용한 결과 학생들은 프로그래밍에 대한 학습 태도에서 일반 학습 경향성, 학습을 통한 성과의 만족도 및 상호작용 항목에서 모두 향상된 결과를 보여주었다.

키워드 : 스크래치, 프로그래밍 학습, 스캐폴딩, 학습 태도

The Scratch Programming Learning Attitude Effects of Scaffolding based Learning Strategy

Seung-Yeon Kim*, In-Kee Jeong**

Wonju Ilsan Elementary School*,

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

ABSTRACT

Necessity of programming education is demanded according to more interested in importance of computer science. However, the existing programming education only focused on function for utilization of the commercial programs. The existing programming education doesn't meet the students' level and doesn't cause that they were interested in programming. Therefore, we applied the learning strategy based the scaffolding when we taught the scratch programming language in the elementary school and we studied the students' attitude towards programming learning. As a result, we found the good results about the general learning patterns, the performance satisfaction through learning and the interaction areas.

Keywords : Scratch, Programming Learning, Scaffolding, Learning Attitude

논문투고 : 2010-04-01

논문심사 : 2010-11-29

심사완료 : 2010-12-03

1. 서 론

1.1 연구 목적 및 필요성

21세기는 지식·정보화, 세계화 시대이다. 즉, 지식과 정보의 중요성이 부각되어지고 있다. 현재는 지식과 정보의 양이 5년 주기로 두 배씩 증가하고 있는 추세이다. 이어 2020년에 이르면 지식은 매 73일마다 두 배로 증가될 것으로 예상하고 있다[15]. 이러한 현실 속에서 정보의 질은 매우 중요한 요소가 되며, 누가 어디서 중요한 가치의 정보를 가지고 있는가가 부의 상징이 되고 있다. 제7차 교육과정에서는 21세기의 세계화·정보화 시대를 주도할 자율적이고 창의적인 한국인 육성을 기본방향으로 하고 있다.

지식과 정보를 활용하는 능력을 고루 갖춘 정보인을 기르는 것을 목적으로 하고 있는 교육과정 아래 프로그래밍 학습은 창의력, 문제해결력, 논리적 사고력 등 알고리즘적 사고력을 길러줄 수 있는 효과적인 분야로 분류되고 있지만 현재의 교육 상황에서 컴퓨터 교육의 비중이 많이 줄어들고 있다. 현재 학교에서 이루어지는 컴퓨터 교과는 고등사고력을 요하는 컴퓨터 과학적 측면이 아닌 단지 기능적 향상을 돕는 도구적 교과로 전락하고 있다.

알고리즘적 사고력을 향상시키기 위한 방안은 정보 과학에서 찾을 수 있다. 정보 과학은 프로그램의 단순 기능 습득에서 벗어나 고차적인 사고력을 요구하는 내용을 담고 있다. 정보 과학에서 논리적인 사고 및 반성적 사고를 향상시킬 수 있는 분야는 프로그래밍 교육 분야가 대표적이다[3][8][13].

프로그래밍 교육은 학습자의 논리적인 사고력, 반성적 사고력, 창의력, 문제해결력에 많은 영향을 준다. 최근 CPS에 기반한 스크래치 EPL이 문제해결력과 프로그래밍 태도에 영향을 줌을 시사하고 있다[10].

스캐폴딩에 기반한 스크래치 프로그래밍 학습은 알고리즘적 사고력에 유의미한 영향을 주지만[7] 구체적 조작기에 해당하는 초등학생에게 프로그래밍 과정은 매우 어렵게 느껴진다. 그러나 교사와 동료간 도움을 통해 자신감, 친근감 등 학습태도에 영향을 줄 수 있다.

교사와 동료 학습자간 도움을 바탕으로 하는 스크래치 프로그래밍 학습을 통해 학습자가 프로그래밍 과목에 대한 필요성, 친근감, 자신감 등 학습태도의 영향에 관한 연구는 아직 미흡한 편이다. 따라서 본 연구에서는 스캐폴딩에 기반한 학습 전략이 스크래치 프로그래밍에 대한 학습 태도를 향상할 수 있는가를 알아보려고 하였다.

1.2 선행연구

이은경(2009)은 프로그래밍 도구의 선택과 학습효과를 최대화하기 위한 교수학습 도구와 학습전략으로 과제중심 스크래치 프로그래밍 학습을 개발하고 문제해결력에 미치는 영향을 분석하였다. 그러나 문제해결력에 미치는 영향이 과제중심의 학습전략에 의한 것인지 혹은 프로그래밍 학습도구의 특성에 따른 것인지 불분명하다[8].

조성환(2007)은 프로그래밍 교육은 학생들의 창의력, 문제해결력, 논리적 사고력 향상 등에 긍정적인 영향을 끼치는 영향을 분석하였다. 특히 학습자에게 프로그래밍에 대한 부정적인 인식이 사용 방법에 있음을 시사하고 있다. 그러나 스크래치는 학습자의 문제해결력 향상과 태도에 유의미한 영향을 줌을 단일 집단 사전사후 검사를 통해 검증하였다[10]. 조성환(2008)은 설계된 학습 내용을 초등학교 6학년 재량활동 시간에 적용한 결과, 스크래치 프로그래밍 학습은 학습자의 내재적 동기와 문제해결력 향상에 효과가 있는 것으로 나타났다. 상이한 프로그래밍 도구의 사용을 통한 보편적인 프로그래밍 학습방법으로 지도한 부분이 한계를 지닌다[11].

