

알고리즘적 사고 향상을 위한 파스스 퍼즐 콘텐츠 개발 및 적용*

조현철¹, 이시훈²

◀◀ 요약 ▶▶

본 연구의 목적은 알고리즘적 사고 향상을 위한 퍼즐게임 콘텐츠를 제안하는 것이다. 2015개정교육과정에서는 SW교육을 통해 알고리즘에 대한 이해를 학습할 것을 제시하고 있다. 퍼즐게임은 게임의 형태를 띠고 있어 학습자들의 흥미와 참여를 유도하기에 적합하며, 기초적인 알고리즘을 이해하는데 효과적이다.

이에 본 연구에서는 파스스 퍼즐을 기반으로 한 콘텐츠를 개발하고 적용하였다. 개발한 퍼즐 콘텐츠는 Drag & Drop 방식으로 작동하여 태블릿에서도 사용이 용이하도록 하였다. 콘텐츠는 텍스트형 프로그래밍의 내용으로 구성하되 지나치게 문법 위주의 교육이 되지 않도록 교사와 함께 알고리즘의 구조를 살펴보도록 수업을 구성하였으며, 퍼즐게임 자체적으로도 학생들이 해당 코드의 의미를 확인할 수 있도록 했다.

개발한 프로그램을 SW교육 경험이 충분히 많은 초등학생들에게 적용한 결과 수업에 높은 흥미도와 알고리즘의 이해를 나타내 퍼즐게임이 텍스트형 프로그래밍 알고리즘 이해에 효과적으로 활용될 수 있는 것으로 나타났다. 반면 텍스트형 프로그래밍의 내용에 대해서는 어려움을 느낀 것으로 나타나 차후 중학생에게 확대 적용하여 후속 연구를 진행할 필요가 있다.

주제어 : 알고리즘, 텍스트형 프로그래밍, 퍼즐게임, 게이미피케이션, 파스스 퍼즐, SW교육

1. 청주신흥고등학교 교사, pran@naver.com (주저자)
2. 갈원초등학교 교사, shoon1984@gmail.com (교신저자)

I. 서론

4차 산업혁명으로 인공지능, 빅데이터, 사물인터넷 등의 분야가 빠르게 발전하면서 정보 활용 능력, 컴퓨터 활용 문제해결 능력, 디지털 문해력 등의 역량 함양이 교육의 중요한 과제로 떠오르고 있다. 2015개정 교육과정에서는 변화한 새로운 시대를 위한 창의·융합적 미래 인재 양성을 위해 SW교육을 초등학교 5, 6학년군에 포함하여 2019년부터 실시되고 있다. SW교육은 단순히 프로그래밍 언어를 익히고 기능을 수행하도록 하는 것을 목표로 하지 않는다. SW교육은 문제해결을 위한 교육으로 알고리즘적 사고를 계발하여 다양한 문제해결 과정을 경험함으로써 컴퓨팅 사고력을 향상시키는 것이 중요하다(최정원 외, 2015).

컴퓨팅 사고력 함양을 위해 다양한 교육용 프로그래밍 언어들이 개발되었고 많은 연구가 진행되고 있음에도 아직 학교 현장에서는 올바른 알고리즘 교육이 부족한 실정이다. 여전히 많은 학생들이 문제를 해결하기 위한 알고리즘적 사고 능력의 부족으로 프로그래밍에 어려움을 느끼고 있다(김용천 외, 2013). 컴퓨팅 사고력 함양을 위해서는 알고리즘을 이해하고 이를 설계해보는 활동이 필요하다. 그러나 구체적 조작기에 해당하는 초등학생들에게 추상적 개념에 대한 이해를 요구하는 알고리즘 교육은 어려울 수 밖에 없다(김은지 외, 2018). 학습자의 발달 단계를 넘어서는 교육활동은 학습자들의 흥미와 동기를 유발하지 못하고 학습에 대한 피로감과 거부감을 일으킨다.

이러한 문제를 해결하기 위한 방법으로 알고리즘 교육을 위한 게이미피케이션이 주목받고 있다. 다양한 교육 분야에서 학습들의 흥미와 몰입을 높이고 교육적 효과를 향상시키기 위하여 다양한 게임과 이를 활용한 교수학습 방법이 시도되고 있다. 특히 논리학습과 알고리즘 사고 교육을 위한 방법으로 퍼즐게임에 대한 많은 연구가 이루어지고 있다. 퍼즐게임은 문제해결과정에서 학습자들이 자연스럽게 알고리즘적 사고를 체험하며 사고력을 향상시키는 데 효과적이다. 또한 특정한 프로그래밍 언어의 문법이나 기능을 익히지 않아도 되기 때문에 초등학생들을 위한 알고리즘적 사고 함양을 위한 프로그래밍 교육에 적합하다는 장점이 있다(박주연 외, 2021).

