

Scratch를 활용한 프로그래밍 교육이 문제해결력 향상에 미치는 영향

김현정
진주교육대학교 컴퓨터교육과
pigia@hanmail.net

The Effect of Programming Education Using Scratch on Improving Problem Solving Ability

Hyun-Jeong Kim
Dept. of Computer Education, Jinju National University of Education

요 약

본 연구는 초등학교 4학년 학생들을 대상으로 창의적이고 자기주도적인 문제해결력 향상에 도움이 되는 프로그래밍 수업을 재량시간을 통해 진행하려 한다. 프로그래밍 수업은 전문 용어, 결과가 눈에 쉽게 보이지 않는 이유 등으로 초등학생에게는 지루하고 어려워서 적용하기에는 부담스러운 면이 있다. 이에 본 연구는 프로그래밍을 아동인지단계에 맞추어 흥미를 느끼고 쉽게 접근할 수 있는 Scratch 프로그램을 활용하여 프로그래밍 수업을 하고자 한다. 단일 집단에게 수업을 실시하고 사전사후 검사를 통하여 Scratch 프로그램을 활용한 프로그래밍 수업이 문제해결력 향상에 도움이 되는지를 검증해 보고자 한다.

1. 서 론

21세기에 사는 우리는 넘쳐나는 정보의 홍수 속에 자신에게 필요한 정보를 잘 선별하여 창의적으로 각종 문제를 해결하고 변화에 적응할 수 있어야 한다. 변화하는 사회가 요구하는 인간상을 길러내기 위해서는 학교교육도 그에 맞추어 변화해야 한다. 그러기 위해서 획일적이고 일방적인 주입식 교육보다는 다양한 문제에 부딪쳐 스스로 문제를 해결하게끔 하는 창의적이고 자기주도적인 교육이 요구된다.

그러나 현 초등학교 컴퓨터 교육과정에는 다음과 같은 문제점들이 제기되고 있다. 정보교과가 별도로 주어져 있지 않고 재량활동시간이나 실과 시간에 속해있어 시수확보가 불안정하다. 이로 인해 체계적인 교육과정이 존재하지 않기 때문에 학년·영역 간 연계성이 결여되어 있고 내용이 중복되는 등 교육 내용이 부실하다. 그 또한 컴퓨터의 조작과 응용 소프트웨어의 사용법에 치중하고 있다[1].

따라서 컴퓨터의 조작이나 응용소프트웨어

를 활용하기 위해 지식을 주입하는 교육보다는 지식 정보화 사회에 적합한 창의적인 문제해결력을 키울 수 있는 교육이 필요하다. 이에 부합하는 것이 바로 프로그래밍 교육이다.

프로그래밍 교육이 학습자의 논리적 사고력 증진, 문제해결능력 신장, 완성을 통한 성취감과 자신감 획득, 토론을 통한 협동심과 상호작용 효과와 같은 고등인지 능력을 향상시킨다는 사실은 많은 연구에서 드러났다. 이에 프로그래밍 언어의 문법을 암기하는 부담감과 학습의 지루함을 줄이고, 학습자의 동기를 유발하고 흥미를 느끼게 하며, 실생활에서의 문제해결능력을 향상시킬 수 있는 환경의 제공을 위해 다양한 교육용프로그래밍 언어가 개발되었다[2].

그러나 무조건 프로그래밍 교육을 한다고 해서 문제해결력과 창의성을 기를 수 있는 것은 아니며 학생들에게 흥미와 관심을 불러일으키는 것도 아니다. 현재 프로그래밍 교육은 초등학생들에게는 어려운 언어와 문법으로 구성되어 있으며, 알고리즘 구현에는 적절한 학

습 시간과 많은 사고력이 요구되나 이에 소요되는 시간 배정이 어려워 완성된 프로그래밍 작성이 어렵다. 또한 개인 수준을 무시한 일관된 교수 학습 방법으로 인해 학습 중반 이후 학습자의 수업 참여도가 현저히 떨어지는 경향이 있다[3].

