

블록코딩 선행학습자를 위한 Python 교육 프로그램 개발

김태령 · 한선관

경인교육대학교 융합교육과

요 약

본 연구는 블록 코딩을 선행 학습한 학생들에게 적용할 수 있는 Python 교육 프로그램의 개발에 관한 것이다. 우선 초중등학교 교육과정과 EPL 교재의 분석을 통해 블록코딩 학습자의 수준을 분석하고 블록 코딩에서 사용된 개념을 중심으로 Python 문법을 추출하여 교육과정을 구성한 뒤 PBL 수업에 맞는 Python 교육 프로그램을 총 16차시로 개발하였다. Python 교육프로그램의 적절성을 검증하기 위해 2차의 전문가의 타당도 검사를 하였다. 검사 결과, 1차 타당도 24문항에서 CVR값 .78 점 이상으로 나타나 일부 수정, 보완하였다. 2차 검사에서는 21개 문항은 타당성을 확보하였고 CVR 최소값인 .99 이하인 3개 문항의 내용에 대해 수정하여 교육 프로그램을 완성하였다. 개발된 교육 프로그램이 스크립트코딩을 학습하기 위한 기초 자료로 유용하게 활용될 것 기대한다.

키워드 : Python, 블록코딩, 교육용 프로그래밍 언어, 소프트웨어 교육

Development of Python Education Program for Block Coding Learners

Taeryeong Kim, Sungwan Han

Dept. of STEAM Education, Gyeong-in National University of Education

ABSTRACT

In this study we have developed a Python education program that can be applied to students who have studied block-based coding. We have developed a Python education program based on the extracted the learners' level of block-based coding by analyzing the programs and the textbooks. We extracted the grammar of the block-based coding and constructed the curriculum. Then, the Python education program was composed by 16 hours. After reviewing the appropriateness of the education program through expert validation, it was concluded that the developed Python education program is suitable for applying to learners of block-based coding. We expect that proposed program will be effectively applied as basic resources to learn script coding in class.

Keywords : Python, Block Coding, Educational Programming Language, SW education

교신저자 : 한선관(경인교육대학교)

논문투고 : 2017-12-07

논문심사 : 2017-12-21

심사완료 : 2017-12-22

1. 서론

2015 개정 교육과정에서는 소프트웨어의 중요성을 역설하고 초·중등교육과정에 소프트웨어 교육을 도입하여 협업적 문제해결 과정을 통해 컴퓨팅 사고력과 의사소통능력, 공동체 의식을 함양하도록 하였다[6]. 이를 위해 스크래치나 엔트리와 같은 블록코딩 기반의 EPL(Educational Programming Language)을 활용하여 초등학생들에게 기초적인 프로그래밍을 체험하도록 하고 중등학생들에게는 순차, 선택, 반복, 변수, 연산 등의 학습요소를 종합적으로 활용하는 프로그램을 작성하도록 하였다.

Python은 2017년 7월 기준으로 공학 및 응용과학 분야의 IEEE Spectrum에서 디바이스 벨, 트렌딩, 고품 수요, 오픈 소스의 활용 등 프로그래밍 인기도 순위에서 모두 첫 번째 언어로 자리하였고[15], 최근 인공지능 이슈의 등장과 빅데이터의 중요성이 떠오르면서 프로그래밍뿐만 아니라 머신러닝과 데이터 분석에 강점을 가진 Python이 해당 분야의 연구에 다양하게 쓰이고 있다[12].

EPL 이후의 교육과정에 대해서 외국 사례를 살펴보면 영국의 경우에는 Key Stage 3에서는 반드시 TPL(Text-based Programming Language), 즉 스크립트 코딩을 학습하도록 하고 있다[14]. 영국뿐만 아니라 핀란드, 인도, 호주 등의 경우에도 우리나라의 중학교 1학년 수준에서 TPL을 학습하고 있다[10].