본 연구에서는 프로그래밍 학습에 끼치는 다양한 요인 중 프로그래밍 도구의 특성에 의해 학습 태도면이 가장 효과적인 결과로 나타날 것이라 판단하였다. 그래서 단일집단의 검사를 통해 검증하기보다는 두 집단의 사전 사후검사를 통해 스크래치 프로그래밍 학습이 일반적인 프로그래밍 학습보다 학습태도에 유의미한 영향을 끼치는지 확인하고자 하였다. 스크래치 프로그래밍 도구의 특성을 파악하고, 특성에 따른 교과내용을 조직하여 교육현장에 투입하였다[2].

2. 이론적 배경

2.1 스캐폴딩

스캐폴딩의 사전적 의미는 ‘건물을 건축하거나 수리할 때 인부들이 건축재료를 운반하며 오르내릴 수 있도록 건물 주변에 세우는 장대와 두꺼운 판자로 된 발판을 세우는 것’을 의미한다[9]. 이를 교수-학습에 연결시켜 보면 학습에서 성인들이나 뛰어난 동료의 도움을 줄 때 아동의 요구에 맞게 민감하게 조절되는 문제 해결을 지원하는 하나의 체계로서 이 연구에서는 교사의 언어적 도움과 힌트의 활용 체계를 말한다.

스캐폴딩에 대한 Hannafin 등의 기능적 분류는 <표 1>와 같다[1]. 즉, 절차적 스캐폴딩은 학습 환경 내에서 제공하는 자원과 도구 사용과 관련된 도움이며, 전략적 스캐폴딩은 학습과제 수행 초기에 필요한 전략 사용과 관련된 도움을 말한다. 또한 개념적 스캐폴딩은 학습 과제와 관련된 주요 개념의 이해를 도와주기 위한 것이며, 초인지적 스캐폴딩은 문제 해결 과정을 점검하거나 조절하는 것과 관련된 도움으로 볼 수 있다.

<표 1> 스캐폴딩의 유형

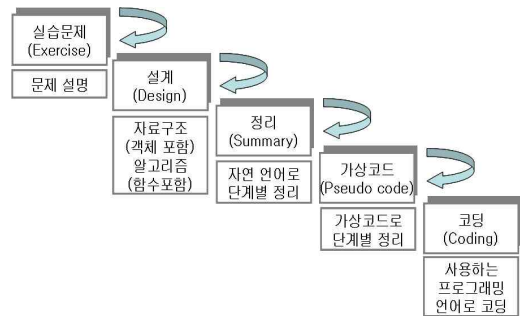
유형	특성	구현방법
절차적 스캐폴딩	시스템 기능과 특징 제시	시스템 사용법에 대한 도움말 제공
전략적 스캐폴딩	과제 또는 문제를 분석하고 접근하는 것을 안내	학습을 시작할 때 과제 해결에 필요한 전략을 생각해 보게 하는 질문 또는 전문가의 조언 제공
개념적 스캐폴딩	핵심 개념에 대한 안내	주요 개념에 대하여 동료들이 서로 공유할 수 있게 개념 설명 장을 제공
초인지적 스캐폴딩	학습 과정 모니터링 및 조절 지원	인지 과정을 외현화하여 자신과 동료의 인지과정을 모니터링 또는 조절하도록 지원해 주는 노트 기능 제공

2.2 프로그래밍 학습 과정

학습자가 실생활의 문제를 표현하고 아주 초보적

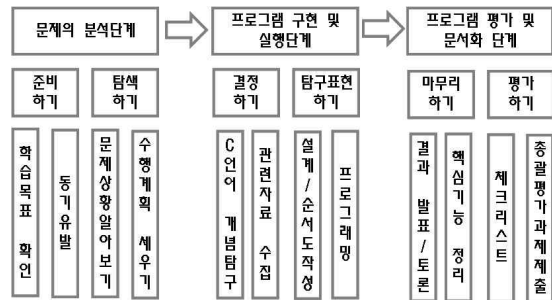
인 문제의 표현은 간단하지만 비교적 복잡한 문제를 표현하기 위해 함수 혹은 객체를 빌딩 블록으로 사용해서 보다 복잡한 문제를 풀어가는 과정을 익히는 것이 필요하다[5].

따라서 문제 해결을 위한 체계적인 프로그래밍 절차를 철저히 준수하는 것이 필요하다. 김홍환은 프로그래밍 언어에 대한 오랜 기간 동안의 교육경험으로 비추어 볼 때에 효과적인 프로그래밍 언어의 교육 방법이 있다고 주장하고 이를 체계적으로 정리하여 사이버 교육 시스템에 접목을 시켜 (그림 1)과 같은 프로그래밍 실습 과정을 제안하였다[5].



(그림 1) 프로그래밍 실습과정의 체계화[5]

또한 로봇 프로그래밍에서 활용되는 일반적인 프로그래밍의 절차는 (그림 2)와 같다[14].



(그림 2) 로봇 프로그래밍 학습과정[14]

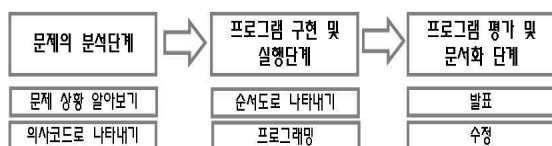
먼저, 문제의 분석 단계로써 주어진 문제의 내용을 파악하고 분석하여 프로그래밍에 대한 구체적인 범위와 절차를 결정한다. 그리고 설계 및 순서도 작성 단계로써 어떠한 형태를 입력하고, 입력한 자료에

대한 실행결과를 어떻게 출력할 것인가에 관한 자료들을 설계한다. 그 다음 문제 해결을 위한 순서와 방법을 상세하게 결정하여 순서도로 나타낸다.