따라서 초등학생의 아동인지발달단계를 고려하여 학생들의 관심과 흥미를 높이고, 여러 가지 조작을 통하여 창의적으로 문제를 해결해나갈 수 있는 Scratch 프로그램을 활용한 프로그래밍 교육이 학생들에게 프로그래밍의 원리를 이해하고 아울러 자기주도적인 문제해결력의 향상에 효과가 있을지 검증해 보고자 한다.

2. 이론적 배경

2.1 프로그래밍 교육의 정의

컴퓨터 프로그래밍은 컴퓨터를 활용하여 사용자가 컴퓨터에게 자신이 무엇을 할 것인지를 알려주는 작업 지시를 작성하는 것이다. 컴퓨터가 이해할 수 있는 명령은 아주 제한적이고 원시적이지만 사용자가 컴퓨터에게 어떤 명령을 내리기 위해 메모리 위치를 일일이 확인할 수 없기 때문에 매개 역할을 해 줄 도구 즉, 프로그래밍 언어를 사용한다[4].

따라서 프로그래밍 교육은 보는 시각에 따라 그 의미가 달라진다고 볼 수 있다. 좁은 시각에서 본다면 프로그래밍 언어를 학습하는 것이고 넓은 시각에서 본다면 고등 인지 기능의 습득인 것이다. 문제를 분석하고 세분화하여 해결방안을 모색하고 실행하는 프로그래밍의 과정을 통하여 고등 인지 기능이 향상되는 것이다[5].

2.2 프로그래밍 교육의 문제점

프로그래밍 교육은 ICT 활용에 있어 논리적 사고력, 비판적 사고력, 창의적 문제해결력과 같은 고차원적인 사고를 필요로 하는 활동

임에도 그동안 공교육에서는 도외시 되어왔다.[6]

제 6차 교육과정까지는 프로그래밍 과정 부분이 있어 모든 학생들에 대하여 프로그래밍 교육을 조금이라도 해왔으나 7차 교육과정에서는 프로그래밍 교육이 실업계 고등학교에 한하여 교육될 뿐이다. 프로그래밍 교육은 컴퓨터 과학교육의 근간이 되며, 수학적 사고력, 문제해결력, 논리적 추론 등의 향상을 위하여 중요한 교과 내용이라 할 수 있다[7].

프로그래밍 교육에 있어 일반적으로 목표설정 달성에 미달하는 큰 요인을 다음과 같이 두 가지로 정리할 수 있다.

첫째, 기능 및 문법 중심으로 교수학습이 진행되기 때문에 많은 사고력이 소요되는 알고리즘 구현에 소요되는 충분한 시간 배정이 어려워 완성도 있는 프로그래밍 작성에 어려움이 있다.

둘째, 학습자의 개별수준을 무시한 교수 학습 방법으로 학습 중반 이후 학습자의 수업 참여도가 현저히 떨어지는 경향이 있다. 인지능력이 중등에 비해 비교적 떨어지는 초등학생을 위한 프로그래밍 교수 학습 환경을 만들기 위해서는 먼저, 학습자들은 자신의 수준에 맞는 능동적인 프로그래밍 활동을 하고 교수자는 모니터만 해 주어 가능한 손길이 적게 가도록 하는 가변적인 교육과정을 만들 필요가 있다[7]. 따라서 효과적인 프로그래밍 교육을 위해서는 이처럼 예상되는 문제점을 최대한 수정하려는 노력이 필요하다.

2.3 프로그래밍 교육의 의의

프로그래밍 교육은 다음과 같은 교육적 의의를 지닌다[8].

첫째, 학습자의 논리적 사고력과 문제 해결 능력과 같은 고등 인지 능력을 향상시키고, 프로그램의 완성을 통한 성취감과 자신감의 획득, 토론을 통한 협동심과 상호작용 효과를 얻을 수 있다.