이처럼 높은 활용 가능성과 직업적 수요, 많은 적용 사례를 가진 Python은 우리나라의 고등학교의 선택과목인 정보교과에 이제 막 도입되기 시작하였으나 정보 교사의 경우 대부분 C언어를 주로 학습하였고 처음 도입되는 시기이기 때문에 낯설어 하는 것도 사실이다. 게다가 TPL에서는 EPL과는 다른 코딩환경과 문법적 차이로 인하여 배우거나 가르치는데 여러 어려움에 부딪치게 된다[5]. Milne과 Rowe가 지적한 것처럼 텍스트 언어에서는 메모리 할당 부분이 지나친 추상화로 인해 어려움을 겪는다거나 Piteria와 Costa가 지적한 것처럼 참조, 매개변수, 에러 핸들링, 라이브러리 활용에 어려움을 겪어 포기하는 경우도 많아진다[2][7].

그러나 Python의 경우 자료형 같은 전통적인 프로그래밍 도구에서 겪는 어려움이 Python 언어의 특징으로 인해 적게 나타나는데다 우리나라의 경우 EPL을 도입하는 시기이기 때문에 학습자의 선수학습 요건이 텍스트 언어 교육 시 인지적 부담을 줄여 TPL 도입하는데

큰 도움을 줄 것으로 기대한다.

따라서 본 연구에서는 앞으로 블록코딩 기반의 EPL을 학습하게 될 학생들에게 적용할 수 있는 Python 교육 프로그램을 설계하여 구체적인 콘텐츠를 개발하고자 한다.

2. 이론적 고찰

2.1 교육용 프로그래밍 언어와 Python

교육용 프로그래밍 언어(EPL, Educational Programming Language)란 교육적인 목적을 가지고 개발된 프로그래밍 언어이다. 교육용 프로그래밍 언어의 선택 기준은 친근감을 주는 언어, 프로그램 작성이 용이한 언어, 확장성이 있는 알고리즘 언어, 호환성과 전이성이 있는 언어이다[8]. 대표적인 EPL은 스크래치와 엔트리가 있으며 주로 블록형 명령어를 조립하는 형태로 코딩하는 방법을 사용한다. 대표적인 스크립트 코딩 언어인 Python은 귀도 반 로섬(Guido van Rossum)이 교육용으로 창시한 언어로 개방형, 공동체 기반 개발 언어로 최소화된 기호를 통한 간단하고 강력한 문법, 강력한 명령어와 사용자와의 상호작용성, 구조적 설계의 강력함, 오픈소스와 용이한 재사용성을 특징으로 하는 언어이다[5].

2.2 선행연구

블록 코딩 기반의 EPL 선행 학습자들을 위한 Python 교육 프로그램을 개발하기 위해 선행 연구를 분석하였다.

김경미 외는 대학의 교양교육과정에 프로그래밍 교육의 일환으로 Python을 5주간 실시한 결과 사고력, 문제해결능력, 창의력, 의사표현능력, 자기주도적 학습 능력이 발달하였음을 확인하였다. 교육과정으로 Python 소개, 알고리즘 기본, 변수, 연산자, 터틀 라이브러리, 조건, 반복, 자료형, 함수 등의 내용을 주제별로 나누어 단계별 과제를 제시하여 내용을 누적할 수 있도록 하였다[4]. 이준영은 스토리텔링 방식으로 교육한 결과 학생들이 이를 어렵지 않게 코딩의 문법과 알고리즘 사고를 받아들이고 인터프리터의 직관성을 통해 명령어의 오류

를 쉽게 수정할 수 있음을 밝혔다[3]. 김성수는 초등학교생을 대상으로 스크래치의 문법과 Python의 문법을 일대일로 대응하여 교육한 결과 학업성취도와 흥미도 면에서 향상을 확인하였다[13].

이상의 선행연구를 분석해보면 Python 교육과정 개발 및 적용에 관한 연구를 통해 Python 교육 적용 가능성 및 적용 방법을 탐구하였다. 대상 학생들의 연령이 낮을수록 인지적 부담을 줄이고 비교적 간단한 내용을 통하여 교육을 실시했으며 원리 이해에 많은 시간을 할애하였다. 반면 스크래치의 개념을 활용한 교육 프로그램의 연구에서는 스크래치와의 대응을 통해 개념 이해를 빠르게 하였으므로 EPL 학습자의 경우 기존 개념을 적절히 활용하여 교육할 필요성을 찾을 수 있었다. 이러한 시사점에도 불구하고 Python의 기초적인 부분 외에 프로그래밍 언어를 종합적으로 가르치는 교육 프로그램에 대한 연구는 아직까지 존재하지 않으며 특히나 현재의 소프트웨어 교육 도입 시기의 알맞은 프로그램은 Python이 가진 가능성에 비해 연구가 부족함을 알 수 있다.

