

# 초등정보영재 대상의 앱 인벤터 프로그래밍 교육의 효과

서형석, 이용배

전주교육대학교 컴퓨터교육과

## 요 약

초등정보영재교육에 있어서 최근에 강조되고 있는 사안 중 하나는 지식기반 사회를 이끌어갈 정보 분야에서의 창의적 인재의 육성에 있다. 이와 함께 학습자에게는 새롭게 제시되는 문제 상황들을 해결하기 위해 고차원적인 지적능력과 문제해결능력 신장이 더욱 요구되고 있다. 본 연구는 초등정보영재 학생들에게 적용 가능한 앱 인벤터 교육 프로그램을 개발·적용 후 초등정보영재 학생들의 문제해결력 및 자기효능감, 프로그래밍 성취도에 미치는 효과를 검증하고자 하였다. 연구 목적 달성을 위하여 초등정보영재 학생들 수준에 적합한 앱 인벤터 교육 프로그램의 교육과정을 설계하였으며 설계한 프로그램을 렌줄리(Renzulli)의 심화학습 3단계 모형을 변형한 교수·학습 모형을 통해 적용하였다. 초등정보영재 20명을 대상으로 5주간 15차시에 걸쳐 교수·학습활동을 진행하였으며 그 결과 앱 인벤터를 활용한 프로그래밍 교육이 초등정보영재 학생들의 문제해결력, 자기효능감 향상에 긍정적인 효과를 미치는 것으로 나타났으며 프로그래밍 개념과 알고리즘을 이해하는데 효과가 있음을 확인할 수 있었다.

키워드 : 앱 인벤터, 초등정보영재, 문제해결력, 자기효능감, 프로그래밍

## The Effects of Learning App Inventor Programming Education Plan for Gifted Elementary Students

Hyung-Seok Seo, Yong-Bae Lee

Dept. of Computer Education, Jeonju National University of Education

### ABSTRACT

Recently, the emphasis on education of informatics gifted elementary students is to make creative leader in knowledge-based society. It need the high mental faculties and problem solving ability to solve many problem situations for students. In this paper, we developed App Inventor programming education plan. We taught this plan to gifted elementary students and observed the effects of their problem solving ability, self-efficacy and achievement of programing process. In order to achieve the research goals, we designed the appropriate App Inventor programming plan for gifted elementary students and we changed the Learning Model of Renzulli's three step for gifted students to achieve this research goals. Designed learning program was utilized by twenty informatics gifted elementary students for 15 times during five weeks period. In conclusion, we confirmed that App Inventor programming education for gifted elementary students affected the problem solving ability, self-efficacy and achievement of programing process positively.

Keywords : App Inventor, Gifted Elementary Students, Problem solving ability, Self-efficacy, Programming

교신저자 : 이용배(전주교육대학교 컴퓨터교육과)

논문투고 : 2016-12-00

논문심사 : 2017-00-00

심사완료 : 2017-00-00

## 1. 서론

구글의 데미스 하사비스(Demis Hassabis)가 개발한 알파고(AlphaGo)라는 바둑 프로그램은 수천, 수만 가지의 경우의 수가 존재하는 바둑의 확률을 계산하고 스스로 문제를 해결해 나갈 수 있는 능력을 가진 인공지능 프로그램이다.

마크 저커버그(Mark Zuckerberg)가 개발한 페이스북은 2016년 2분기 기준으로 DAU(Daily Activity Users)의 수가 11억 명을 넘었다[5].

데미스 하사비스나 마크 저커버그와 같은 인물들이 세계에 끼친 영향을 생각해 볼 때 정보 분야에서의 영재 혹은 뛰어난 인재의 가치는 매우 높다고 할 수 있으며 그에 따른 정보영재 교육은 매우 중요하고 시급하다고 할 수 있다.

초등정보영재교육에 있어서 최근에 강조되고 있는 사안 중 하나는 바로 지능정보 사회를 이끌어갈 정보 분야에서의 창의적 인재의 육성에 있다. 지능정보 사회 속에서 새로운 문제 상황들은 끊임없이 나타나고 있으며 이를 해결하기 위해 고차원적인 지적능력과 문제해결능력을 신장시키는 것이 필요하다. 그러나 지금까지 우리 영재교육 현장에서의 정보 교육은 컴퓨터 소양 교육과 소프트웨어 활용 교육에 치중해 왔던 것이 사실이다.