둘째, 지식기반사회의 인프라 구조에서 가장

핵심 요소인 소프트웨어 개발 능력을 예비하기 위한 기초 교육과정이라는 점이다. 기존 소프트웨어 활용 중심의 교육만으로는 미래 지식기반사회에서 주도적, 생산적 역할을 수행하지 못하고 정보산업 강대국이 공급하는 독점적 소프트웨어만을 활용하게 될 것이다.

셋째, 학습자에게 다양한 교육 활동의 기회를 제공한다는 점이다. 컴퓨터의 이용 범위가 확산되고 있는 만큼 몇 가지 소프트웨어의 활용법 교육만으로 컴퓨터 교육은 부족하다. 프로그래밍 교육을 통해 광범위하게 이용되는 컴퓨터 분야를 알 수 있는 기회가 주어져야 한다.

넷째, 프로그래밍 교육은 컴퓨터 교육의 기초·기본 교육 및 내부 동작 원리를 이해하는데 많은 도움을 주며, 좀 더 나아가 컴퓨터를 보다 깊이 이해하고 창의력을 향상시킬 수 있는 계기가 된다.

다섯째, 각종 소프트웨어를 더 잘 알고, 활용할 수 있는 기초를 닦아준다. 많은 다양한 소프트웨어도 운영체제를 바탕으로 동작됨으로 프로그래밍 교육을 통해 일반적인 소프트웨어 동작을 이해할 수 있고, 새로운 소프트웨어가 등장하더라도 쉽게 활용할 수 있다.

2.4 문제해결력

Mayer(1999)는 문제해결력을 “명백하게 해결 방법을 알 수 없을 때 목적 상황 속에서 현재 상황을 유도하는 인지적 처리과정”이라고 하였다[9].

또한 임정훈(1999)은 문제해결력을 독립된 개념, 원리, 절차 혹은 아이디어에 대한 적용 능력이나 이해력을 벗어나, 어떠한 문제 상황이 주어졌을 때 문제를 확인하고 원인을 분석하며 해결안으로 도출하는 일련의 절차와 원리들을 응용할 수 있는 능력을 의미한다고 하였다[10].

무엇보다도 문제해결력은 학교 교육과정의 주요 교육목표 중의 하나이며, 체계적 분석, 효과적인 협력기술 등과 함께 21세기에서 반

드시 알아야 할 9가지 학습 기술 중의 하나이다[11].

2.5 아동인지발달단계

Piaget는 인지발달 이론을 통해 감각 운동기, 전조작기, 구체적 조작기, 형식적 조작기의 네 가지 인지발달 단계가 있다고 하였다. 네 가지 단계 중 구체적 조작기는 보통 7세에서 11세에 해당하며, 구체적인 문제에 대해 논리적 사고가 가능한 시기이다. 구체적 조작기에서는 특정 사실에 따라 사물을 분류할 수 있게 되어 사물을 위계에 따라 분류하는 것이 가능하다 형식적 조작기는 보통 12세 이후에 나타나는 단계로 추상적인 사고가 가능하다. 추상적 사고란 융통성 있는 사고, 효율적인 사고, 복잡한 추리, 가설을 세우고 체계적으로 검증하는 일, 직면한 문제 사태에서 해결 가능한 모든 방안을 종합적으로 고려해 보는 일 등과 같은 것을 말한다[12].

<표1> Piaget의 인지발달단계

단계	특성	발달촉진방법
감각 운동기	생각은 감각적 정보에 기초	환경에 반응할 수 있는 기회 제공
전 조작기	상징과 언어의 사용 증가 제한된 논리 자기중심적 견해	언어와 상징 사용
구체적 조작기	알기위해 보고, 느끼고, 만지고, 냄새 맡고, 들어보려 하는 것	조작경험
형식적 조작기	추상적 사고	문제, 가설, 확률, 상관, 비례형식으로 추상화 형식적 사고 가능

2.6 Scratch 프로그램

스크래치는 스쿼크(Squak, 일종의 프로그래밍 언어로, smalltalk-80으로 작성되었으며, 그 자체에 smalltalk-80과 기본 라이브러리를 가지고 있는 개발 툴)을 기반으로 MIT Media Lab에서 개발한 8세 이상의 어린이를 위한 개발 툴로, 로고와 같이 생각한 내용을 애니메이션

선, 게임, 음악 등으로 쉽게 표현할 수 있고 웹을 통해서 다른 사람들과 공유할 수 있게 해주는 장점을 가지고 있는 프로그램이다[13].