둘째, 프로그램 구현 및 실행단계로써 프로그래밍 언어를 일정한 양식의 용지에 작성한다. 작성된 프로그램을 입력한 후 오류를 검색하여 프로그래밍 언어의 문법과 규칙에 맞지 않으면 오류내용을 수정한 다음 원시 프로그램의 오류가 없을 때까지 이 과정을 반복한다.

셋째, 프로그램의 평가 및 문서화 단계로써 과정 중의 모든 자료와 실행 결과를 분석하고 평가해서 체계적으로 정리한다.

이와 같은 프로그래밍 학습 과정을 종합하면 (그림 3)과 같이 정의할 수 있다.



(그림 3) 일반적 프로그래밍 학습 과정

2.3 학습 태도

학습 태도는 학습자의 학습에 대한 평가적 신념과 감정에 관한 것이다. 학습자의 학업 성취이자 참여자의 활동적, 지속적 행동에 영향을 끼치는 정의적 영역에 속한다. 학습태도에 대한 여러 학자들이 견해를 살펴보면 학습 태도를 대상에 대한 개인의 신념이나 의견을 기초로 가지게 되는 대상에 대한 평가적인 반응이라고 할 수 있다[16].

또한 학습 태도는 학습에 대하여 나타나는 개인 내부의 심리적 특성으로 학습에 대한 호의적, 비호의적 선택으로 종합하여 보면 학습에 영향을 미치는 정의적 특성 중 대표적인 요인인 학습 태도는 경험에 의해 조직된 준비 상태이며 개인에게 관계되는 모든 상황에서 반응에 영향을 주는 마음의 준비상태라고 할 수 있다[7]. 따라서 학습 태도란 학습자가 갖고 있는 긍정적 혹은 부정적 태도로써 학습에 대해 나타나는 일관성 있는 경향이라고 할 수 있다.

3. 교육과정의 개발

3.1 교육내용의 구성방향

본 연구에서는 스캐폴딩 기반 학습 전략이 스크래치 프로그래밍 학습 태도에 어떻게 영향을 미치는가를 알아보고자 하였다. 따라서 실험집단의 경우에 스캐폴딩에 기반한 스크래치 프로그래밍 학습을 하게 한 반면, 통제집단의 경우에는 일반적 스크래치 프로그래밍을 학습하도록 하였다.

초등학생을 위한 스캐폴딩에 기반한 스크래치 프로그래밍 교육 내용의 구성 방향은 다음과 같다[6].

첫째, 교육내용의 적용대상이 초등학생이다. Piaget의 인지발달 이론에 따르면 초등학생은 구체적 조작기 단계에 해당한다. 구체적 조작기 단계의 특징은 유목화, 서열화, 가역적 사고를 할 수 있도록 하여야 한다.

둘째, 알고리즘 교육내용은 개정된 컴퓨터교과의 다양한 교과서 및 교재에 의거하여 교수학습 활동에 손쉽게 이용할 수 있도록 구체적이고 심도 있게 제공하여야 한다.

3.2 단원의 구성 및 학습내용

본 연구에서는 알고리즘의 교육 내용을 총 3단원, 12차시 분량으로 구성되는데 크게 ‘프로그래밍의 기초’, ‘프로그램의 분석’, ‘내가 만드는 프로그램’으로 구성하였다. 이 중에서 ‘프로그래밍의 기초’ 대단원 부분은 알고리즘과 의사코드, 순서도, 스크래치 프로그래밍의 이해 등으로 <표 2>와 같다.

<표 2> ‘프로그래밍의 기초’ 대단원의 구성

대단원	소단원	차시	학습내용
1. 프로그래밍의 기초	알고리즘과 의사코드	1	▶ 알고리즘의 의미 ▶ 알고리즘의 필요성 ▶ 생활 속 알고리즘
		2	▶ 의사코드의 의미 ▶ 프로그래밍의 의미와 필요성
	순서도	3	▶ 순서도의 의미, 기호 ▶ 순서도의 필요성 ▶ 편리한 점
	스크래치 프로그래밍의 이해	4	▶ 스크래치의 의미, 기능 ▶ 스크래치의 활용

‘프로그램의 분석’ 대단원에서는 <표 3>의 내용과 같이 스캐폴딩의 상황 속에서 효과적인 프로그램을 할 수 있도록 스크래치의 기본 프로그램 및 다양한 수준별 게임 프로그램, 검색과 정렬 알고리즘을 교재로 직접 제공하여 분석을 통해 학습태도를 향상시키고자 하였다.