최근에 들어서 소프트웨어 연구학교, 소프트웨어 선도학교 및 여러 정보영재교육기관에서 다양한 교육용 프로그래밍 언어(EPL : Educational Programming Language)를 활용하여 프로그래밍 교육을 실시하고 있다. 엔트리(Entry), 스크래치(Scratch), 앨리스(Alice) 등과 같은 블록기반 프로그래밍 언어는 학습자들에게 인지적 부하를 줄여주고 알고리즘을 직관적으로 쉽게 이해할 수 있도록 하는 장점이 있다. 반면 기존의 텍스트 기반 프로그래밍 언어(C언어, Java, Python 등)로 학습하는 것은 프로그래밍 결과물이 컴퓨터라는 가상 세계에서 구현할 수 있기 때문에 학습자가 실제 세계와 연관을 짓지 못하고 결과물을 피상적인 수준에서 확인하는 단점이 있다.

이러한 단점을 극복하기 위하여 가상 세계에서만 확인할 수 있었던 프로그래밍 결과물을 실제 물리 세계에서 구현할 수 있도록 도와주는 교구들이 많이 등장하고 있다. 이를 피지컬 컴퓨팅(Physical Computing)이라고

하며 아두이노(Arduino), 비트브릭(Bitbrick), 레고 마인드스톰(Lego Mindstrom), 메이키 메이키(Makey Makey), 리틀비츠(LittleBits), 코드이노(Codeino) 등이 피지컬 컴퓨팅과 관련된 교구라 할 수 있다. 하지만 이러한 피지컬 컴퓨팅 도구는 프로그래밍 과정보다 물리적인 교구 자체를 학습하는데 다소 시간이 걸리며 개인적으로 교구를 구입하여 관리하기에는 경제적으로 부담스러운 단점이 있다.

앱 인벤터는 초등정보영재 학생들의 고등사고능력을 개발할 수 있고 스마트폰의 물리적 요소(카메라, GPS, 블루투스, 사운드 등)를 쉽게 프로그래밍 교육과 융합할 수 있으며 일상생활의 문제를 해결하는데 더욱 가깝게 접근할 수 있는 스마트폰 애플리케이션(Application) 개발을 위한 교육용 프로그래밍 언어이다[14]. 앱 인벤터는 스크래치와 같은 블록기반 프로그래밍 언어와 유사하게 블록 쌓기를 통해 프로그래밍을 할 수 있으며 다양한 애플리케이션을 제작하면서 주어진 문제를 해결해 나가는데 도움을 줄 수 있다. 또한 애플리케이션을 심화·보충하면서 스스로 오류를 수정하고 산출물을 직접 확인하는 경험을 통해 학습자 본인의 능력에 대한 기대감, 자신감 등과 같은 자기효능감이 더욱 향상될 것으로 기대된다.

이에 본 연구에서는 초등정보영재 학생들에게 적용 가능한 앱 인벤터 교육 프로그램을 개발하였다. 즉 앱 인벤터를 활용한 프로그래밍 교육이 초등정보영재의 문제해결력과 자기효능감에 미치는 영향을 분석해보고 프로그래밍의 개념과 알고리즘에 대한 학습이 어느 정도 도달하는지를 프로그래밍 성취도 평가를 통해 알아보고자 하였다.

## 2. 관련 연구

### 2.1 프로그래밍 교육이 문제해결력과 자기효능감에 미치는 영향

프로그래밍 교육이 문제해결력과 자기효능감에 어떤 영향을 미치는지에 관한 다양한 연구 결과가 나타나 있으며 이를 요약하면 <Table 1>과 같다.

<Table 1> The relationship between programming education and problem solving ability, self-efficacy

Researcher	Contents of research
Bandura[3]	To improve self-efficacy requires achievement experience, substitute experience, verbal persuasion, stability of anxiety
Salomon & Perkins [11]	Programming education improves ability to solve problems, logical reasoning and expressive ability, knowledge and thinking, modeling ability of learning, ability to create cognitive types, passion, perseverance. etc.
Tae-hun Kim [15]	Programming education increases computational thinking ability and extends learning ability beyond programming
Tae-ok Song [16]	Programming education not only learns programming language but also enhances reflective thinking ability by improving logical thinking in programming process and verifying and correcting errors
Yu-sun Lee [18]	Through the programming process, it is necessary to find out how to perform the task and to modify it. This process improves logical thinking ability.
Seong-geun Lee [13]	Programming provides students with a learning environment that enhances their cognitive thinking and problem-solving skills, and has value as a tool to develop problem-solving skills and develop logical thinking and creativity.