3. 연구 내용 및 방법

3.1 연구의 내용 및 절차

프로그램을 개발하기 위해 분석(Analysis), 설계(Design), 개발(Development), 구현(Implement), 평가(Evaluation)의 단계를 순환적으로 이루어지는 ADDIE 교수체제설계의 개발 모형을 적용하였다.

분석에서는 현재 블록 코딩 기반의 EPL 학습 형태를 살펴보고 Python의 교육 방법을 적용하도록 하였다. 설계에서는 분석을 토대로 개발 방향을 설정한 후 교육 프로그램의 과정 및 교수학습 방법을 설계하였다. 개발 과정에서 전체 교재를 개발하도록 하고 구현하였다. 이후 전문가의 검토를 통해 프로그램의 타당도를 검토하여 프로그램의 내용을 평가하였다. 본 연구는 실제 적용 전 교육 프로그램을 구성하고자 하는 데에 목적이 있으므로 실행과 평가 대신 전문가의 검토를 통해 개발 내용에 대한 피드백 과정을 대체하고 그 결과를 토대로 블록 코딩 기반의 EPL 선행 학습자에게 적합한 Python 교육과정을 제시하였다.

3.2 분석(Analysis)

분석은 Python 교육의 필요성에 의거하여 기존의 2015 개정 교육과정의 성취기준과 시중의 EPL 교재를 중심으로 진행되었다. 성취기준의 해설과 EPL 교재의 구성을 통해 학습자들이 EPL 프로그래밍 도구를 배울 때 어떤 방식으로 배우는가에 대한 분석을 실시하였다.

2015 개정 교육과정 중 EPL 프로그래밍 도구를 활용하여 교육하는 내용으로는 초등학교의 절차, 조건, 반복을 체험하고 중등 정보 교육과정에서 추가로 변수와 연산자를 이용하여 자료의 입출력과 처리가 가능한 프로그램을 설계하고 구현하는 것을 목표로 한다.

기존의 EPL 교재는 인터넷 도서 구입 사이트 중 가장 인기가 높은 인터넷사이트에서 스크래치와 엔트리로 검색(yes24, 2017. 8. 기준)하여 IT모바일 분야에서 인기도 순으로 정렬하여 피지컬 컴퓨팅 분야와 자격증 서적을 제외한 상위 38개 서적의 목차를 확인하여 학습 내용의 순차가 어떠한 방식인지 살펴보았다. 대부분의 경우 문제중심 또는 주제중심 등의 구성주의 학습(13권)이었고 전통적인 프로그래밍 문법 중심의 학습(9권)이 그 뒤를 이었다. 또한 컴퓨팅 사고력에 따라 학습 순서를 구성한 책(5권)들이 그 뒤를 이었다. 기존의 프로그래밍 관련 책들이 대부분 문법 및 프로그래밍 개념을 중심으로 이루어진 것과는 대조적으로 학습자 중심의 구성주의 교수방법이 <Table 1>과 같이 많은 비율을 차지하였다.

<Table 1> Teaching order in EPL Books

Teaching Method	PBL SBL	Grammar	CT centered	STEAM	Etc
Count	13	9	5	7	4
Ratio	34%	24%	13%	18%	11%

Python 관련 서적은 인터넷 사이트의 인기도 순위를 기준으로 기초부터 제시한 서적 중 상위 14개를 선정하여 분석한 결과 개념과 문법 중심의 서적이 압도적임을 확인할 수 있었다. 텍스트 프로그래밍의 경우 대상의 연령층이 높고 문법 학습에 시간이 더 오래 걸리기 때문으로 풀이된다.

<Table 2> Teaching order in Python Books

Teaching order	Subject Based	Grammar Based
Count	1	13
Ratio	7%	93%

3.3 설계(Design)

Python 교육 프로그램 설계를 위해서 교과용 도서 선정 기준을 참고하여 개발방향, 학습 수준, 학습 순서, 학습 단계, 소재 선정, 교수 전략으로 설정하고 각각의 개발 기준을 PBL 학습과 프로그래밍 교육 원리에 따라 선정하였다.

