

블록 프로그래밍 환경 기반 온라인 평가 시스템 개발

심재권[†] · 채정민^{††}

요 약

초중등 프로그래밍 교육에서 Scratch로 대변되는 블록 프로그래밍 환경은 학습자의 특성과 인지수준에 적합하여 활용이 권장되고 있을 뿐 아니라 프로그래밍 초보자도 쉽게 사용할 수 있어 전세계적으로 활용되고 있다. 블록 프로그래밍 환경 이후 데이터 처리과정에 대한 이해, 알고리즘의 정교성과 효율성 측면에 대한 이해, 간단한 SW를 제작하는 활동을 위해 텍스트 프로그래밍 환경으로의 전이는 필수적일 뿐 아니라 교육과정에서도 단계적으로 제시하고 있다. 본 연구에서는 블록 프로그래밍 환경에서 텍스트 프로그래밍 환경으로 전이하기 위한 목적으로 온라인 평가 시스템인 WithBlock를 개발하였다. 개발한 시스템은 동일한 알고리즘 문제를 블록과 텍스트 프로그래밍 환경 모두에서 풀이가 가능할 뿐 아니라 작성한 코드를 자동으로 채점하여 즉시적인 피드백을 제공할 수 있어 초중등 프로그래밍 교육에 활용할 수 있다. 개발한 시스템의 프로그래밍 교육에 적용 가능성을 분석하기 위한 목적으로 사용성, 학습 가능성, 흥미와 만족을 설문하였고, 설문결과 프로그래밍 교육에 활용 가능성을 보여주었다.

주제어 : 블록 프로그래밍, 텍스트 프로그래밍, 전이, 프로그래밍 교육

Development of On-line Judge System based on Block Programming Environment

Jaekwoun Shim[†] · Jeong Min Chae^{††}

ABSTRACT

Block programming environment, which is represented by Scratch in elementary and middle school programming education, is suitable for learner's characteristics and cognitive level, and is recommended not only for beginners. Transference to the text programming environment after the block programming is essential for understanding the data processing process, understanding the accuracy and efficiency aspects of algorithms, and creating SW activity. In addition, it is presented step by step in the programming curriculum. In this study, developed WithBlock the online evaluation system for the purpose of transference from a block programming to a text programming environment. The developed system can solve the same algorithm problem in both block and text programming environment, and it can be used for elementary and secondary programming education by automatically scoring the written code and providing immediate feedback. In order to applicable to programming education in elementary and secondary surveyed the usability, learning possibility, interest and satisfaction of WithBlock. The results of the survey showed that it can be used for programming education.

Keywords : Block Programming, Text programming, Transference, Programming Education

[†] 정 회 원: 고려대학교 영재교육원 연구교수(교신저자)
^{††} 종신회원: 고려대학교 정보창의교육연구소 연구교수
 논문접수: 2018년 3월 29일, 심사완료: 2018년 7월 3일, 게재확정: 2018년 7월 4일

1. 서론

정보사회에서 능동적으로 컴퓨팅 파워를 활용하기 위해서는 알고리즘을 설계하고 코딩할 수 있는 프로그래밍 능력이 중요하다[1]. 제4차 산업혁명의 도래로 인해 SW분야의 국가경쟁력 확보를 목적으로 초등학교부터 프로그래밍 교육을 조기에 시작하려는 노력을 기울이고 있다[2][3][4]. 교육과정의 핵심적인 목표는 컴퓨팅 사고력의 향상으로 프로그래밍 개념의 이해와 알고리즘 설계 활동을 제시하고 있다[5]. 교육과정의 목표를 달성하기 위해 학습자에게 제공하는 학습환경과 도구로 초중등 수준에서는 블록 프로그래밍 환경을 제시하고, 중고등 수준에서는 텍스트 프로그래밍 환경을 제안하고 있다[3][6].

Scratch로 대표되는 블록 프로그래밍 환경은 추상적인 개념을 구체화된 블록들의 조합된 모습으로 시각화하여 제공할 수 있을 뿐 아니라 프로그래밍한 결과를 즉시 확인할 수 있는 장점이 있다[7]. 프로그래밍 교육에서 블록 프로그래밍 환경은 초중등 학습자의 인지수준에 적합할 뿐 아니라 흥미와 관심을 끌 수 있는 시청각적인 요소를 충분히 제공하고 있어 초중등학생을 대상으로 하는 프로그래밍 교육에서 활용되고 있다[8]. 하지만 블록 프로그래밍한 결과물을 알고리즘 차원에서 평가하기에는 수준이 낮고, 데이터의 흐름을 제어하거나 데이터 구조 등의 데이터 처리기능을 낮은 수준에서 제공하고 있다[9]. 블록 프로그래밍 활동 이후, 정교하고 효율적인 프로그래밍 활동을 수행하기 위해서는 알고리즘 설계, 데이터 처리 등의 추가적인 학습이 요구된다[6][10]. 즉, 블록 프로그래밍 환경을 통해 프로그래밍에 대한 학습자의 흥미와 관심을 불러일으킨 이후, 문제를 해결하기 위한 정교하고 효율적인 알고리즘을 설계하여 프로그램으로 구현할 수 있는 텍스트 프로그래밍 환경으로의 전이가 필요하다.