<표 3> ‘프로그램의 분석’ 대단원의 구성

대단원	소단원	차시	학습내용
II. 프로그램의 분석	기본 프로그램을 이해하자!	5	▶ 선형 프로그램 ▶ 선택형 프로그램 ▶ 순환형 프로그램 ▶ 혼합형 프로그램
	게임 프로그램을 이해하자!	6	▶ 수준별 선택하기 - 춤추는 댄서 - 키보드로 움직이는 우주선 - 페인터 - 닥터게임
	검색알고리즘	7	▶ 선형검색 알고리즘 ▶ 이진검색 알고리즘 ▶ 비교횟수 세기
	정렬알고리즘		▶ 선택정렬 ▶ 퀵정렬 ▶ 수행횟수세기
	효과적 프로그램 (시간을 줄여라.)		▶ 효과적 프로그램의 정의 ▶ 비교횟수의 필요성 ▶ 효과적 프로그램의 기준 (시간, 공간)

‘내가 만드는 프로그램’ 대단원에서는 <표 4>의 내용과 같이 분석과정을 통해서 직접 프로그래밍을 할 수 있도록 하였다. 먼저 ‘프로그래밍 과정 이해하기’ 소단원에서는 프로그래밍의 과정을 자세히 이해하는 활동을 하며, 이를 토대로 ‘프로그램을 설계하자!’ 소단원에서는 자신이 개발할 프로그램을 직접 설계하기 위해 프로그램의 유형을 파악하고, 문제를 이해하며 이를 통해 의사코드와 순서도를 그려봄으로써 프로그래밍의 과정을 손쉽게 이해할 수 있으며 설계가 가능하도록 한다. ‘나도 이제 프로그래머’ 소단원에서는 설계된 프로그램을 스크래치 언어로 코딩하고 보다 효과적인 프로그램을 구현하는 방법에 대하여 설명한다.

<표 4> ‘내가 만드는 프로그램’ 대단원의 구성

대단원	소단원	차시	학습내용
III. 내가 만드는 프로그램	프로그래밍 과정 이해하기	8	▶ 문제를 파악하기 ▶ 의사코드 작성하기 ▶ 순서도 그리기 ▶ 코딩하기
	프로그램을 설계하자!	9	▶ 프로그램 유형 파악 ▶ 문제를 이해하기 ▶ 의사코드로 나타내기 ▶ 순서도로 나타내기
	나도 이제는 프로그래머!	10	▶ 스크래치 프로그래밍하기
		11	▶ 수행횟수를 세기
		12	▶ 효과적인 프로그램 만들기

3.3 학습내용 체계에 따른 스캐폴딩 전략

프로그래밍 과정에서 다른 사람이 작성한 프로그램을 분석하는 것이 먼저 선행되면 창조적인 프로그래밍을 하는데 많은 도움을 줄 수 있다. 그러나 초등학생의 경우 프로그램을 스스로 분석하는데 한계가 있다. 그러므로 학습자가 이해하기 쉬운 스크래치 프로그래밍을 통해 수준별 학생의 프로그래밍 능력의 차이를 극복할 수 있도록 스캐폴딩 전략을 구성하였다. 스캐폴딩에 대한 Hannafin 등의 기능적 분류에 따라 스캐폴딩의 유형별로 절차적, 전략적, 개념적, 초인지적 스캐폴딩으로 구현하는 방법은 <표 5>와 같다.

<표 5> ‘프로그램의 분석’의 스캐폴딩 구현 방법

학습 내용	스캐폴딩 유형	구현방법
기본 프로그램	절차적 스캐폴딩	순서도를 활용하여 프로그램의 절차적 과정을 이해함으로써 스크래치의 기본적인 기능과의 연관을 생각하게 한다.
수준별 프로그램	전략적 스캐폴딩	다양한 프로그램의 예시를 직접 구현해봄으로써 다양한 방법을 터득하도록 하며, 이를 통해 다양한 전략을 세워 새로운 프로그램을 구상할 수 있는 기회를 제공하게 한다.
효과적 프로그램	초인지 스캐폴딩	검색 알고리즘과 정렬 알고리즘을 예제로 효과적인 프로그램을 시간복잡도를 줄일 수 있는 방안을 생각하도록 한다.

<표 5>에 나타난 학습 내용 중에서 기본 프로그램에 대한 절차적 스캐폴딩의 내용과 스크래치 프로그램의 예는 <표 6>과 같다.

<표 6> 기본 프로그램의 기능 이해를 위한 절차적 스캐폴딩

종류	스크래치 프로그램 예	절차적 스캐폴딩
선형		선형 프로그램의 경우, 가장 기초가 되는 내용으로 선형 프로그램의 기능의 이해와 특징을 파악하도록 도와준다.
선택형		선택형 프로그램의 경우, 선택의 기능을 많이 사용하는 사례를 통해서 기능을 이해하고 특징을 파악하도록 도와준다.
순환형		순환형 프로그램의 경우, 조건에 맞으면 계속 순환함을, 아니면 끝으로 가는 기능을 이해하고 특징을 파악한다.
혼합형		혼합형 프로그램의 경우, 순환형과 선형, 선택형 프로그램의 특징과 기능을 전체적으로 이해할 수 있도록 절차적 도움을 주는데 의의가 있다.

<표 5>에 나타난 학습 내용 중에서 수준별 프로그램에 대한 전략적 스캐폴딩의 내용과 스크래치 프로그램의 예는 <표 7>과 같다.

<표 7> 기본 프로그램의 기능 이해를 위한 전략적 스캐폴딩

종류	스크래치 프로그램 예	전략적 스캐폴딩
춤추는 댄서		'춤추는 댄서' 프로그램의 경우, 순환형과 선형 프로그램이 혼합된 프로그램으로 모양의 변화를 통해서 마치 춤을 추는 듯한 모습을 연출하기 위한 프로그래밍을 이해하는데 도움을 준다.
키보드로 움직이는 우주선		선택형 프로그램의 경우, 선택의 기능을 많이 사용하는 사례를 통해서 기능을 이해하고 특징을 파악하도록 도와준다.
페인터		'페인터' 프로그램의 경우, 순환형, 선택형, 선형 프로그램이 모두 포함된 형태로 변수의 사용에 따른 기능을 이해할 수 있으며, 자신이 만들 프로그램을 전략적으로 생각할 수 있도록 도와 준다.
다트		다트 프로그램의 경우, 순환형과 선형 프로그램을 포함하고 있으며, 두 스프라이트를 프로그래밍을 함으로써 서로의 연관성을 파악할 수 있다. 이를 통해 자신의 프로그램을 전략적으로 구상을 할 수 있도록 도와준다.