개발 방향은 분석을 통하여 선정된 EPL 기 학습자와 PBL 방식의 교육 프로그램을 혼합하여 설정하였고 다른 분류 및 개발기준의 상위 목표로 설정하였다. 학습 수준은 교육과정 및 외국의 사례와 시대적 요구에 따라 형식적 조작기의 학생을 대상으로 하여 순차, 반복, 조건, 변수 등이 포함된 산출물을 만들어낼 수 있는 대상으로 하였다.

학습 순서의 경우 텍스트 프로그래밍 문법의 순서를 고려하여 EPL에서 접하여 충분히 개념적으로 익숙한 단계별로 제시하고 그 이후에 EPL에서 배우지 않은 객체지향, 함수 등으로 나아갈 수 있도록 한다. 문제의 경우도 CT Concepts의 중첩이 낮은 곳에서 높은 곳으로 흐를 수 있도록 하였다.

학습 내용의 단계는 PBL의 단계를 활용하여 학습자가 자기주도적 학습과 협력학습을 할 수 있도록 단계에서 문법을 배우는 과정과 프로그래밍 문법을 익숙하게 할 수 있는 연습이나 다양한 예제의 실제 실습과정을 배치하였다. 소재의 선정은 PBL의 철학에 어울리는 문제를 선정하지만 지나치게 비 구조화된 상황으로 집중력이 떨어지는 문제가 일어나지 않도록 적절한 수준의 문제를 제시하였다. 또한 연습 과정에서도 개발보다 문제해결력에 초점을 맞추어 실제 예시로 다양하게 구현할 수 있는 문제를 이용하였다.

학습의 전략은 PBL의 단계와 프로그래밍 교육 원리를 결합하여 학습의 전이를 일으킬 수 있는 문제를 제시함으로써 학습을 이끌어가는 방법을 이용하고 문법 연습의 과정에서는 모방과 시연을, 문법의 적용을 통한 연습에서는 자기주도적 학습을, 선행 지식과 연계하는 장에서는 협동 학습을 통해 계열화된 단계의 수준에 알맞은 전략을 실행하도록 하였다. 문제를 해결하는 과정

에서는 프로젝트 형태로 문제 해결안의 설계와 개발을 통한 해결을 실행할 수 있도록 하였다. 위 6가지의 개발 기준을 명세화하여 <Table 3>으로 제시하였다.

<Table 3> Python Education Program Design

Classification	Principle
Development Direction	<ul style="list-style-type: none"> Python education program by Problem Based Learning for the student that already learned EPL For the student that already learned EPL. For the students that already know basic CT Concept like sequence, loop, condition, operator, data For the student that in early formal operational stag
Learning Level	<ul style="list-style-type: none"> From easy to difficult in aspect of problem algorithm From easy to difficult in aspect of programming grammar From small to large in aspect of CT concept overlap involved in the problem
Learning Order	<ul style="list-style-type: none"> First - Present a authentic problem & Introduce grammar Second - Programming Grammar Practice third - Programming Example exercise fourth - Various products for problem solving
Learning Content Step	<ul style="list-style-type: none"> Ill-structured & Authentic problem
Material Selection	<ul style="list-style-type: none"> Appropriate level Focus on solving problem Programming example that have autonomy
Teaching Strategy	<ul style="list-style-type: none"> First - Direct Method Second - Self-directed learning (Grammar Practice & Application) Third - Small group cooperative learning (Link to prior knowledge) Fourth - Project based

문헌 분석과 선행 연구의 프로그래밍 교육 원리에 따라 개발 기준 및 방향을 설정한 후 분석한 책에서 기본적인 과정으로 제시하는 내용을 중심으로 문법을 선정하였다.

공통적으로 제시하는 개념 및 문법은 변수(리스트 등), 연산, 조건, 반복, 함수, 객체지향이었다. 이를 명령어 단위로 분해한 뒤 제시할 문제를 CT Concepts의 중첩도를 고려하며 개발하였다. 이에 따라 구성한 교육과정은 <Table 4>와 같다.