블록에서 텍스트 프로그래밍 환경으로의 전이는 개인의 역량 차원에서는 컴퓨팅 파워를 본격적으로 활용하기 위해 필요할 뿐 아니라 국가 차원에서 SW분야 인력양성을 위해서도 필수적으로 요구되는 일이다[11][12]. 전세계의 SW교육과정, 정보교육과정을 살펴보면, 초등수준에서는 블록

프로그래밍 도구를 사용하여 프로그래밍 활동을 수행한 이후, 중등수준에서는 텍스트 프로그래밍 언어를 사용하여 프로그래밍 활동을 수행하도록 권장하고 있다[2][3]. 따라서 초등부터 중등을 거쳐 고등에 이르는 SW교육, 정보교육을 완성하기 위해서는 흥미와 관심을 불러일으키는 블록 프로그래밍 환경 이후 알고리즘 설계와 데이터 처리를 다룰 수 있는 텍스트 프로그래밍 환경으로 전이하기 위한 전략과 도구에 관한 연구가 중요하다[6].

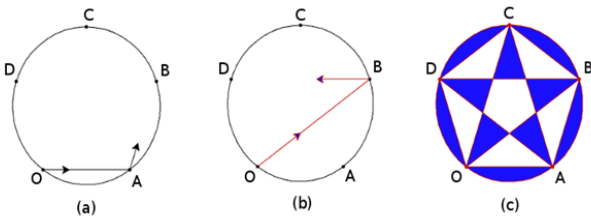
고등수준의 프로그래밍 교육에서는 알고리즘의 정확성과 효율성을 학습하기 위한 목적으로 활용되는 온라인 평가 시스템이 널리 활용되고 있다[13]. 세계적인 수준의 알고리즘 경진대회들은 알고리즘 자동평가 시스템을 이미 사용 중이고, 최근에는 컴퓨터학, 컴퓨터과학을 전공하는 대학에서 학부생 대상의 교육에서 활용하고 있다[14]. 온라인 평가 시스템의 장점은 학습자가 설계한 알고리즘에 대해 자동으로 채점하여 교정적 피드백을 즉시적으로 제공할 수 있어, 학습자가 자기주도적으로 알고리즘 설계와 프로그래밍 학습을 할 수 있다는 점이다[14][15]. 또한 시스템에서 학습자가 풀이한 결과를 종합하여 문제풀이 순위, 알고리즘 효율성, 문제 정답률 등과 같은 경쟁요소를 통해 학습동기와 흥미를 제공할 수 있다[13].

본 연구에서는 블록 프로그래밍 환경에서 온라인 평가 시스템의 장점을 활용하고, 블록에서 텍스트 프로그래밍 환경으로 전이를 지원하고자 하는 목적으로 블록 프로그래밍 환경 기반 온라인 평가 시스템인 WithBlock를 개발하였다[16]. 제안하는 시스템은 하나의 알고리즘 문제를 블록과 텍스트 프로그래밍 환경에서 동시에 풀이하고 결과를 확인할 수 있다. 즉, 알고리즘의 난이도는 동일하지만 블록에서 텍스트로 프로그래밍 방식을 자연스럽게 변경할 수 있어 블록에서 텍스트로 전이를 가능케 하는 환경을 제공할 수 있다. WithBlock의 사용성을 평가하기 위한 목적으로 초중등 학습자를 대상으로 프로그래밍 활동을 수행한 이후, 사용성, 학습 가능성, 흥미와 만족에 대해 설문하였고, 그 결과 개발한 WithBlock이 프로그래밍 교육에서 충분히 활용할 수 있음을 보여주었다.

2. 관련연구

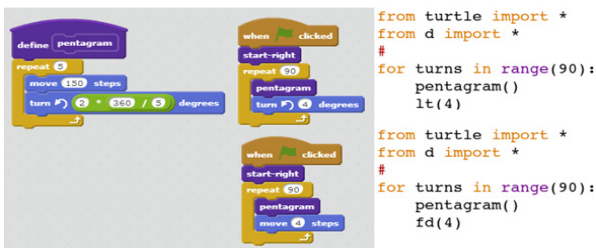
2.1 블록 프로그래밍 환경에서 텍스트 프로그래밍 환경으로의 전이

블록 프로그래밍 환경에서 텍스트 프로그래밍 환경으로 전이하기 위한 연구들을 살펴보면 다음과 같다. Mark와 Dave(2015)는 초중등 학생을 대상으로 도형을 그리는 활동을 Unplugged, Scratch, Python 순으로 전이하는 방안을 제안하였다[10]. 초중등 학생이 시각적으로 쉽게 이해할 수 있는 도형을 그리는 콘텐츠를 바탕으로 첫째, 각도, 선, 원, 사각형, 오각형 등을 그리는 Unplugged 활동을 [그림 1]과 같이 수행하도록 하였다.



[그림 1] 도형을 그리는 Unplugged 활동 예

둘째, Scratch로 프로그래밍하여 결과를 확인하고, 이후 Python turtle을 사용하여 동일한 그림을 그리는 프로그래밍 활동을 [그림 2]와 같이 수행하도록 하였다.