스크래치 프로그램의 대표적인 특징으로는 다음과 같은 것들이 있다.

첫째, 블록놀이를 하듯이 프로그래밍을 즐길 수 있다. 8개 색상의 스크래치 블록 카테고리에서 원하는 블록 요소들을 스크립트 영역에 drag and drop 방식으로 서로 끼워 맞출 수 있게 되어 있어 구문 에러가 발생하지 않는다.

둘째, 풍부한 미디어의 투입, 조작이 가능하여 재미있게 학습하는 프로그래밍이 가능하다. 그래픽, 애니메이션, 음악, 소리 등을 투입하여 다양한 종류의 작품으로 만들 수 있다.

셋째, 뛰어난 공유성(Sharing)으로 다른 사람과의 협업(Collaboration)이 가능하여 창의적인 작품제작이 이루어진다.

넷째, Scratch 프로그램을 즐기다보면 저절로 반복, 조건, 변수, 데이터형식, 이벤트, 프로그래밍 과정에 대한 개념을 습득하게 된다.

Scratch는 어린 아이들에게 프로그래밍의 기본 개념과 알고리즘을 가르치기 위해 개발되었는데 문제 중심의 게임 만들기나 애니메이션 프로그램 제작활동을 통해 프로그래밍 학습에의 효과성 검증에 대한 시도가 이루어지고 있다[14][15].

2.7 선행 연구의 고찰

본 연구와 관련된 선행 연구를 살펴보면 다음과 같다.

김철동(2007)은 QBASIC을 이용한 초등학교 프로그래밍 교육과정 연구에서 프로그래밍 교육은 문제해결능력을 기르기 위한 논리적, 창의적 사고력 향상이라는 측면에서 초등학교 단계에서부터 실현되어야 한다고 하였다[16].

황진욱(2006)은 컴퓨터 프로그래밍 교육의 중요성 및 효과적인 교육방법 연구에서 컴퓨터 프로그래밍을 효과적으로 교육하기 위해서

는 몇 가지 이론과 모델에 따라 수업하는 것이 아니라 학습자와 학습상황의 특성에 따라서 수준별 수업이 되도록 하여야 하며 다양한 교육방법을 적용하고 가장 적합한 것을 선택하여야 한다고 하였다[16].

조성환, 송정범, 김성식, 백성혜(2008)는 스크래치를 이용한 프로그래밍 수업 효과 연구에서 교육용 프로그래밍 언어(EPL) '스크래치'를 이용하여 중학생을 대상으로 게임제작 프로그래밍 수업을 실시한 결과 남녀모두 메타인지에 긍정적인 효과를 주고 있음을 확인하였다[17].

위에서 살펴본 바와 같이 프로그래밍 교육은 각종 고등 인지 능력을 개발하는데 긍정적인 효과를 보여주고 있으며 최근 그 중요성을 인지하여 우리나라의 프로그래밍 교육이 다양한 방법으로 시도되고 있으나 그 대상이 제한적이다. 이에 본 연구는 초등학생에게 더욱 효과적인 프로그래밍 교육을 실시하기 위해서 학습자의 인지발달단계를 고려한 스크래치 프로그램을 도입하여 본 연구를 실행하고자 한다.

3. 연구의 방법

3.1 연구의 대상

본 연구는 경상남도 김해시에 위치한 관내 초등학교 4학년 학생 30명을 대상으로 실시하며 남녀의 비율은 남학생이 15명, 여학생이 15명이다.

3.2 연구의 설계

본 연구는 Scratch 프로그램을 활용한 프로그래밍 교육이 문제해결력의 향상에 영향을 미치는지 알아보기 위하여 단일집단의 사전 사후 검사 설계 방법을 적용하고자 한다.