2.2 앱 인벤터를 활용한 프로그래밍 교육

스마트폰이 생활화되고 일반 사용자 및 학습자들에게 애플리케이션 개발과 제작에 대한 관심이 확대되었다. 앱 인벤터를 활용한 연구들이 해외에서 활발하게 진행되고 있지만 국내에서는 아직 시작 단계에 있다고 할 수 있다. 하지만 최근 앱 인벤터가 계속 업데이트 되고 있고 현재 영어권 이외의 나라의 언어가 지원되고 있기 때문에 앞으로 국내에서도 앱 인벤터를 활용한 연구들이 활발하게 진행될 것으로 예상된다.

국내·외에서 진행된 선행 연구들을 살펴보면 <Table 2>와 같다.

<Table 2> Prior research contents of domestic and overseas

Researcher	Contents of research
Amber [1]	Study on students' interest in computers through education using app inventory in summer camp for middle and high school students who are interested in computing
Morelli [9]	Study on app inventory training for university students and high school teachers and value research as a tool to improve computing thinking in K-12 course
Wolber[4]	Organized a seminar course for new students of computer major to conduct app inventory training
Moon-gu Seol [7]	Developed an application inventory programming tutorial with a waterfall model. Study on the change of higher thinking ability of elementary school 5th and 6th grade students
Sang-jin An [12]	Analyze the responses of teachers and students who have learned app inventory. Research what you need to consider when creating an App Inventor training plan
Hwa-kyung Rim [6]	Observing changes in learner thinking process after teaching programming using elementary school students' app inventory
Sung-jin Hwang [14]	It is confirmed that education using app inventory has a positive effect on creative problem solving ability and learning commitment level of elementary information gifted

기존 연구를 종합해 보면 앱 인벤터를 활용한 프로그래밍 교육에 대한 효과를 검증하는 부분이 미비하였다. 본 연구에서는 초등정보영재에게 적합한 앱 인벤터 교육 프로그램의 설계에 중점을 두고 프로그래밍 교육 이후 학생들의 문제해결력, 자기효능감, 프로그래밍 성취도의 변화 정도를 알아보려고 한다.

3. 앱 인벤터 교육 프로그램 설계

3.1 교육 프로그램 개발의 방향

첫째, 교육 과정의 난이도는 앱 인벤터를 처음 접하는 초등 4~6학년으로 구성된 정보영재교육 학생들이 학습할 수 있는 수준으로 개발한다.

둘째, 실생활에 유용한 애플리케이션을 제작하는 활동을 통해 다양한 문제를 해결하는 과정 속에서 여러 기능을 익힐 수 있도록 교육 프로그램을 개발한다. 단순히 기능을 전달받고 외우는 형태의 학습 보다는 프로그래밍에 대한 전반적인 이해를 통해 주어진 문제를 해결할 수 있도록 한다.

셋째, 기존의 앱 인벤터 교육 프로그램 연구 내용을 바탕으로 하여 앱 인벤터 교육을 위한 학습 주제를 선정하고 그와 관련한 학습 요소(프로그래밍 개념 및 알고리즘)를 추출하여 학습하고 이를 바탕으로 학생들이 자신의 아이디어를 심화시킬 수 있도록 교육 프로그램을 개발한다.

### 3.2 프로그래밍 도구 앱 인벤터

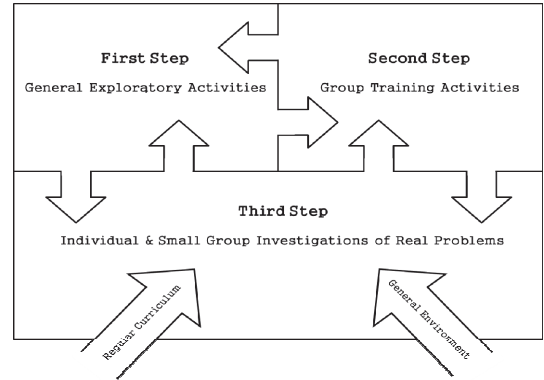
앱 인벤터(App Inventor)는 구글과 MIT(Massachusetts Institute of Technology)가 함께 만든 안드로이드 기반 스마트폰이나 에플레이터에 애플리케이션을 개발할 수 있도록 해주는 프로그래밍 도구이다. 2009년에 개발되어 MIT, 하버드 대학 등 미국 고등교육기관에서 학습도구로 활용되었으며 2010년에 일반인에게 공개되었다. 최근에는 MIT 모바일 학습센터(MIT Center for Mobile Learning)에서 프로그램을 운영하고 있으며 2013년 12월에 앱 인벤터 2 버전이 발표되어 현재까지 사용되고 있다[2]. 앱 인벤터는 교육용 프로그래밍 언어로써 명령어들을 직접 타이핑하는 방식이 아니라 프로그래밍과 관련한 블록들을 이용해 드래그 앤 드롭(Drag & Drop)방식으로 프로그래밍을 할 수 있는 비주얼 프로그래밍 환경을 제공하기 때문에 초보자도 쉽게 접근하고 활용할 수 있다.