<표 5>에 나타난 학습 내용 중에서 효과적 프로그램에 대한 초인지 스캐폴딩의 내용과 스크래치 프로그램의 예는 <표 8>과 같다.

<표 8> 정렬, 검색 프로그램을 통한 초인지 스캐폴딩

종류	자료 제시 예	초인지 스캐폴딩																																																		
순차 검색	<table border="1"> <tr><td>1</td><td>3</td><td>5</td><td>7</td><td>13</td></tr> <tr><td>1</td><td>3</td><td>5</td><td>7</td><td>13</td><td>17</td><td>20</td><td>31</td><td>35</td><td>40</td></tr> </table>	1	3	5	7	13	1	3	5	7	13	17	20	31	35	40	순차검색과 이진검색을 통해 얻어진 수행횟수를 통해 비교를 하게 되며 프로그램의 효과성을 판단할 수 있게 된다.																																			
1	3	5	7	13																																																
1	3	5	7	13	17	20	31	35	40																																											
이진 검색	<table border="1"> <tr><td>1</td><td>3</td><td>5</td><td>7</td><td>13</td></tr> <tr><td>1</td><td>3</td><td>5</td><td>7</td><td>13</td><td>17</td><td>20</td><td>31</td><td>35</td><td>40</td></tr> </table>	1	3	5	7	13	1	3	5	7	13	17	20	31	35	40																																				
1	3	5	7	13																																																
1	3	5	7	13	17	20	31	35	40																																											
선택 정렬	<table border="1"> <tr><td>7</td><td>5</td><td>1</td><td>9</td><td>33</td><td>17</td><td>13</td><td>88</td><td>45</td><td>67</td></tr> <tr><td>1</td><td>7</td><td>5</td><td>9</td><td>33</td><td>17</td><td>13</td><td>88</td><td>45</td><td>67</td></tr> <tr><td>1</td><td>5</td><td>7</td><td>9</td><td>33</td><td>17</td><td>13</td><td>88</td><td>45</td><td>67</td></tr> <tr><td>1</td><td>5</td><td>7</td><td>9</td><td>18</td><td>17</td><td>13</td><td>88</td><td>45</td><td>67</td></tr> <tr><td>1</td><td>5</td><td>7</td><td>9</td><td>18</td><td>17</td><td>33</td><td>45</td><td>67</td><td>88</td></tr> </table>	7	5	1	9	33	17	13	88	45	67	1	7	5	9	33	17	13	88	45	67	1	5	7	9	33	17	13	88	45	67	1	5	7	9	18	17	13	88	45	67	1	5	7	9	18	17	33	45	67	88	선택정렬과 퀵정렬의 수행 횟수를 비교하여 최적의 정렬 방법이 무엇이 있는지 확인을 하고, 프로그래밍에서 같지만 효과적인 방법이 무엇이 있는가를 확인하도록 한다. 첫 번째부터 키가 되어 반대의 위치에 있는 키들을 비교하여 첫 번째 키보다 작으면 왼쪽으로 크면 오른쪽으로 이동하게 된다.
	7	5	1	9	33	17	13	88	45	67																																										
1	7	5	9	33	17	13	88	45	67																																											
1	5	7	9	33	17	13	88	45	67																																											
1	5	7	9	18	17	13	88	45	67																																											
1	5	7	9	18	17	33	45	67	88																																											
퀵 정렬	<table border="1"> <tr><td>7</td><td>5</td><td>1</td><td>9</td><td>33</td><td>17</td><td>13</td><td>88</td><td>45</td><td>67</td></tr> <tr><td>1</td><td>5</td><td>7</td><td>9</td><td>33</td><td>17</td><td>13</td><td>88</td><td>45</td><td>67</td></tr> <tr><td>1</td><td>5</td><td>7</td><td>9</td><td>33</td><td>17</td><td>13</td><td>88</td><td>45</td><td>67</td></tr> <tr><td>1</td><td>5</td><td>7</td><td>9</td><td>18</td><td>17</td><td>13</td><td>88</td><td>45</td><td>67</td></tr> <tr><td>1</td><td>5</td><td>7</td><td>9</td><td>18</td><td>17</td><td>33</td><td>45</td><td>67</td><td>88</td></tr> </table>	7	5	1	9	33	17	13	88	45	67	1	5	7	9	33	17	13	88	45	67	1	5	7	9	33	17	13	88	45	67	1	5	7	9	18	17	13	88	45	67	1	5	7	9	18	17	33	45	67	88	
7	5	1	9	33	17	13	88	45	67																																											
1	5	7	9	33	17	13	88	45	67																																											
1	5	7	9	33	17	13	88	45	67																																											
1	5	7	9	18	17	13	88	45	67																																											
1	5	7	9	18	17	33	45	67	88																																											

‘프로그래밍의 기초’와 ‘프로그램의 분석’ 대단원의 내용을 학습하게 되면 프로그램을 작성할 수 있는 단계에 이르게 되며 ‘내가 만드는 프로그램’ 대단원을 통하여 학생들이 스크래치 언어를 활용하여 직접 프로그램을 작성할 수 있다. 이때 절차적, 전략적 및 초인지 스캐폴딩의 구현 방법은 <표 9>와 같다.