본 연구는 SW 교육 현장에서 알고리즘적 사고 향상을 위한 퍼즐게임을 개발하여 수업에 적용하여 그 결과를 분석해보고, 알고리즘 교육에 파스너 퍼즐 형태의 게이미피케이션을 적용한 방법에 대해 제안하고자 한다.

II. 이론적 배경

1. 알고리즘 교육

2019년도부터 실시된 소프트웨어(SW)교육은 컴퓨팅 사고력(Computational Thinking) 함양을 목표로 한다. 이를 위해 언플러그드, 교육용 프로그래밍 언어 활용 교육, 피지컬 컴퓨팅 학습 등 다양한 방법으로 문제를 해결하는 과정을 학습하게 된다. 컴퓨터가 이해할 수 있도록 문제를 해결하는 일련의 절차를 표현한 것을 알고리즘이라 하며(Adamuthe, A. C. 외, 2020), 알고리즘적 사고는 컴퓨터로 문제를 해결하는데 필요한 문제해결 절차를 구성해내는 능력으로 정의된다(Jenkins, 2002). 학생들은 SW교육을 통해 효과적인 알고리즘을 구성하는 방법을 학습하게 되는데, 일반적으로 알고리즘 설계를 먼저 수행한 후 프로그래밍을 학습하는 단계가 제시된다(김은지, 이태욱, 2017). 실제로 초등학생을 위한 교육용 프로그래밍 도구를 활용하는 알고리즘 사고 중심 교육 결과(이용배, 박지은(2012), 김용천 외(2013)) 프로그래밍 도구를 활용한 교육 전 알고리즘을 설계하는 활동이 타당한 것으로 나타났다(심재권, 권대용, 2019). 세계 각국에서는 알고리즘의 교육을 위해 다양한 요소를 교육과정에서 제시하고 있으며 그 내용은 <표 1>과 같다.

<표 1> 알고리즘적 사고 구성요소(심재권, 2019)

구분	권대용 (2008)	미국 (2011)	인도 (2013)	영국 (2014)	최정원 (2014)
순차	○	○	○	○	○
선택	○	○	○	○	○
반복	○	○	○	○	○
함수	○	○			
데이터			○		
변수		○		○	
입출력				○	
이벤트					○

알고리즘 교육에서는 학습자가 문제를 해결하는 과정에서 어떤 절차와 방법을 거치게 되는지가 중요하므로, 컴퓨팅 사고력을 바탕으로 학습자가 경험하게 되는 방법과 절차를 구체적인 프

로그래밍 언어로 작성하는 형태로 수업이 이루어져야 한다(김은정, 2019, Durak, H. Y., 2019). 이러한 과정에서 학습자는 자신만의 방법으로 알고리즘을 설계하고 개발하는 경험을 통해 알고리즘에 대한 이해가 깊어지고 문제해결 능력도 향상 될 수 있다. 한편, 초등학생들은 알고리즘을 구성할 때 보이는 물체의 움직임이나 구체적인 모양 등 시각적인 부분을 중심으로 알고리즘 패턴을 구성하는 경향이 있으며, 시각적으로 드러나지 않는 부분에 대해서는 미흡한 부분이 나타났는데(김정량, 2020), 이는 알고리즘 교육에서 시각적인 부분의 중요성을 의미한다.

2. 텍스트형 프로그래밍 언어 교육 실태

학교에서 주로 활용되는 교육용 프로그래밍 언어는 블록형 프로그래밍 언어와 텍스트형 프로그래밍 언어로 구분할 수 있다. 일반적으로 텍스트형 프로그래밍 교육에서는 C, Java, Python 등이 사용되는데, 기본 문법이나 구조가 복잡하여 이를 익히기 위해서는 상당한 시간이 필요하다. 이 언어들을 활용하기 위한 프로그래밍 언어 수업 설계를 살펴보면 문법과 기능 위주의 내용으로 구성되어 있어 실질적인 알고리즘 교육이 실현되기 어렵다. 따라서 프로그래밍에 대한 구조화와 이해가 아직 충분히 이루어지지 못한 초등학생에게 텍스트형 프로그래밍 언어를 그대로 교육용으로 사용하기에는 무리가 있다. 또한 기존의 텍스트형 프로그래밍 언어는 디버깅에 많은 시간을 소모하게 되어 초보자들은 실제 문제를 해결하는데 알고리즘을 적용하기도 전에 흥미를 잃고 학습으로부터 멀어지게 되는 요인이 되기도 한다(최정원 외, 2014).