<Table 4> Python Education Program Curriculum

Hour	Curriculum	
1	Subject('S')	Hello, Python
	Command('Cd')	print
	Primary CT Concepts('CT')	operator
2	S	What is BMI index?
	Cd	variable
3	CT	sequence, operator, data
	S	Unit converter
	Cd	input, int, float, remark(#)
4	CT	operator, data, event
	S	Resident registration number judgment
5	Cd	conditional, data, event
	CT	if, elif, else, True & False
6	S	Factorial Calculation
	Cd	for(range), while
7	CT	loop
	S	Turtle racing
8	Cd	import, pyturtle's command(library)
	CT	sequence, loop
9	S	Diagram vending machine
	Cd	pyturtle's command(library)
10	CT	sequence, loop, conditional
	S	Optical art by pyturtle
11	Cd	random
	CT	sequence, loop, operator
12	S	Morpheme analysis
	Cd	String manipulation (str, find, len, replace, split, type)
13	CT	data
	S	Bingo game
14	Cd	List command(remove, insert, append, sort, count, index)
	CT	data
15	S	Morse code conversion
	Cd	Dictionary command(keys, items)
16	CT	data
	S	Diagram stamp
17	Cd	def, random
	CT	sequence, event
18	S	Draw trees
	Cd	def(parameter)
19	CT	sequence, loop, event
	S	Caesar password
20	Cd	built-in functions, external functions
	CT	sequence, loop, conditional
21-22	S	Game transform
	Cd	All Commands
23-24	CT	conditional, loop, event, operator, data

3.4 1차 타당도 검토

개발 방향에 따른 교육과정 설정과 개발한 한차시 분량의 예시 교재를 토대로 1차 내용 타당도 검증을 실시하였다. 검토에 참여한 전문가는 관련 석사 학위 이상을 가지고 있거나 EPL 과정을 현장에서 적용하는 교육자 9명으로 구성하였다. 타당도 검토에는 리커트 5점 척도를 이용하였고 Lawshe의 내용타당도비율(CVR)을 이용하여 검증하였다[1]. 개발 방향과 교육과정에 대하여 조사한 타당도 결과는 <Table 5>와 같다.

<Table 5> 1st Content Validity Ratio

Question	CVR
(1) Development Direction	1
(2) Learning Level	.78
(3) Learning Order	1
(4) Learning Content Step	1
(5) Material Selection	.78
(6) Teaching Strategy	1

개발 방향에 따른 교육과정 및 예시교재의 내용타당도 결과 사례수 9명에 대한 타당도 확보의 CVR 최소값인 .78을 충족하였으므로 설계 단계에서의 개발 방향에 따른 교육과정 및 예시교재의 내용타당도가 확보되었다.

3.5 프로그램의 개발

1차 전문가 검토를 토대로 총 16차시 분량의 전체 교재를 개발하였다. 개발 교재는 설계 원리에 따라 정해진 교육 과정을 통해 순서대로 개발하였으며, 1차 타당도 부분에서 검증받았더라도 점수가 평균에 비해 낮은 부분에 대하여 수정 및 고안 후 1차에 제시되었던 한 차시 분량 예시의 구성과 디자인을 이용하여 전체 16차시의 교재를 개발하였다. 순서대로 개발하는 과정에서 교재 내용에서 학습의 계열성을 검증하는 작업을 하였으며, 지식의 확장을 위해 주 학습내용이 아닐지라도 비슷한 개념의 문법이나 매개변수와 같은 성질은 참고용으로 제시하였다. 특징이 있는 부분은 학습의 중간 부분인 객체지향 라이브러리를 이용하는 부분에서는 8차시에 학생들의 다양한 산출물을 통해 학습의 흥미도를 제고할 수 있도록 창의적인 설계와 성취감을

느낄 수 있는 차시로 변형하여 제시하였다. 16차시 분량을 개발하고 제시된 예제들의 프로그래밍 기준 답안과 학습 방향성에 대한 해설을 추가하여 실제 교수활동 시 사용할 수 있는 교사용 교안까지 제시하여 전체 교육 프로그램의 교재 개발을 하였다.

2차 전문가 타당도 검토에서는 전체 학생용 교재와 교사용 교재, PBL 교수학습 모형을 구분하는 세부 프로그래밍 언어 교수학습 전략을 포함하여 기존에 제시했던 개발방향, 교육과정과 함께 타당도 검증을 실시하였다. 2차 타당도 검토에서 새롭게 추가된 내용 중 교육 프로그램의 교재의 단계에 따른 교수학습 방법은 PBL 단계를 따랐으며 그에 따른 교수 전략은 <Table 6>과 같다.