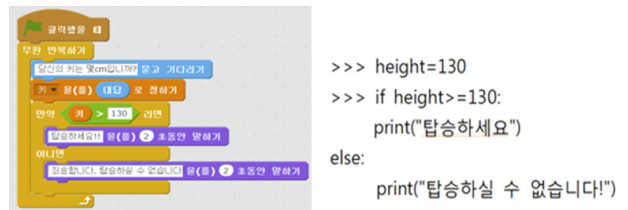


[그림 2] 별을 그리는 Scratch와 Python 예

셋째, Scratch 코드와 Python 코드를 비교하는 활동을 통해 블록에서 텍스트로 전이할 수 있도록 비계(scaffolding) 역할을 수행하는 활동을 제시하였다. 도형을 그리는 콘텐츠를 반복, 함수, 패턴의 프로그래밍 개념에 따라 학습의 단계를 설정하여 반복적으로 Unplugged > Scratch > Python의 순으로 학습하는 구체적인 단계를 제시

하였다.

초등학생을 대상으로 블록 프로그래밍 환경을 통한 선행학습이 후행하는 텍스트 프로그래밍 학습에 도움이 되는지 확인하고자 하는 목적으로 진행된 소미현과 김자미(2016)의 연구는 다음과 같다[11]. 초등학생을 대상으로 Scratch를 사용하여 [그림 3]과 같은 문제인 1부터 10까지 합을 계산, 키에 따라 놀이기구의 탑승여부를 출력하는 프로그램, 나이를 계산하는 프로그램, 자판기 프로그램 등의 알고리즘 문제를 프로그래밍하는 활동을 수행하고, 동일한 문제를 Python으로 프로그래밍하는 활동을 수행한 결과를 분석하였다. 분석 결과, 선행하는 블록 프로그래밍 환경인 Scratch에서 학습한 결과와 후행하는 텍스트 프로그래밍 환경인 Python에서 학습한 결과가 통계적으로 유의한 상관이 있는 것으로 분석되어 블록에서 텍스트로 전이가 이루어진 것으로 확인하였다.



[그림 3] 입력된 값에 따라 출력하는 문제 예

블록 기반 프로그래밍 언어에서 텍스트 기반 프로그래밍 언어로 긍정적인 학습 전이를 위한 전략을 제시한 이소율과 이영준(2018)의 연구를 정리하면 다음과 같다[17]. 첫째, 학습자의 인지구조를 고려하여 블록 프로그래밍과 관련된 선행 지식과 텍스트 프로그래밍과 관련된 새로운 개념들의 자연스러운 연결이 될 수 있도록 선행조직자를 제시할 필요가 있다. 둘째, 블록 프로그래밍 도구와 텍스트 프로그래밍 도구에서 동시에 풀이할 수 있는 문제해결에 초점을 둔 학습 콘텐츠를 제공할 필요가 있다. 셋째, 학습자의 수준과 특성을 고려하여 시각적이며 직관적인 화면을 구성할 필요가 있다.

이상의 연구들을 토대로 도출한 블록에서 텍스트 프로그래밍 환경으로의 전이를 위한 환경의 요구사항은 다음과 같다. 첫째, 블록 프로그래밍 환경에서 학습자가 습득한 도구적 사용성이 텍스

트 프로그래밍 환경으로 전이되기 위한 프로그래밍 도구가 제공될 필요가 있다. 텍스트 프로그래밍 문법과 체계를 고려한 블록 프로그래밍 도구의 개발은 학생이 텍스트 프로그래밍 방식의 문법과 체계로 전이할 수 있도록 도움을 줄 수 있다[18]. 둘째, 초중등 학습자의 수준을 고려한 알고리즘 중심의 문제가 제공될 필요가 있다[13]. 추상적인 알고리즘의 수준은 동일한 상태에서 프로그래밍 방식의 차이를 통해 자연스럽게 블록에서 텍스트 프로그래밍 방식으로 전이되는 비계를 제공할 수 있다. 이를 기반으로 본 연구에서 제시하는 블록에서 텍스트 프로그래밍 환경으로의 전이는 <표 1>과 같다.

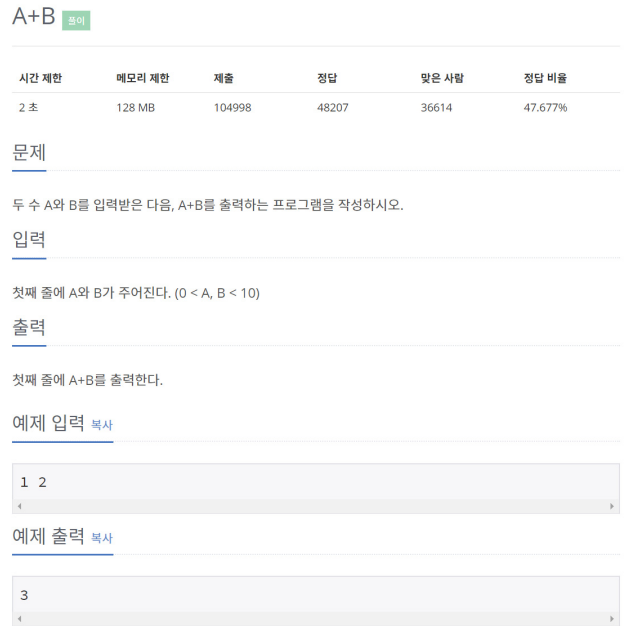
<표 1> 블록에서 텍스트 프로그래밍 환경 전이 전략

대상	초등학생	초중학생 (전이 전략)	고등학생
도구	블록 프로그래밍 도구	텍스트 프로그래밍 방식을 반영한 블록 프로그래밍 도구	텍스트 프로그래밍 도구
문제	흥미과 재미에 기반한 산출물 제작	간단한 알고리즘 문제 (제어구조, 변수, 연산자 등)	복잡한 알고리즘, SW제작 (중첩제어구조, 배열, 함수 등)