### 3.3 교수·학습 모형 설계

본 연구에서는 앱 인벤터 프로그래밍 학습이 효과적으로 이루어질 수 있도록 렌줄리의 심화학습 3단계 모형을 수정하여 적용하였다. 일반적인 렌줄리의 심화학습 3단계 모형의 내용은 [Fig. 1]과 같이 일반적인 탐색 활동 → 소집단 단위의 학습 활동 → 개인 또는 소집단 단위의 문제 해결 활동의 순서로 진행된다[10].

그러나 정보영재학생들의 문제해결력 향상과 자기효능감 향상을 위해 렌줄리의 심화학습 3단계 모형을

<Table 3>과 같이 변형하여 적용하였다.



[Fig. 1] Renzulli's Enrichment Triad Model

<Table 3> Learning Model of App Inventor programming education for gifted elementary students

Step	Learning programming	Activity
Basic navigation activity	Understanding the example given and conceptual learning	-Understanding learning elements and programming principles -Preview the completed app and explore the programming process
Learning activity	Basic examples solving and learning through programming	-Follow the steps to build a basic example app
Deepening problem solving activity	Doing creative programming	-Resolve deepening problems through creative programming

### 3.4 학습 내용

전체적인 학습 내용은 앱 인벤터 익히기, 생활 속 간편 앱 만들기, 간단한 게임 앱 만들기, 퀴즈 앱 만들기 등의 총 4가지의 주제로 구성되어 있으며 학습 내용은 <Table 4>와 같다.

<Table 4> Learning contents of App Inventor programming education

Chapter	Time	Contents and activities	Learning elements
Learning the app inventor	1	-Introducing App Inventor and User Preferences	Understanding programming
	2	-Explore designer and block windows, learn designer components -Handling blocks with event handlers	Sequential execution/ Event handling
	3	-Create a 'talking app'	Sequential execution/ Event handling
Create an easy-to-use app	4	-Create a 'memo app' -(Deepening)Programming to adjust line thickness	Sequential execution/ Variable/ Event handling
	5	-Create a 'paint app' -(Deepening)Add more than 5 colored buttons -Program the picture to be output on the canvas after taking a picture	Sequential execution/ Event handling
	6	-Create a 'sound recorder app' -(Deepening)Add and program the 'pause' button -(Deepening)Create two recording buttons to program each button to record	Sequential execution/ Loop/ Conditional statement/ Event handling
	7	-Create a 'mp3 player app' -(Deepening)Create your own mp3 player to listen to three or more sources	Sequential execution/ Loop/ Conditional statement/ Event handling
Create a simple game app	8-9	-Create a 'dice game app' -(Deepening)Shake your smartphone to program the dice to output two random eyes -(Deepening)Programming to output the sum of two dice	Sequential execution/ Loop/ Variable/ Conditional statement/ Logical operation/ Event handling
	10-11	-Create a 'shooting game app' -(Deepening)Programming to display scores -(Deepening)Added button to initialize all game contents	Sequential execution/ Loop/ Variable/ Conditional statement/ Event handling

Chapter	Time	Contents and activities	Learning elements
Create a quiz app	12-13	-Create a 'mole game app' -(Deepening)Programming by listing the positions of the holes so that the mole appears in the nine holes	Sequential execution/ Loop/ Variable/ List/ Event handling
	14-15	-Create a 'quiz app' -(Deepening)Programming a quiz app that can make a list and solve more than 5 problems	Sequential execution/ Loop/ Variable/ Conditional statement/ List/ Input / output of data/ Event handling

#### 4. 연구 방법

##### 4.1 연구 대상

본 연구에서는 ○○교육지원청 영재교육원의 영재학급에 소속된 4~6학년 정보영재 1학급을 대상으로 선정하였다. 연구 대상자는 실험집단 1학급 20명(남 11, 여 9)으로 구성하였다.