남	O ₁	X ₁	O ₂
여			

O₁ : 사전검사(문제해결력 검사)

X₁ : Scratch 프로그램을 활용한 프로그래밍 교육

O₂ : 사후검사(문제해결력 검사)

3.3 프로그래밍 교육내용

구체적 조작기에서 형식적 조작기로 가는 과도기 단계이며 프로그래밍을 처음 접하는 초등 4학년의 수준을 고려하여 스크래치의 기초적 기능과 조작 방법을 익히고 응용·실습하는 단계의 활동으로 구성한다. 수업시간은 주 1회 창의적 재량활동 시간을 활용하여 운영하고자 하며 1차시는 40분씩 운영된다. 각 단계별 주제 및 학습 내용은 다음과 같다.

<표2> 단계별 주제 및 학습 내용

차시	주제	학습내용
1	스크래치 둘러보기	용어 및 기능 익히기
2	말풍선으로 대화출력하기	새 스프라이트 추가, 명령어 블록의 개념 이해
3	애니메이션 효과 연출하기	애니메이션 효과 연출을 위해 모양 바꾸는 명령어 블록 이해, 블록 삭제
4	방향키로 스프라이트 이동시키기	키보드 방향키 활용, 방향 바꾸어 이동, 스크립트 복사
5	드럼 비트에 맞춰 이동하기	드럼 소리 연주, 일정한 간격으로 스프라이트 이동
6	부딪히면 방향 바꾸기	스프라이트 중심확인, 벽에 닿으면 방향 전환
7	마우스 포인터 따라 움직이기	스프라이트 크기조절, 자취 남기기
8	스프라이트 색깔 바꾸기	소리 추가, 색 바꿈 효과 설정, 소리 재생
9	스프라이트에 소용돌이 효과주기	그래픽 효과 설정 방법 이해
10	드럼 비트에 맞추어 색상 바꾸기	스프라이트 이름 바꾸기, 색상 변환
11	미끄러지듯 이동하기	스프라이트 두 개 사용, 특정위치로 이동
12	마우스 위치만큼 소용돌이 효과내기	포인터 위치따라 설정값 변환, 모양 자동변환
13	변수를 이용해 점수 계산하기	변수생성, 점수계산
14	간단한 프로젝트 만들기	게임 만들기
15	간단한 프로젝트 만들기	게임 만들기

3.4 문제해결력 검사도구

연구대상의 문제해결력 검사를 위해서 2003년에 OECD가 실시한 PISA(Programme for International Student Assessment)의 공개된 문제해결력 소양 영역의 19개 문항<표2> 중 10문항을 선정·수정하여 실시하고자 한다<표3>. 선정된 문항은 전문가 집단의 타당도를 검증받았고, Pilot Test를 통해 신뢰도를 분석한 결과 .721(Cronbach α)이었다. 문항별 채점은 정답의 경우 10점, 오답의 경우 0점을 부여하였으며, PISA에 제시된 채점기준에 따라 부분점수 5점을 부여하였다.

<표3> PISA 2003 문제해결력 소양영역 평가요소

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

<표4> 문제해결력 측정 문항

문제형태	문항유형
의사	3(OC), 5(CC), 6(M)
체제	1(OC), 2(M), 7(OC), 8(CC),
문제	4(M), 9(CC), 10(OC)

*형태 : 의사 결정(의사), 체제 분석 및 설계(체제), 문제점 해결(문제)

*유형 : 선택형(M), 개방형 서술형(OC), 폐쇄형 서술형(CC)

4. 결론 및 기대효과

프로그래밍 교육의 중요성은 나날이 제기되고 있으나 초등학교 단계에 도입하기에는 어려운 점이 많다. 이에 본 연구에서는 그에 대한 방안으로 Scratch 프로그램을 선택하여 초등학교 4학년 학생들에게 프로그래밍 수업을

진행하여 학생들의 문제해결력 향상에 영향을 주는지를 검증해보고자 하였다.