2.2 온라인 알고리즘 평가 시스템

온라인 알고리즘 평가 시스템은 문제에서 제한하는 프로그램의 실행 시간과 공간복잡도 내에서 사용자가 제출한 코드로 입력 데이터에 따라 출력이 올바른지 확인하여 정답의 유무를 제공하는 시스템이다. 시스템을 통해 사용자가 정답의 유무를 즉각적으로 확인할 수 있어 자기주도적으로 학습할 수 있을 뿐 아니라 제출한 코드에서 발생한 오류에 대해 즉시적으로 교정적인 피드백을 제공받을 수 있어 알고리즘적 사고와 프로그래밍 능력의 향상을 이끌 수 있다는 장점이 있다[13]. 다양한 텍스트 프로그래밍 언어를 사용하여 프로그래밍할 수 있고, 다른 사용자간 코드의 유사성을 검사하는 등의 다양한 기능을 통해 컴퓨터과학, 컴퓨터학을 배우는 고등교육에서 널리 활용되

고 있다. 국내에서는 Baekjoon Online Judge, code up, jungol, koistudy, withCS, Algospot, 코딩도장 등이 있고, 국외에서는 UVa Online Judge, Codeforces, usaco, coderbyte, topcoder, Project Euler 등 많은 온라인 알고리즘 평가 시스템이 구축되어 활용되고 있는 중이다[14][19].



[그림 4] 온라인 평가 시스템에서 출제 문제 예 (Baekjoon Online Judge)

초중등을 대상으로 하는 정보교육에서 온라인 평가 시스템을 활용한 연구는 다음과 같다. 전통적인 프로그래밍 교육에 비해 온라인 평가 시스템을 활용하는 교육에서 프로그래밍 능력의 향상이 더 높은 것으로 나타났다[15]. 특히, 온라인 평가 시스템에서 제공하는 피드백과 문제풀이 순위 공개를 통한 경쟁적인 요소를 통해 학습자는 보다 적극적으로 프로그래밍 학습에 몰입하는 것으로 나타났다. 또한 2015 개정 교육과정의 고등학교 과학 계열 전문 교과인 ‘정보 과학’ 과목에서 자동평가 시스템의 활용을 제시하고 있어 텍스트 프로그래밍 언어의 문법에서부터 알고리즘 설계까지 폭넓게 활용할 수 있다[13][20].

온라인 평가 시스템에서 사용하는 언어는 Python, JAVA, C 등의 텍스트 프로그래밍 언어로 초중등 수준에서 사용하기 어려울 뿐 아니라 인지적인 부담이 높아 Scratch, 엔트리와 같은

GUI기반의 교육용 프로그래밍 도구들을 활용하고 있다[21]. 하지만 Scratch, 엔트리와 같은 교육용 프로그래밍 도구는 학생의 흥미와 관심을 증진시키기 위해 애니메이션과 같은 시각적인 결과에 집중하여 데이터 중심으로 진행되는 온라인 평가 시스템과 같은 알고리즘 문제를 평가하기에는 부족하다. 이러한 점을 해소하고자 Code.org에서는 초중등학생의 수준에 적합 알고리즘 문제를 제시하고, 블록을 조립하여 작성한 프로그램을 평가한 결과를 피드백과 함께 제공하였다[22]. Kalelioglu(2015)의 연구에서 초등학교 4학년 학생을 대상으로 5주 동안 적용한 결과 충분히 프로그래밍 활동을 수행할 수 있는 것으로 나타났고, 남학생 보다는 여학생의 문제해결 능력이 일부 향상된 것으로 분석되었다. 하지만, 애니메이션을 기반으로 제공되는 문제의 수준이 초등학생 수준을 고려하고 있어 중고등학생이 풀이하기에는 수준이 낮아 데이터 중심의 알고리즘적 사고를 다루기에는 한계가 있다고 할 수 있다.