##### 4.2 수행 방법

본 연구는 앱 인벤터를 활용한 프로그래밍 교육이 초등정보영재의 문제해결력, 자기효능감, 프로그래밍 성취도에 미치는 영향을 알아보기 위한 것이다. 실험 처치는 5주간(15차시) 초등정보영재 학생들을 대상으로 실시하였다. 문제해결력, 자기효능감, 프로그래밍 성취도의 효과 여부는 동일집단 전후 검사 결과를 분석하였다.

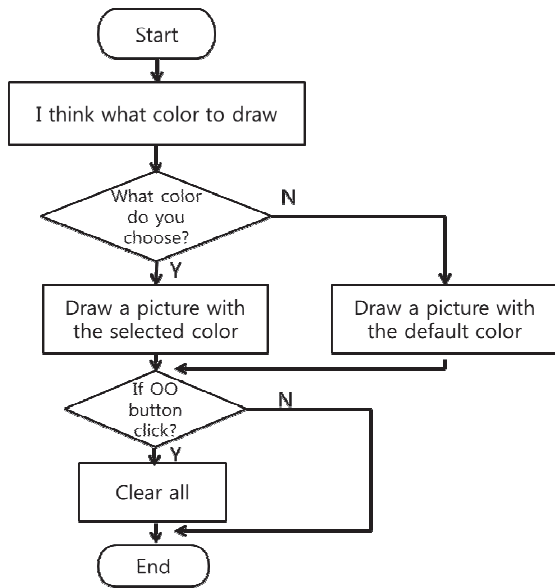
##### 4.3 검사 도구

초등정보영재 학생들의 문제해결력 측정을 위해 우옥희(2000) 문제해결과정 검사지를 사용하였다[8]. 문제해결 과정은 문제의 발견, 문제의 정의, 문제의 해결책 고안, 문제의 해결책 실행, 문제 해결 검토의 5단계로 분류되어 있다.

며 각 단계별로 5문제씩 총 25문항으로 구성되어 있다.

프로그래밍 학습에 대한 일반적 자기효능감의 측정을 위해 최연희(2001)의 자기효능감 검사지를 사용하였으며 활동시작 능력기대, 활동지속 능력기대, 활동수행 능력기대, 혐오 경험극복 능력 기대 등 총 24개 문항으로 구성되어 있다[17]. 각 문항별로는 매우 그렇다, 대체로 그렇다, 대체로 그렇지 않다, 전혀 그렇지 않다는 4단계 척도로 구성되어 있다.

프로그래밍 성취도를 알아보기 위하여 검사 도구는 앱 인벤터 교육 프로그램의 학습 내용을 기초로 하여 프로그래밍 기본 개념과 알고리즘 영역으로 구분하여 검사지를 자체 개발하였다. 프로그래밍 성취도 검사지는 앱 인벤터 교육 프로그램과 관련된 내용으로 구성하되 순서도를 활용하여 문항을 구성하였다. [Fig. 2]는 「그림판 앱」과 관련한 것으로 알고리즘을 이해하고 분석하는 정도를 확인하기 위한 문항의 예시이다.



[Fig. 2] The evaluation question of programming achievement

#### 4.4 연구 결과

##### 4.4.1 문제해결력 검사 결과

문제해결력에 대한 사전·사후 검증 결과는 <Table

5>와 같다.

<Table 5> Problem solving ability pre-post test result (p<.05)

Elements		N	Avg	Std	t	P
Problem solving ability	Pre	20	93.45	10.665	-3.421	.003
	Post	20	101.00	12.469		

문제해결력 검증 결과 사전·사후 검사의 평균은 통계적으로 p<0.05 수준에서 유의한 차이(t=-3.421, p=0.03)가 있는 것으로 나타났다. 따라서 앱 인벤터 프로그래밍 교육이 초등정보영재의 문제해결력 향상에 효과가 있다고 볼 수 있다. 문제해결력의 하위 요소별로 사전·사후 검증을 실시한 결과는 <Table 6>과 같다.

<Table 6> Problem solving ability pre-post test result by sub-element (p<.05)

Elements		N	Avg	Std	t	P
Problem finding	Pre	20	19.65	2.681	-1.594	.127
	Post	20	20.60	3.068		
Problem defining	Pre	20	18.95	3.332	-2.105	.049
	Post	20	20.05	2.800		
Developing solutions	Pre	20	19.15	2.720	-2.077	.052
	Post	20	20.30	3.389		
Applying solutions	Pre	20	16.95	2.523	-2.538	.020
	Post	20	18.75	2.863		
Reviewing	Pre	20	18.75	2.447	-3.780	.001
	Post	20	21.30	2.557		

문제해결력의 하위 요소 중에서 문제의 정의, 문제의 해결책 실행, 문제해결의 검토의 요소에서 사전·사후검사의 평균은 모두 p<0.05 수준에서 통계적으로 유의한 차이가 있는 것으로 확인되었다.