<표 9> ‘내가 만드는 프로그램’ 대단원의 스캐폴딩 구현 방법

학습내용	스캐폴딩	구현방법
프로그래밍 과정 이해하기	절차적 스캐폴딩	프로그래밍의 과정을 절차적으로 배움으로써 기본적인 방법을 익히는데 의의가 있다. 먼저 의사코드를 작성하고, 의사코드를 그대로 순서대로 나타내어 스크래치 프로그램을 하는 과정을 배운다.
프로그램을 설계하자!	전략적 스캐폴딩	학생이 자신이 생각한 프로그램을 작성하는데 있어서 과제 해결에 필요한 전략을 생각해 보게 하는 질문 또는 전문가의 조언을 제공한다.
나도 이제 프로그래머!	초인지 스캐폴딩	학생이 작성한 프로그램을 반성적 사고를 통하여 모니터링을 함으로써 최적의 프로그램으로 나아가도록 조언을 제공한다.

<표 9>에 나타난 학습 내용 중에서 ‘프로그래밍 과정 이해하기’ 소단원에 대한 스캐폴딩의 구현 방법은 <그림 10>과 같다.

<표 10> ‘프로그래밍 과정 이해하기’ 소단원의 절차적 스캐폴딩

과정	학생 활동	절차적 스캐폴딩
문제를 파악하기	다양한 실생활에서 프로그래밍을 할 내용을 탐색해 본다.	프로그래밍이 순서적으로 어떻게 이루어지는지를 절차적으로 과정을 익히게 된다.
의사코드 작성하기	실생활의 문제를 파악하고 의사코드로 나타낸다.	
순서도 그리기	의사코드로 나타낸 것을 순서도로 그리기	
코딩하기	순서도로 나타낸 것을 스크래치 프로그램으로 프로그래밍하기	

<표 9>에 나타난 학습 내용 중에서 ‘프로그램을 설계하자!’ 소단원에 대한 스캐폴딩의 구현 방법은 (그림 4)와 같다. 여기에서 전략적 스캐폴딩으로 교사는 학생이 작성하고자 하는 프로그램을 전략적으로 구상하고 목적인 바대로 나아가도록 도움을 주게 된다.



(그림 4) ‘프로그램을 설계하자!’ 소단원의 스캐폴딩 구현 방법

<표 9>에 나타난 학습 내용 중에서 ‘나도 이제 프로그래머!’ 소단원에 대한 스캐폴딩 구현 방법은 <표 11>과 같다. 여기에서 자신이 구현한 프로그램을 수행 횟수를 세어 시간 복잡도를 계산함으로써 얼마나 효과적으로 수행되는지 확인할 수 있다. 초인지적 스캐폴딩을 통한 교사와 동료의 도움을 받아 보다 효율적인 프로그램으로 나아가기 위한 다양한 방향을 고려하여 가장 최적의 프로그램이 되도록 한다.

<표 11> ‘나도 이제는 프로그래머!’ 소단원의 특성에 따른 스캐폴딩 구현 방법

과정	학생 활동	초인지적 스캐폴딩
반성을 통한 순서도 수정	<ul style="list-style-type: none"> 동료 학생들이 순서도를 돌려 보고 자신의 생각을 말해 보도록 하고 더 좋은 방법이 있는지 방향을 생각해 본다. 	인지 과정을 외 화 하여 자신과 동료의 인지과정을 모니터링 또는 조절하도록 지원해 준다.
반성을 통한 코딩 수정	<ul style="list-style-type: none"> 자신의 프로그램을 발표함으로써 모든 학생들에게 공개를 하고, 더 좋은 방법이 있는지 생각해 보도록 한다. 	

4. 연구 방법 및 결과

본 연구에서 사용되는 학습태도 평가지의 구성 영역은 김은형[4]과 정경호[12] 등의 선행연구를 중심으로 조사 항목을 추출한 다음, 이 연구의 목적과 내용에 알맞게 재구성하였다.

4.1 학습태도 평가지의 개발

본 연구에서 사용되는 학습 태도 평가 항목을 학습자의 일반적인 학습 경향성, 학습을 통한 성과에 대한 만족도, 교사와 학습자간 혹은 학습자간 상호작용 등으로 분류하였는데 그에 따른 문항은 <표 12>부터 <표 14>까지와 같다.

<표 12> 학습자의 일반 학습 경향성

분류	문항
감정	나는 프로그래밍 공부 시간이 즐겁다.
	나는 프로그래밍 공부 시간이 지루하다.
집중력	나는 프로그래밍 시간에 다른 생각을 많이 한다.
	나는 프로그래밍 시간에 바른 자세로 열심히 수업에 집중한다.
재학습성	나는 프로그래밍 시간에 다른 학생과 장난을 하지 않는다.
	나는 프로그래밍 과목은 꼭 예습을 한다.
	나는 프로그래밍 시간에 배운 것을 꼭 복습을 한다.
자기주도적 학습력	나는 프로그래밍 시간에 배운 것을 확실히 알고 넘어간다.
	나는 누가 시키지 않아도 스스로 프로그래밍 공부를 한다.
자신감	나는 프로그래밍 공부를 잘 할 수 있다.