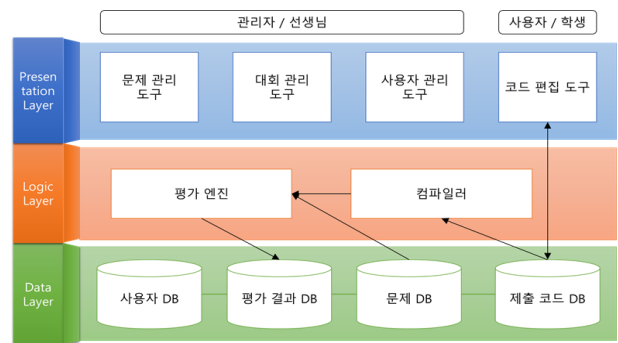


[그림 5] Code.org에서 제공하는 문제 예

이상의 온라인 평가 시스템과 관련된 사례와 연구들을 토대로 할 때, 중고등학생을 대상으로 알고리즘적 사고의 향상을 위한 온라인 평가 시스템의 개발은 다음을 고려할 필요가 있다. 첫째, 알고리즘 문제를 제시하고 평가하여 피드백을 제공하는 방식은 기존의 온라인 평가 시스템이 제공하는 교정적인 피드백과 경쟁의 요소를 포함할 필요가 있다[13][14][15][20]. 둘째, 알고리즘 문제를 해결하기 위해 사용하는 도구는 텍스트 프로그래밍에 익숙하지 않은 학습자의 특성을 고려하여 블록 프로그래밍 환경을 제공할 필요가 있다 [21][22].

3. WithBlock

본 연구에서는 블록형 프로그래밍 환경에서 텍스트형 프로그래밍 환경으로 전이하는 과정을 지원하기 위한 목적으로 블록형 프로그래밍 환경인 WithBlock를 개발하였다[16]. WithBlock은 Scratch와 엔트리와 같이 블록 프로그래밍 활동을 통해 알고리즘의 정확성과 효율성을 자동으로 평가하여 피드백하는 온라인 알고리즘 평가 시스템이다. WithBlock은 프로그래밍 문제, 프로그래밍 활동, 프로그래밍한 결과를 확인하는 단계로 구분된다.



[그림 6] WithBlock의 아키텍처

3.1 프로그래밍 문제

프로그래밍 문제는 문제에 대한 설명과 입력 데이터 예시, 출력 데이터 예시를 제공하는 기존의 온라인 평가 홈페이지들과 동일한 형태로 제공되고 있다[15][19]. 두 숫자를 입력받아 속한 결과를 출력하는 문제의 예시는 [그림 7]과 같고, 문제에서 제공하는 입력에서 2와 3에 따른 출력 결과는 6이다.



[그림 7] 두 수의 곱 문제

No.	난이도	제목	BEST 3	성공률
1000	★	A+B	ilicornewonyang, ksh0405, g1017	52%(867/169)
4249	★	hello_world	ilicornewonyang, g1017, ksh0405	65%(871/133)
4250	★	문자열 출력	ilicornewonyang, g1017, ksh0405	38%(13/34)
4251	★	회차별 점수로	ilicornewonyang, ksh0405, happyabby	38%(8/21)

[그림 8] 알고리즘 문제 목록

WithBlock에서는 다양한 문제 리스트를 제공하고, 학생들이 자유롭게 문제를 풀이할 수 있도록 [그림 8]와 같이 제공하고 있다. 문제 리스트에서 제공하는 정보는 학생이 풀이한 코드의 정확성과 효율성에 따라 1~3순위를 표시하여 학생들 간에 경쟁을 통한 문제해결의 욕구를 자극할 수 있다.

3.2 프로그래밍 활동

WithBlock에서는 문제를 해결하기 위해 블록형 프로그래밍 방식을 제공하고 있다. 학생은 기존의 Scratch와 엔트리와 동일하게 블록을 드래그 앤 드롭하여 연결하고 블록 내의 수치 값을 수정, 입력하거나 항목을 선택하는 형태로 프로그래밍 활동을 수행할 수 있다. 1에서 10까지 더한 값을 출력하는 프로그램을 WithBlock에서 제공하는 블록형 프로그래밍 도구로 프로그래밍한 예시는 [그림 9]과 같다.



[그림 9] 1에서 10까지 더하는 WithBlock 프로그램

작성한 프로그램을 실행하면, 화면에서 시각적으로 에러가 발생한 부분을 표시하거나 결과를 하단의 화면에서 제공하는 등의 피드백을 통해 학생이 즉각적인 교정을 수행할 수 있도록 도움을 준다. 또한 코드를 절차적으로 수행해나가는 단계를 확인 할 수 있도록 블록의 하이라이트 기

능을 통해 어느 부분에서 학생의 코드가 오류를 발생하였는지 보다 구체적이고 직접적으로 확인할 수 있다. WithBlock에서 제공하는 블록 카테고리에는 입출력, 만약, 반복, 조건, 수학, 텍스트, 리스트, 변수, 함수로 제공되는 블록들을 활용하여 텍스트 프로그래밍 언어와 동일한 수준에서 알고리즘 설계와 문제해결이 가능하다.

3.3 프로그래밍 평가

WithBlock은 학생이 블록형 프로그래밍 도구로 주어진 문제를 해결하여 코드를 제출하면, 시스템에서 자동적으로 평가하여 결과를 [그림 10]과 같이 즉각적으로 안내한다. 제공되는 결과는 연산에 소요된 시간, 연산에 사용된 메모리, 코드의 길이, 코드의 평가결과가 제공된다. 코드의 평가결과는 문제의 시간, 메모리 등의 제한조건 내에 해결된 경우 “맞았습니다”란 메시지가 출력되고, 정답이 아닌 경우에는 에러에 따라 “출력형식 에러”, “시간 초과”, “메모리 초과” 등의 메시지가 출력된다.