이러한 결과는 본 연구에서 변형·적용한 렌줄리의 심화학습 3단계 모형과 관련이 있다고 볼 수 있다. 학생들은 교사와 함께 주어진 예제를 살펴 보면서 애플리케이션 프로그래밍 과정을 탐색하게 된다. 또 심화된 애플리케이션을 제작하고 그 결과물을 스마트폰이나 태블릿 컴퓨터로 확인해보면서 학생들은 오류를 수정하고 프로그래밍 과정을 익히게 된다. 이러한 학습 활동이 세 가지 하위요소에 긍정적인 영향을 미친 것으로 보인다.



4.4.2 자기효능감 검사 결과

초등정보영재 학생들의 자기효능감 향상에 대한 사전·사후 검증 결과는 <Table 7>과 같다.

<Table 7> Self-efficacy pre-post test result (p<.05)

Elements		N	Avg	Std	t	P
Self- efficacy	Pre	20	76.55	10.395	-2.642	.016
	Post	20	81.25	8.914		

자기효능감에 대한 분석 결과 사전·사후 검사의 평균은 통계적으로 p<0.05 수준에서 유의한 차이(t=-2.642, p=0.016)가 있는 것으로 나타났다. 따라서 앱 인벤터를 활용한 프로그래밍 교육이 초등정보영재 학생들의 자기효능감 향상에 효과가 있다고 볼 수 있다.

자기효능감의 하위 요소별로 사전·사후 검증을 실시한 결과는 <Table 8>과 같다.

<Table 8> Self-efficacy pre-post test result by sub-element (p<.05)

Elements		N	Avg	Std	t	P
Expectation of ability to start activities	Pre	20	9.75	1.743	-1.116	.278
	Post	20	10.15	1.565		
Expectation of activity sustain ability	Pre	20	15.45	2.544	-3.168	.005
	Post	20	17.00	2.077		
Expectation of ability to perform activities	Pre	20	30.55	5.808	-2.154	.044
	Post	20	32.45	6.312		
Expectation of ability to overcome disgust	Pre	20	18.80	3.443	-1.450	.163
	Post	20	19.65	3.646		

자기효능감의 4가지 하위 요소 중에서 활동지속 능력 기대 영역의 사전·사후검사의 평균이 통계적으로 p<0.05 수준에서 유의한 차이(t=-3.168, p=0.005)가 있는 것으로 나타났다. 또 활동수행 능력기대 영역의 사전·

사후검사의 평균이 p<0.05수준에서 유의한 차이(t=-2.154, p=0.044)가 있는 것으로 나타났다. 이는 주어진 예제를 통해 프로그래밍을 학습하고 자신의 아이디어를 투입하여 산출물을 심화시키는 일련의 과정을 거친 결과가 두 가지 하위요소에 긍정적인 영향을 미친 것으로 보인다.

4.4.3 프로그래밍 성취도 검사 결과

초등정보영재 학생들의 프로그래밍 성취도의 향상 여부를 알아보기 위하여 실시한 사전·사후 검증 결과는 <Table 9>와 같다.

<Table 9> Programming achievement pre-post test result (p<.05)

Elements		N	Avg	Std	t	P
Programming achievement	Pre	20	37.00	22.501	-4.744	.000
	Post	20	66.50	19.808		

프로그래밍 성취도에 대한 분석 결과 사전·사후 검사의 평균은 통계적으로 p<0.05 수준에서 유의한 차이(t=-4.744, p=0.000)가 있는 것으로 나타났다. 따라서 앱 인벤터 프로그래밍 교육이 초등정보영재 학생들의 프로그래밍 성취도 향상에 효과가 있다고 볼 수 있다. 프로그래밍 성취도에 대한 평가 문항은 프로그래밍 개념과 알고리즘의 분석, 프로그램의 설계, 실행으로 구성되어 있다. 이들 하위 요소에 대한 사전·사후 검증을 실시한 결과는 <Table 10>과 같다.

<Table 10> programming achievement pre-post test result by sub-element (p<.05)

Elements		N	Avg	Std	t	P
Concept of programming	Pre	20	1.50	4.894	-6.439	.000
	Post	20	13.50	8.127		
Algorithm	Pre	20	33.50	22.775	-3.317	.004
	Post	20	51.00	19.974		

프로그래밍 성취도의 2가지 하위 요소 모두 사전·사후검사의 평균이 통계적으로 p<0.05 수준에서 유의한 차이가 있는 것으로 나타났다.