<Table 6> Composition of Dimension of AI Image

PBL	Teaching Method	Teaching Strategy
Presentation	▷ INTRO	
	· Pose a problem · introduce a learning contents for solving problem	
Provisionally solving problem and find preconception	▷ CT LESSON	Demonstration and imitation
	· Demonstration and imitation target command	
Self-direct learning	▷ CT LAB	Practice and Application
	· Practice and utilization target command in various situation	
Cooperative learning	▷ CT PRACTICE	Link to prior knowledge
	· Applying in various situation · Solving small problem conjunction with pre-content	
Solving	▷ CT CHALLENGE	Project Design-Development
	· Plan the algorithm · Solving problem · Debugging	
Evaluation	▷ Organizing&Evaluating	
	· Orgarnize work and lessons · Compare a work	

3.6 2차 타당도 검토

2차 타당도 검토는 Python 전체 교육 프로그램이 들어가고 각 내용에 대한 검증이 필요하기 때문에 교육학을 전공하고 EPL 및 Python을 교육을 적용해본 사례가

있는 전문가 5인을 선정하여 진행하였다. 검토 과정에서 자세한 정보를 얻기 위해 질문을 세분화하고 리커트 5점 척도와 개방형 문항을 사용하였고 마찬가지로 CVR 검증을 이용하였다. 결과는 <Table 7>과 같다. 블록코딩 기반 EPL 선행 학습자를 위한 Python 교육 프로그램의 2차 전문가 검토를 통해 설문한 문항 23개 중 총 20개의 문항에 관하여 타당하다는 결론이 내려졌다. 3개 문항의 경우 5인에 대한 CVR 검증의 최소값인 .99를 충족하지 못하였다. 개방형 문항을 통해 충족하지 못한 문항에서 두 번째와 세 번째는 PBL의 몇가지 주제의 학습 문제에서 형식적 조작기 초기에 맞는 현실적인 문제가 제공될 필요가 있다는 의견과 PBL의 경우 관련 내용을 처음 배우는 학생의 경우 해매지 않도록 더 자세한 자료 제시가 필요하다는 의견이 있었고, 첫 번째는 이에 대한 의견에 동조한 것으로 확인하였다.

<Table 7> 2nd Content Validity Ratio

Question	CVR
(1) Development Direction	0.60
(2) Learning Level	1.00
(3) Learning Order	1.00
(4) Learning Content Step	0.90
(5) Material Selection	0.92
(6) Teaching Strategy	1.00

2차 전문가 타당도 검토 결과를 통해 2차 수정을 실시하였다. 개방형 문항에서 의견이 나온 5차시의 부분의 제공되는 문제를 수정하고 이에 수반되는 문제 상황을 변형하였으며 9차시의 경우 문제의 개념이 어렵지 않으므로 추가 설명을 제공하여 문제에 대한 이해를 도울 수 있도록 한 후 최종 프로그램을 개발하였다.

<Table 8> Corrected version

Hour	Curriculum
5	S Additional Calculation
	Cd for(range), while
	CT loop
9	S Morpheme analysis
	Cd String manipulation (str, find, len, replace, split, type)
	CT data

4. 결론 및 제언

본 연구는 'EPL을 학습한 학생들에게 그 이후의 프로그래밍 교육으로 무엇을 하면 좋을까?'의 문제로부터 출발하였다. 이를 위하여 ADDIE 모형에 근거하여 현재의 상태와 문헌 분석을 통해 Python 교육 프로그램을 설계하고 개발하였다. 개발 과정에서는 EPL 학습 방식과 선행연구, 프로그래밍 교재, 프로그래밍 교육 원리를 토대로 PBL 기반의 16차시 분량의 Python 교육 프로그램을 구성하였다. 프로그램의 적절성은 두 차례의 전문가 타당도 검토를 통해 개발 원리에 따른 프로그램의 타당성을 확보하였고 두 차례의 수정을 통해 최종으로 블록코딩 기반 EPL 선행 학습자를 위한 Python 교육 프로그램을 개발하였다.