Scratch 프로그램을 활용한 프로그래밍 교육은 학생들이 지루하고 어렵게 생각하는 프로그래밍에 흥미를 가지고 적극적으로 참여하여 프로그래밍 교육 자체에 대한 관심을 가지고 그에 따라 문제해결력 향상에 영향을 미칠 것으로 생각된다.

5. 참고문헌

- [1] 신은희, “초등학교 컴퓨터 교육과정의 문제점과 개선에 관한 연구 : 부산광역시 초등학교를 중심으로”, 부산외대 교육대학원 석사학위논문, 2007.
- [2] 김수환, 이원규, 김현철, “개정된 정보교육 과정에서 교육용프로그래밍언어의 교육적 적용방안”, 컴퓨터교육학회논문지, 2009.
- [3] 문외식, “초등학생들이 프로그래밍 학습시 발생하는 오류 유형 분석”, 한국컴퓨터정보학회 논문지, 제11권, 제2호, pp.319-327, 2006년 5월.
- [4] 김용일, “베이직 프로그래밍 언어 교육을 위한 교수·학습 자료의 개발과 적용-초등학교 5,6학년을 대상으로”, 공주교육대학교 교육대학원 석사학위논문, pp.4, 2007.
- [5] 강혜진, “프로그래밍 기초 능력 배양을 통한 아동의 논리적 사고력 신장에 관한 분석”, 숙명여대 교육대학원 석사학위논문, pp.9, 2005
- [6] 전우천, “초등학교 컴퓨터 교육과정 개정의 주요 특징과 의미-소양과 활용의 조화를 통한 정보교육의 실현.”, 교원교육 Vol. 23, No. 1, pp. 461-474, 한국교원대학교교육연구원, 2007년.
- [7] 이주연, “8차 교육과정에 컴퓨터 교육시스템의 개선을 위한 연구 : 프로그래밍 언어 교육 중심으로”, 홍익대 교육대학원 석사학위논문, pp.46, 2007.
- [8] 박미화, “목표모형에 기반한 중등 일반계 프로그래밍 교육과정 개발”, 한국교원대학교 대학원 석사학위논문, pp.17, 2006.
- [9] Mayer, "Problem solving in Encyclopedia of Creativity Vol2, Elsevier, pp.437-447, 1999.
- [10] 임정훈, “상호작용 관점에서 조망해 본 웹기반 교육의 이론적 기저”, 교육공학연구집, 제 15권 3호, pp.29-54, 1999.
- [11] 조성환, 송정범, 김성식, 이경화, “CPS에 기반한 스크래치 EPL이 문제해결력과 프로그래밍 태도에 미치는 효과”, 한국정보교육학회논문지, 제12권 제1호, PP.77-87, 2008.
- [12] 한건우, “프로그래밍 교육에서 문제해결력 신장을 위한 동료 에이전트 시스템 개발”, 한국교원대학교 대학원 석사학위논문, pp.8-10, 2007.
- [13] 이창희, “Scratch Programming”, 에이스 ME, 2008
- [14] 배학진, 이은경, 이영준, “창의성 신장을 위한 Scratch기반 초등 프로그래밍 교육 프로그램 설계”, 한국컴퓨터교육학회 동계학술대회 논문집, 2008년 1월.
- [15] 이채영, 조성환, 이재운, 김성식, “성격유형에 따른 Scratch EPL 수업이 학습자의 자아효능감과 창의성에 미치는 효과 연구”, 한국컴퓨터교육학회 동계학술논문집, 2008년 1월.
- [16] 김철동, “QBASIC을 이용한 초등학교 프로그래밍 교육과정 연구”, 한국정보과학회 논문집, 2007
- [17] 황진욱, “컴퓨터 프로그래밍 교육의 중요성 및 효과적인 교육방법 연구”, 아주대 교육대학원 학위논문, 2006.
- [18] 조성환, 송정범, 김성식, 백성혜, “스크래치를 이용한 프로그래밍 수업 효과”, 한국정보교육학회논문지, 제12권 제4호, pp.375-384, 2008.