#	문제번호	아이디	결과	시간	메모리	언어	길이	제출시간
98317	튜토리얼 4.1 문자열 대하기	tmpuser1079014231	맞았습니다	16ms	667M	Python	1652	3월 2일 오후 1시
98316	튜토리얼 1.1 "Hello World"	tmpuser1079014231	맞았습니다	12ms	667M	Python	1613	4월 2일 오후 2시
98315	튜토리얼 1.1 "Hello World"	test01	틀렸습니다	10ms	667M	Python	1747	2018년 2월 4일
98314	튜토리얼 1.1 "Hello World"	onacload	틀렸습니다	12ms	667M	Python	2393	2018년 2월 27일
98313	튜토리얼 1.1 "Hello World"	onacload	틀렸습니다	12ms	667M	Python	2393	2018년 2월 27일
98312	기본 피아리드	onacload	맞았습니다	17ms	667M	Python	2200	2018년 2월 9일
98311	튜토리얼 1.2 숫자 입력받기	엘리원준	맞았습니다	12ms	667M	Python	1681	2018년 2월 8일
98310	튜토리얼 1.1 "Hello World"	엘리원준	맞았습니다	11ms	667M	Python	1615	2018년 2월 8일

[그림 10] 문제해결을 위한 알고리즘 평가결과 목록

4. 연구방법

4.1 연구대상

블록 프로그래밍 환경에서 텍스트 프로그래밍 환경을 연계하기 위해 개발된 프로그래밍 도구인 WithBlock의 사용성과 학습 가능성을 분석하기 위한 목적으로 초등학교 6학년에서 고등학생까지의 학생들을 연구대상으로 선정하였다. 연구에 참여한 학생은 51명으로 성별과 블록 프로그래밍 환경의 경험 여부에 따른 세부적인 내용은 <표 2>와 같다.

<표 2> 연구대상

구분	전체	성별		블록 프로그래밍 경험 여부	
		남	여	경험	무경험
초등학교 (6학년)	12	11	1	12	0
중학교 (1~3학년)	22	19	3	20	2
고등학교 (1~2학년)	17	7	10	2	13
합계	51	37	14	36	15

4.2 연구절차

실험은 개발한 WithBlock의 사용성을 측정할 목적으로 초중등 학생에게 다음과 같은 순서로 적용하였다. WithBlock의 사용법을 20분 동안 따라하기 형태로 학습한 이후, 40분 동안 등록된 3개의 알고리즘 문제를 학생 스스로 해결하도록 하였다. 문제풀이 활동 이후 사용성, 학습 가능성, 흥미와 만족을 설문하였다. 학생이 풀이한 알고리즘 문제는 “분을 초로 바꾸기”, “두 수를 나눈 나머지 값”, “입력 받은 값에서 카운트 다운하기”이었다.

4.3 연구도구

WithBlock의 사용성, 학습 가능성, 흥미와 만족도를 측정하기 위한 설문을 제작하였다. 제작한 설문은 사용성 3문항, 학습 가능성 3문항, 흥미와 만족도 2문항이었다. 연구도구의 신뢰도는 <표 3>과 같이, 사용성이 .908, 학습 가능성이 .870, 흥미가 .913으로 모든 변인이 .70이상의 신뢰도를 보임으로서 신뢰할 만한 것으로 나타났다.

<표 3> 연구도구의 신뢰도

요소	신뢰도(Cronbach 알파)
사용성	.908
학습 가능성	.870
흥미와 만족	.913

5. 결과

5.1 학교급에 따른 차이분석

WithBlock에 대한 사용성, 학습 가능성, 흥미에 대한 설문결과 사용성이 3.86점, 학습 가능성이

4.12점, 흥미와 만족이 4.29점으로 높은 것으로 나타났다.

<표 4> 학교급에 따른 차이분석

요소	전체 (n=51)	초등학교 (n=12)	중학교 (n=22)	고등학교 (n=17)	F (Post-hoc)
사용성	3.86 (1.02)	4.64 (0.69)	3.94 (0.84)	3.24 (1.07)	8.777*** (초,고)
학습 가능성	4.12 (0.94)	4.72 (0.40)	4.18 (0.76)	3.61 (1.14)	6.127** (초,고)
흥미와 만족	4.29 (0.90)	4.83 (0.33)	4.41 (0.73)	3.76 (1.09)	6.501** (초,고)

*** p<.001, ** p<.01, * p<.05

학교급에 따른 분석결과 학교급 간에 사용성, 학습 가능성, 흥미와 만족 모두 통계적으로 차이가 있는 것으로 나타났고, 사후분석에서 모든 변인이 초등학교와 고등학교 간에 차이가 있는 것으로 분석되었다. 초등학교가 모든 변인에서 가장 높은 것으로 나타났고, 중학교, 고등학교로 갈수록 낮아지는 것으로 분석되었다. 프로그래밍과 관련된 활동에서 중고등학생 보다는 초등학생의 인식과 태도, 흥미와 만족 등이 높게 나타난다는 기존연구와 유사한 결과로, 학령이 낮을수록 블록 프로그래밍 활동을 선호하는 것으로 해석할 수 있다[23].