본 연구에서 도출한 교육 프로그램은 문헌 분석 및 전문가 타당도를 통해 이루어졌으므로 교육의 효과성 및 만족도를 검증하기 위해서는 구성된 교육 프로그램과 교수 설계를 기반으로 실제 적용을 통해 효과성에 대한 후속 연구가 필요할 것이다. 교육용 프로그래밍 언어를 효과적으로 가르치기 위한 선행연구들[9][10]이 많이 개발되고 있지만 본 연구에서 제시한 블록 코딩 선행 학습자들을 대상으로 Python 등의 스크립트 프로그래밍 언어를 가르치기 위한 효과적인 교수학습 전략의 개발에 관한 다양한 연구도 필요하다.

소프트웨어 교육에 대하여 관심이 높아지고 있는 우리에게 다양한 프로그래밍 교육에 대한 연구는 하나의 담론으로서의 기능을 할 것이다. 본 연구 역시 준비된 미래로 이끌어 갈 수많은 노력 중 하나가 되길 기대한다.

참고문헌

- [1] C. H. Lawshe (1975). A quantitative approach to content validity. *Personnel psychology*, 28(4), 563-575.
- [2] I. Milne, G. Rowe (2002). Difficulties in learning and teaching programming—views of students and tutors. *Education and Information technologies*, 7(1), 55-66.
- [3] J. Y. Lee (2015). Designing an instructional model for the python programming using storytelling methods. Yonsei University Master thesis.
- [4] K. M. Kim, H. S. Kim (2014). A Case Study on Necessity of Computer Programming for Interdisciplinary Education. *Journal of digital convergence*, 12(11), 339-348.
- [5] L. Grandell, M. Peltomäki, R. J. Back, T. Salakoski (2006). Why complicate things?: introducing programming in high school using Python. *In Proceedings of the 8th Australasian Conference on Computing Education*, 52(1), 71-80
- [6] Ministry of Education (2015). *2015 Revised Curriculum Practice/Informatics Curriculum* (No2015-74).
- [7] M. Piteira, C. Costa (2013, 7). Learning computer programming: study of difficulties in learning programming. *In Proceedings of the 2013 International Conference on Information Systems and Design of Communication, Lisboa. ACM*, 1(1), 34-42.
- [8] M. Y. Ryu, S. K. Han (2015). Development of Computational Thinking-based Educational Program for Software Education. *Journal of The Korean Association of Information Education*, 19(1), 11-20.
- [9] S. H. Kim, S. K. Han (2012). Design-Based Learning for Computational Thinking. *Journal of The Korean Association of Information Education*, 16(3), 319-326.
- [10] S. J. Jun, S. K. Han (2016). Development of UMC Teaching and Learning Strategy for Computational Thinking. *Journal of The Korean Association of Information Education*, 20(2), 109-117.
- [11] S. K. Shin, Y. K. Bae (2015). Study on the Implications about Curriculum Design through the Analysis of Software Education Policy in Estonia. *Journal of The Korean Association of Information Education*, 19(3), 361-372.
- [12] S. Mohammed, O. Mohammed, J. Fiaidhi, S. J. Fong, T. H. Kim (2013). Classifying Unsolicited Bulk Email(UBE) using Python Machine Learning Techniques, *International Journal of Hybrid Information Technology*, 6(1), 43-56.
- [13] S. S. Kim (2014). Developing a Python Programming Instruction Model Using Scratch: Focusing on the 5th Grade of Elementary School. Yonsei University

Master thesis.

- [14] CAS (2013). Computing in the national curriculum: A guide for primary teachers. Retrieved December 5, from <http://www.computingschool.org.uk/data/uploads/CASPrimaryComputing.pdf>
- [15] IEEE Spectrum (2017). Interactive: The Top Programming Languages. Retrieved July 18, from <https://spectrum.ieee.org/static/interactive-the-top-programming-languages-2017>.

저자소개

김 태 령



2012 경인교육대학교
(교육학학사)
2012 경인교육대학교 융합인재교육 석사과정
2016~서울세검정초등학교 교사
관심분야: SW교육, Computational Thinking, STEAM교육, Unplugged Computing, SW 융합, 프로그래밍
e-mai: crossallover@gmail.com

한 선 관



1991 경인교육대학교
(교육학학사)
1995 인하대학교 교육대학원
(컴퓨터교육학석사)
2001 인하대학교 전자계산공학과
(전산학 박사)
2002~현재 경인교육대학교 컴퓨터교육과 교수
관심분야: 창의컴퓨팅 교육, SW교육, 인공지능, ITS, STEAM교육, 초등정보교육, 미래교육
e-mail: han@gin.ac.kr