<표 13> 학습을 통한 성과에 대한 만족도

분류	문항
난이도	나는 프로그래밍 공부가 쉽다.
	나는 프로그래밍 시간이 끝났을 때 무엇을 배웠는지 잘 모른다.
의욕	나는 프로그래밍에 대해 더 많이 배우고 싶다.
	나는 프로그래밍 시간에 배운 것을 응용해 보고 싶다.
자신감	나는 프로그래밍 공부를 많이 하고 싶다.
	나는 프로그래밍에 소질이 있는 것 같다.
	나는 프로그래밍만큼은 잘 할 수 있다.
	나는 프로그래밍 시험을 본 후 점수를 빨리 알고 싶다.
	나도 이만하면 프로그래밍을 잘 하는 학생이라고 생각한다.
학습의 중요성	나는 프로그래밍 시험에서 좋은 점수를 얻는다.
	나는 프로그래밍이 앞으로 공부하는데 꼭 필요한 수업이라고 생각한다.

<표 14> 교사와 학습자간 혹은 자신과의 상호작용

분류	문항
교사와의 상호작용	나는 프로그래밍을 잘해서 칭찬을 받은 적이 있다.
	나는 프로그래밍 시간에 발표하는 것을 좋아한다.
자신과의 상호작용	나는 프로그래밍 시간이 끝난 후 그 시간에 배운 것을 머릿속에 정리해 본다.

4.2 연구 결과

연구 대상 학생을 실험집단과 비교집단으로 구성하였고, 두 집단에 대해 사전검사를 실시하여 동질집단 여부를 살펴보았다. 두 집단의 알고리즘적 사고력의 사전검사의 결과는 <표15>와 같다.

<표15> 사전 검사에 대한 t-검증 결과

구분	사례수	평균	표준편차	t	p(sig.)
실험집단	34명	83.05	7.08	-.345	.732
비교집단	34명	82.26	11.42		

실험집단과 비교집단 간의 차이를 살펴 본 결과, p-value의 값이 .05보다 크기 때문에 동일한 집단임을 알 수 있다.

스캐폴딩에 기반한 스크래치 프로그래밍을 적용한

실험집단과 일반적 스크래치 프로그래밍을 적용한 비교 집단 내에서의 학습태도에 대한 사전·사후 검사 결과는 <표 16>과 같다.

<표 16> 학습 태도에 대한 사전·사후 검사 차이 분석

구분	사례수	시기	평균	표준편차	t	p(sig.)
실험집단	34명	사전검사	83.74	9.12	-2.227	.029
		사후검사	88.18	7.21		
비교집단	34명	사전검사	85.09	9.06	-.453	.652
		사후검사	85.94	6.85		

<표 16>에서 보면 실험집단이 사전·사후 평균치가 4.44점, 비교집단이 0.85점으로 나타났으며 p-value의 값이 실험집단은 .029로 .05보다 작으므로 스크래치 프로그래밍 수업이 학습태도에서 긍정적인 효과를 얻었음을 알 수 있다(p<.05).

또한 프로그래밍에 대한 학습태도를 일반 학습 경향성, 학습을 통한 성과의 만족도, 상호작용 등 3가지 항목으로 나누어 분석해 보면 다음과 같다.

첫째, 학습자의 일반 학습 경향성의 사전·사후 검사 결과는 <표 17>과 같다.

<표 17> 학습자의 일반 학습 경향성에 대한 사전·사후 검사 차이 분석

구분	사례수	시기	평균	표준편차	t	p(sig.)
실험집단	34명	사전검사	85.74	7.60	-2.088	.041
		사후검사	89.38	6.78		
비교집단	34명	사전검사	86.76	8.52	.400	.690
		사후검사	86.95	6.49		

<표 17>에서 보면 p-value의 값이 실험집단은 .041로 실험집단이 비교집단에 비해 일반 학습 경향성 면에서 긍정적인 효과를 얻었음을 알 수 있다(p<.05).

둘째, 학습을 통한 성과에 대한 만족도는 실험집단과 비교 집단 내에서의 사전·사후 검사 결과는 <표 18>과 같다.

<표 18> 학습을 통한 성과의 만족도에 대한 사전·사후 검사 차이 분석

구분	사례수	시기	평균	표준편차	t	p(sig.)
실험집단	34명	사전검사	82.91	11.20	-2.905	.005
		사후검사	87.65	7.56		
비교집단	34명	사전검사	84.47	8.76	.628	.532
		사후검사	84.26	6.98		

<표 18>에서 보면 p-value의 값이 실험집단은 .005로 스크래치 프로그래밍 수업이 학습을 통한 성과의 만족도 면에서 긍정적인 효과를 얻었음을 알 수 있다(p<.05).

셋째, 교사와 학습자간 혹은 학습자간 상호작용은 실험집단과 비교 집단을 비교한 사전·사후 검사 결과는 <표 19>와 같다.

<표 19> 교사와 학습자간 혹은 학습자간 상호작용에 대한 사전·사후 검사 차이 분석

구분	사례수	시기	평균	표준편차	t	p(sig.)
실험집단	34명	사전검사	81.23	12.91	-4.505	.000
		사후검사	92.24	6.01		
비교집단	34명	사전검사	73.41	9.01	-2.316	.024
		사후검사	87.65	5.71		

<표 19>에서 보면 p-value의 값이 실험집단과 비교집단 모두 .000, .024로 .05보다 작으므로 실험집단이 효과적임을 알 수 있다(p<.05).