5.2 성별에 따른 차이분석

성별에 따른 분석결과 사용성과 학습 가능성에서는 통계적으로 유의미한 차이가 있는 것으로 나타났고, 흥미와 만족에서는 남학생과 여학생 간에 차이가 나타나지 않았다.

<표 5> 성별에 따른 차이분석

요소	남학생 (n=37)	여학생 (n=14)	t
사용성	4.09(0.96)	3.29(0.99)	2.649*
학습 가능성	4.32(0.85)	3.60(0.97)	2.590*
흥미와 만족	4.43(0.88)	3.93(0.87)	1.835

*** p<.001, ** p<.01, * p<.05

여학생이 남학생에 비해 사용성, 학습 가능성, 흥미와 만족에서 모두 낮은 것으로 분석되었다. 프로그래밍에 대한 인식과 태도를 성별에 따라 분석한 기존의 연구들과 유사한 결과로, 여학생이 남학생에 비해 인식과 태도가 낮게 응답하는 경향이 나타난 것으로 해석할 수 있다[24][25][26].

5.3 사전 블록 프로그래밍 경험유무에 따른 차이분석

사전에 Scratch, 엔트리와 같은 블록 프로그래밍 환경의 경험 유무에 따른 차이분석 결과 통계적으로 유의미한 차이가 있는 것으로 분석되었고, 사전에 블록 프로그래밍 환경을 경험한 학생의 사용성, 학습 가능성, 흥미와 만족이 더 높은 것으로 나타났다.

<표 6> 블록 프로그래밍 경험에 따른 차이분석

요소	경험자 (n=36)	무경험자 (n=15)	t
사용성	4.18(0.81)	3.13(1.13)	3.711**
학습 가능성	4.35(0.69)	3.56(1.21)	2.389*
흥미와 만족	4.53(0.62)	3.73(1.19)	2.445*

** p<.01, * p<.05

활동이 종료된 이후 학생을 인터뷰한 결과, 초등학생과 중학생의 경우에는 Scratch, 엔트리를 배운 경험이 있어 블록 프로그래밍 환경에 익숙할 뿐 아니라 기초적인 알고리즘 문제에 대한 이해가 있었지만, 대부분의 고등학생 경우에는 프로그래밍 활동에 대한 경험이 없을 뿐만 아니라 블록 프로그래밍 환경을 처음 접한 것으로 응답하였다. 이는 초등학생이라도 블록 프로그래밍 환경에 대한 경험이 있으면, 쉽게 블록 프로그래밍 환경 기반의 알고리즘 문제해결이 가능함을 확인하였고, 사전에 블록 프로그래밍 환경에서 프로그래밍 활동의 경험 유무가 후속하는 WithBlock의 알고리즘 문제해결 활동에 큰 영향을 미치는 것으로 해석할 수 있다[11].

6. 결론 및 논의

본 연구는 블록 프로그래밍 환경에서 텍스트 프로그래밍 환경으로 전이하는 과정에서 학습자들에게 도움을 주기 위해 개발된 온라인 평가 시스템인 WithBlock의 프로그래밍 교육에 적용 가능성을 평가하기 위한 목적으로 연구되었다. 연구 결과, 기존의 블록 프로그래밍 환경에 익숙한 학생일수록 WithBlock을 선호하는 것으로 나타났을 뿐 아니라 처음 접하는 학생들에게도 프로그래밍 학습시 도움이 줄 수 있는 것으로 분석되었다. 이

러한 결과는 WithBlock에서 제공하는 블록 프로그래밍 환경, 즉시적이고 시각적인 피드백, 평가 결과에 대한 경쟁적인 요소 등이 종합적으로 나타난 결과라고 해석할 수 있다. 즉, WithBlock은 초등학교 수준에서 블록 프로그래밍 환경을 통해 흥미와 관심에 따라 프로그래밍 기초활동을 체험한 이후, 중고등학교 수준에서 알고리즘 문제를 해결하는 형태로 프로그래밍 교육을 진행하는데 충분히 도움을 줄 수 있는 환경이라 할 수 있다.

2015개정교육과정의 적용과 SW선도·연구학교의 증가에 따라 학교현장에서는 가르치는 교육내용 뿐 아니라 학생의 프로그래밍 능력에 대한 평가방법과 기준 또한 중요한 이슈가 되어가고 있다[6][27]. WithBlock은 블록 프로그래밍 환경을 경험한 학생들이라면 쉽게 자기주도적으로 프로그래밍할 수 있는 환경에서 알고리즘 문제를 풀이하는 능력을 정량적으로 평가할 수 있다는 점에서 초중등을 대상으로 하는 프로그래밍 교육에서 활용가치가 높다고 할 수 있다[13]. 추가적으로 고등학교 수준에서는 이미 자동평가를 활용하는 것을 권장하고 있어 교육현장에 즉시적인 적용 또한 어렵지 않을 것으로 판단할 수 있다[20].

향후 연구로는 블록에서 텍스트로의 전이를 학습자의 성취도 수준에서 확인할 수 있는 연구가 진행될 필요가 있고, 초중등 학습자들의 수준을 고려한 온라인 자동평가 문항들의 추가적인 개발이 필요할 것으로 보인다.