유의미한 차이의 원인은 다음과 같이 요약할 수 있다.

첫째, 앱 인벤터를 활용한 프로그래밍 학습이 프로그래밍 개념을 이해하는데 효과적이다. 프로그래밍 방법이나 문법적인 지식 습득에 치우쳤던 과거의 프로그래밍 학습 방법에 비해 앱 인벤터는 학생들이 쉽게 조작할 수 있는 블록으로 프로그래밍을 할 수 있기 때문에 프로그래밍 개념을 학습하는데 효과적이다.

둘째, 알고리즘에 대한 접근 빈도가 높아졌다. 학생들이 주어진 과제를 학습하고 심화된 산출물을 제작하기 위해서는 프로그래밍의 과정을 분해해보고, 단계적으로 실행해보는 노력이 필요하다. 이러한 과정을 거치면서 학생들은 알고리즘의 분석, 설계, 실행의 과정을 학습하였다. 산출물을 단계별로 점검하고 실생활에서 활용이 가능한 애플리케이션을 제작하면서 학생들은 자연스럽게 알고리즘을 이해할 수 있게 되었다고 할 수 있다.

## 5. 결론

본 연구는 앱 인벤터를 활용한 초등정보영재 교육 프로그램을 개발하여 이를 적용한 후 초등정보영재의 문제해결력과 자기효능감의 변화, 프로그래밍의 성취도의 향상 정도를 살펴보기 위하여 실시한 연구이다. 검사 도구는 문제해결력 검사지와 자기효능감 검사지, 자체 제작한 프로그래밍 성취도 검사지를 사용하였다.

연구는 초등 4~6학년으로 구성된 초등정보영재 1학년 20명을 대상으로 진행하였다. 실험 처치 전 문제해결력과 자기효능감 검사와 프로그래밍 성취도 검사를 실시하였으며, 앱 인벤터를 활용한 교육 프로그램을 5주간 총 15차시에 걸쳐 진행하였다. 실험 처치 후 문제해결력과 자기효능감, 프로그래밍 성취도 검사를 실시하였으며 분석 결과는 다음과 같다.

첫째, 문제해결력 검사 결과 사전검사의 평균보다 사후검사의 평균이 더 높게 나타났으며 사전·사후 검사의 평균은 통계적으로  $p < 0.05$  수준에서 유의한 차이가 있는 것으로 나타났다. 특히 문제해결력의 5가지 하위 요소 중에서 문제의 정의, 문제의 해결책 실행, 문제 해결의 검토의 3가지 하위 요소에서 사전·사후 검사의 평균이 통계적으로 유의하게 나타났다. 이러한 결과는 학생들이 학습 요소를 습득한 후 그것을 활용하여 자신이 직접 애플리케이션을 수정·보완하고 그 산출물을

스마트폰이나 태블릿 컴퓨터에서 실연하면서 발생하는 오류들을 찾아내고 수정하는 작업을 거치면서 학생들의 문제해결력이 향상된 것으로 보인다.

둘째, 자기효능감 검사 결과 사전검사의 평균보다 사후검사의 평균이 더 높게 나타났으며 사전·사후 검사의 평균은 통계적으로  $p < 0.05$  수준에서 유의한 차이가 있는 것으로 나타났다. 특히 자기효능감의 4가지 하위 요소 중에서 활동지속 능력기대 영역과 활동수행 능력기대 영역의 사전·사후 검사의 평균이  $p < 0.05$  수준에서 유의한 차이가 있는 것으로 나타났다. 이는 학생들이 주어진 예제를 통해 프로그래밍을 학습하고 자신의 아이디어를 투입하여 산출물을 심화시키는 일련의 활동을 수행하면서 학생들이 자신의 활동지속 능력과 활동수행 능력에 대한 기대감을 높인 효과로 해석된다.

셋째, 프로그래밍 성취도에 대한 검사 결과 사전검사보다 사후검사의 평균이 높게 나타났으며 사전·사후 검사의 평균은 통계적으로  $p < 0.05$  수준에서 유의한 차이가 있는 것으로 나타났다. 또 프로그래밍 개념과 알고리즘의 2가지 하위 요소에서도 사전·사후 검사의 평균은 통계적으로 유의미한 차이가 나타났다. 이는 앱 인벤터를 활용한 교육 프로그램이 학생들의 프로그래밍 성취도에 긍정적인 영향을 준 것으로 보인다.