5. 결론 및 제언

스크래치 프로그래밍은 드래그 앤 드롭 형식의 프로그래밍이 가능하며, 이는 구체적 조작기인 학습자가 복잡한 문제 상황에서 체계적으로 문제를 해결하기 위한 과정을 보다 쉽게 이해할 수 있도록 한다. 스크래치 프로그래밍은 학습자에게 호기심을 자극하여 학습에 대한 동기와 집중력을 높여 효과적으로 학습이 되도록 함을 많은 연구 결과에서 보여주고 있다. 그러나 프로그래밍 작업은 여전히 초등학생들에게 힘든 작업이다. 그러므로 이러한 인식을 불식시

키고 도전해 볼만한 과제임을 알게 하는 것이 중요하다.

스캐폴딩에 기반한 학습 전략은 이러한 초등학생의 어려움을 해소시켜 줄 수 있는 방안 중의 하나이다. 따라서 본 연구에서는 스캐폴딩에 기반한 학습 전략이 스크래치 프로그래밍 학습 태도에 어떤 영향이 있는가를 알아보았다.

연구 결과로 교사와 학습자간 혹은 학습자간의 상호작용 면에서 효과가 있는 것으로 나타났다. 그러므로 스크래치 프로그래밍 학습에 있어서 스캐폴딩에 기반한 학습 전략의 적용은 학생들의 학습태도에 긍정적 영향을 끼치고 있음을 알 수 있다.

이에 본 연구는 다음과 같은 의의를 지닌다.

첫째, 스크래치 프로그래밍 교육이 교사와 학습자간의 학습도움을 통해 학습 태도 향상에 유의미함을 알 수 있다.

둘째, 스캐폴딩에 기반한 학습 전략의 적용이 스크래치 프로그래밍 학습에서 학생들의 학습 태도에 긍정적인 영향을 끼친다.

셋째, 스캐폴딩에 기반한 학습 전략은 프로그래밍을 어려워하는 초등학생들을 위한 학습 전략 중의 하나가 될 수 있다.

이와 더불어 향후에는 본 연구의 결과를 토대로 초등학생 수준에 맞는 스크래치 프로그래밍 언어의 교육과정의 개발 및 교수방법의 탐구가 이루어져야 할 것이다.

참고문헌

[1] 김근희(2008). 웹기반 학습 환경에서 스캐폴딩 유형에 따른 학업성취도와 학습만족도의 차이. 석사학위논문, 숙명여자대학교.

[2] 김승연(2010). 스캐폴딩에 기반한 스크래치 프로그래밍 학습이 알고리즘적 사고력에 미치는 영향. 석사학위 논문, 춘천교육대학교.

[3] 김은진(2009). 웹퀘스트를 활용한 스크래치 프로그래밍 학습에 관한 연구. 석사학위 논문, 서울교육대학교.

[4] 김은형(2008). 초등학생의 수학 학습 태도를 형성하는 요인에 대한 연구. 석사학위 논문, 서울교육

대학교.

[5] 김홍환(2003). 프로그래밍 언어의 효율적인 교육을 위한 사이버 교육시스템에 대한 연구. 응용과학연구, 12-1.

[6] 오승아(2000). 알고리즘 지도 방향에 관한 연구. 석사학위 논문, 서울대학교.

[7] 이계수(1994). 자기평가를 통한 이해도 진단과 피드백이 학업성취 및 학습태도에 미치는 영향. 석사학위논문, 중앙대학교.

[8] 이은경, 이영준(2008). Scratch 활용 프로그래밍 교육이 중학생의 몰입수준과 프로그래밍 능력에 미치는 영향. 중등교육연구.

[9] 임선희(2009). 스캐폴딩 개념을 적용한 운영체제 교과의 웹 코스웨어 설계 및 구현 : 가상기억장치 중심으로. 석사학위 논문, 홍익대학교.

[10] 조성환, 송정범, 김성식, 이경화(2007). CPS에 기반한 스크래치 EPL이 문제해결력과 프로그래밍 태도에 미치는 효과. 한국교원대학교.

[11] 조성환, 송정범, 이태욱(2008). 스크래치 프로그래밍 학습이 학습자의 동기과 문제해결력에 미치는 영향. 정보교육학회논문지.

[12] 정경호(2005). 웹 기반 교육의 학습 성과에 관한 연구 : 학습자 태도와 상호 작용 요인을 중심으로. 석사학위논문, 전남대학교.

[13] 정은숙(2009). 중학교 프로그래밍 수업에서 순서도 학습이 논리적 사고력과 성취도에 미치는 영향. 석사학위논문, 한국교원대학교.

[14] 채재호, 배영권, 유인환(2008). 로봇프로그래밍 학습과정. Korean Journal of Teacher Education.

[15] Bijan b, Gillani(2007). 학습이론과 e-러닝환경 설계. 서울 : 교육과학사.

[16] Howard Gardner(1985). *The attitude / motivation test battery: Technical Report. University of Western Ontario.*

저 자 소 개

김 승 연



2007년 : 춘천교육대학교 졸업(학사)

2010년 : 춘천교육대학원 컴퓨터교육과 졸업(교육학 석사)

2007년 - 현재 : 원주일산초등학교 교사

관심분야 : 스크래치 프로그래밍 학습, 웹 2.0 학습

e-mail : comsykim@naver.com

정 인 기



1988년 : 고려대학교 전산학과(학사)

1990년 : 고려대학교 전산학 전공(석사)

1996년 : 고려대학교 전산학 전공(박사)

1997년 - 현재 : 춘천교육대학교 컴퓨터교육과 교수

관심분야 : 데이터베이스, 자료구조 및 알고리즘

e-mail : inkey@cnue.ac.kr