참고 문헌

- [1] 교육부(2016). 소프트웨어교육 활성화 기본계획
- [2] Department for Education(2013). National Curriculum in England : frame work for key stage 1 to 4.
- [3] K - 12 Computer Science Framework Steering Committee(2016). K-12 Computer Science Framework.
- [4] Iirs Z. B., Bazil P. Dvir L.(2013). A Nationwide Exam as a Tool for Improving a New Curriculum, *Proceedings of the 18th ACM conference on Innovation and*

- technology in computer science education*. 267-272.
- [5] 교육부(2015). 교육부 고시 제2015-74호 [별책10]. 실과(기술·가정)/정보과 교육과정.
- [6] 신승기, 배영권(2015). 학교 급별 연계성 있는 소프트웨어 교육 체제 설계를 위한 연구. **정보교육학회논문지**. 19(4). 533-544.
- [7] Mitchel Resnick et al.(2009). Scratch: Programming for All. *Communications of the ACM*, 52(11). 60-67.
- [8] Computer Science For All(2016). <https://obamawhitehouse.archives.gov/blog/2016/01/30/computer-science-all>
- [9] 손원성(2016). 블록기반 프로그래밍 코드의 수준 및 취약수준 측정방안. **정보교육학회논문지**. 20(3). 293-302.
- [10] Mark, D., Dave, W.(2015). Scratch : A Way to Logo and Python, *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*. 191-196.
- [11] 소미현, 김자미(2016). 블록형 프로그래밍 학습에서 텍스트형 프로그래밍 학습으로의 전이. **컴퓨터교육학회논문지**. 19(6). 55-68.
- [12] 나성현(2013). SW 인력양성을 위한 정책제언. 정보통신정책연구원. Premium Report 13-11.
- [13] 장원영, 김성식(2017). 정보 교육에서 프로그래밍(알고리즘) 자동평가 시스템의 활용 가능성에 대한 고찰. **컴퓨터교육학회논문지**. 20(1), 13-26.
- [14] Andy K., Andrew L., Brenda C.(2001). Online Judge. *Computer & Education*. 36. 299-315.
- [15] 장원영, 김성식(2014). 알고리즘 자동평가 시스템의 개발 및 적용 : 프로그래밍 학습 효과 분석. **컴퓨터교육학회논문지**. 17(4). 26-34.
- [16] <http://withblock.com>
- [17] 이소율, 이영준(2018). 블록 기반 프로그래밍 언어에서 텍스트 기반 프로그래밍 언어로의 학습 전이를 위한 프로그램 설계 방안. **한국컴퓨터교육학회 학술대회발표대회논문지**. 22(1). 29-31.
- [18] C.C. Brown, M. Jens, B. Anthony, W. David(2016). Future Directions of Block-based Programming, *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*. 315-316.
- [19] Gui P. W., Shu Y. C., X. Y. and Rui F.(2016). OJPOT: online judge & practice oriented teaching idea in programming courses. *European Journal of Engineering Education*. 41(3). 304-319.
- [20] 교육부(2015). 교육부 고시 제2015-74호 [별책 20]. 과학계열 전문교과 교육과정.
- [21] Maloney, J., Peppler, K., Kafar, Y., Resnick, M., & Rusk, N. (2008). Programming by choice: Urban youth learning programming with scratch. *Proceedings of the 39th ACM Technical Symposium on Computer Science Education*. 367 - 37.
- [22] Kalelioglu, F. (2015). A new way of teaching programming skills to K-12 Code.org, *Computer in Human Behavior*. 52. 200-210.
- [23] 심재권, 권대용, 이원규(2014). 초중등 프로그래밍 교육을 위한 방과후 교육과정안 설계. **한국컴퓨터교육학회 학술발표대회논문집**. 18(2). 13-16.
- [24] 심재권, 김현철, 이원규(2016). 2015 개정 정보 교육과정에 따른 피지컬 컴퓨팅을 활용한 정보교육에서 성별에 따른 태도와 성취도 차이분석. **컴퓨터교육학회논문지**. 19(4). 1-9.
- [25] 유병건, 김자미, 이원규(2012). 성별에 따른 프로그래밍 성취도와 문제해결과정의 관계 분석. **컴퓨터교육학회논문지**. 15(6). 1-10.
- [26] 배영권(2007). 성별의 차이를 고려한 로봇프로그래밍 교수전략에 관한 연구. **컴퓨터교육학회논문지**. 10(4). 27-38.
- [27] 신수범(2017). 블룸 분류 기반 초중등 프로그래밍교육의 평가 기준 탐색. **정보교육학회논문지**. 21(5). 547-555.



심재권

2007 경인교육대학교
컴퓨터교육과(교육학사)
2012 고려대학교
컴퓨터교육학과(이학석사)

2017 고려대학교 컴퓨터교육학과(이학박사)
2017~현재 고려대학교 영재교육원 연구교수
관심분야: 컴퓨터교육, 프로그래밍 교육
E-Mail: jaekwoun.shim@gmail.com



채정민

2003 고려대학교
컴퓨터교육과(이학사)
2005 고려대학교
컴퓨터교육과(이학석사)

2012 고려대학교 컴퓨터교육학과(이학박사)
2012~현재 고려대학교 정보창의교육연구소
연구교수
관심분야: 텍스트마이닝, 컴퓨터교육
E-Mail: onacloud@gmail.com