연구 결과를 종합하면 렌줄리의 심화학습 3단계 모형을 변형·적용한 앱 인벤터 교육 프로그램이 학생들의 문제해결력과 자기효능감을 향상시키는데 긍정적인 효과가 있음을 알 수 있었다. 또한 학생들이 프로그래밍의 개념을 익히고 알고리즘을 학습하는데 효과적인 도구임을 확인할 수 있었다.

## 참고문헌

- [1] Amber Wagner, Jeff Gray, Jonathan Corley, & David Wolber(2013). Using App Inventor in a K-12 Summer Camp. ACM special Interest Group on Computer Science Education. 621-626.
- [2] App Inventor. <http://ai2.appinventor.mit.edu/>
- [3] Bandura, A.(1997). Self-efficacy : The exercise of control. NY : W. H. Freeman And Company.
- [4] David Wolber(2011). App Inventor and Real-World



- Motivation. ACM special Interest Group on Computer Science Education. 601-606.
- [5] Facebook. <https://www.facebook.com/>
- [6] Hwa-kyung Rim(2013). Android App. Implementation Teaching using App. Inventor for Elementary school students, *Journal of Korea Multimedia Society* 16(12). pp.1495-1507.
- [7] Moon-gu Seol, Chang-ik Son(2013). A Study on Development of Teaching Materials for App Inventor Programming Using the Waterfall Model, *Journal of The Korea Association of Information Education* 17(4), December 2013. pp.409-419.
- [8] Ok-hee Woo(2000). The Effects of a PBL(Problem-Based Learning) on the Problem Solving Process of Students by Their Meta-cognitive Levels, Major in Curriculum Graduate School of Korea National University of Education Chung-Buk, Korea.
- [9] Ralph Morelli(2010). 'Can Android App Inventor Bring Computational Thinking to K-12?', Computer Science Department, Trinity College Hartford, CT, USA.
- [10] Renzulli, J. S.(1984). The Three Ring Conception of Giftedness : A Developmental Model for Creative Productivity.
- [11] Salomon, G. & Perkins, D. N.(1987). Transfer of cognitive ability from programming : When and how?. *Journal of Educational computing Research* 3, 149-169.
- [12] Sang-jin An, Young-jun Lee(2014). Elementary and Secondary Programming Education Plan Using App Inventor, *Journal of The Korea Association of Computer Education* 17(5).
- [13] Seong-geun Lee(2003). A Web-based Programming Learning System Improve Problem-solving Ability, Major in Elementary Computer Education Graduate School of Education Daegu National University of Education.
- [14] Sung-jin Hwang(2015). Effect of Programming Education using App Inventor on Informatics Gifted Elementary Student's Creative Problem Solving Ability and Learning Flow, Major in Informatics Gifted Education Graduate School of Korea National University of Education Chung-Buk, Korea.
- [15] Tae-hun Kim, Jong-hoon Kim(2013). The Effect of Kodu Programming Learning on Logical Thinking and Learning Interest of Elementary Students, *Journal of The Korea Association of Computer Education* 16(3).
- [16] Tae-ok Song, Sung-hoon Ahn, Tae-young Kim(1999). Design and Implementation of Education Hangul LOGO Language using Object-Oriented Development Method, *Journal of The Korea Association of Information Education* Vol. 2, No.4.
- [17] Yeon-hee Choi(2001). The Relationships Between Children's Attribution Style and Learned Helplessness Self-Efficacy, Major in Educational Psychology Graduate School of Education Korea National University of Education Chung-Buk, Korea.
- [18] Yu-sun Lee(1995). The Computer Programming Education for the Extension of Logical Thinking and Problem Solving Faculties : A comparative study of BASIC and LOGO programming, Major in Education Technology Graduate School of Education Ewha Womans University.

저자소개



**서형석**

2007 전주교육대학교 컴퓨터교육  
과(교육학 학사)

2017 전주교육대학교 컴퓨터교육  
과(교육학 석사)

현재 대흥초등학교 근무

관심분야 : 정보교육, 프로그래밍  
교육

E-Mail : ttaenggam@hanmail.net



**이용배**

1998 충남대학교 컴퓨터과학과(이  
학석사)

2003 충남대학교 컴퓨터과학과(이  
학박사)

2003-현재 전주교육대학교 컴퓨  
터교육과 교수

관심분야: 알고리즘 교육, 정보 검  
색, 자동 분류

E-Mail: yblee@jnue.kr