이러한 이유로 현재 우리나라의 초등교육에서는 주로 블록형 프로그래밍 언어가 사용된다. 초보자들이 이해하고 사용하기 쉽기 때문이다(Brad Myers, 1990). 실제로 우리나라의 SW교육 선도학교 실태조사 결과 스크래치(29.1%)와 엔트리(84.4%)의 블록형 프로그래밍 언어 비중이 압도적으로 큰 것으로 나타났다(마대성, 2021).

그러나 블록형 프로그래밍 언어는 캐릭터를 조작하는 것에 특화되어 있다는 점이 한계로 지적되기도 한다(Tuyano Syoda, 2016). 반면 텍스트형 프로그래밍 언어는 학습에 많은 시간이 필요하지만, 블록형 프로그래밍 언어에 비해서 컴퓨팅 사고력을 향상시키는데 도움이 된다는 연구 결과가 있다. 이러한 한계를 극복하기 위해 초등학생을 위한 텍스트형 프로그래밍 언어를 활용한 교육 방안이 연구되기도 한다(문미예 외, 2018). 영국의 Computing 교육과정에서는 11~14세에 해당하는 Key stage 단계에서 2가지 이상의 프로그래밍 언어를 사용하며, 최소한 하나의 언어는 반드시 텍스트형 프로그래밍 언어를 사용하도록 하고 있다(정구원, 2015). 이러한 점들을 종합하여 볼 때 학습자의 수준을 고려한 텍스트형 프로그래밍 언어 활용 교육방법이 종합적

으로 연구될 필요성이 있다.

3. 게이미피케이션

게이미피케이션(Gamification)이란 게임이 갖는 흥미나 속성 등을 바탕으로 다양한 분야에서 사용자의 인식이나 행동을 변화시키는 방법을 말한다. 교육분야에서는 게이미피케이션의 적용이 학습자의 동기유발이나 몰입에 효과적(정의준, 2013)이라는 기대감에 점점 관심이 높아지고 있다(한안나, 2018). 특히 기존의 게임기반 학습이나 교육용 게임은 특정한 교육과정 전체를 게임처럼 구성하는 경향이 강했다면, 게이미피케이션 교육은 일부 게임 메커니즘을 활용하여 교육활동에 특정한 부분에 적용하여 다양한 교육적 효과를 높이기 위한 접근법이 강조되고 있다(Dicheva, D. 외, 2015). 다양한 방법 중 퍼즐 게임은 예전부터 다양한 형태로 교육에 적용되고 있는 방법 중 하나이며, 문제해결과정에서 알고리즘적 사고를 학습하게 된다는 장점이 있다(최정원 외, 2015).

장정아(2009)의 연구에서는 초등 정보교육에서 정보과학적 사고 향상을 위한 새로운 교수 학습도구로 퍼즐을 제안한 바 있으며, 퍼즐의 문제해결과정과 프로그래밍에서 일어나는 사고 과정을 비교함으로써 정보 교육의 형태로서 퍼즐의 가능성을 제시하였다. 박주연(2021)의 연구에서는 퍼즐게임 활용 교육에 있어 기초 프로그래밍의 원리와 알고리즘 탐구에 적용할 수 있음을 제안하고 있다. 조현철(2007)은 퍼즐게임 형태의 Java언어 학습을 위한 코드자석 게임을 만들어 텍스트 프로그래밍 언어를 블록 형태의 게임으로 이해시키고자 시도하기도 하였다.

4. 파슨스 퍼즐(Parsons Puzzle)

파슨스 퍼즐은 프로그래밍을 학습하는데 유용한 방법으로 알려진 퍼즐형 문제해결 방법이다(Parsons, Haden, 2006). 파슨스 퍼즐에서는 프로그램을 완성하기 위한 코드들이 퍼즐조각처럼 랜덤하게 제시되며 학습자는 이들을 Drag & Drop 방식으로 올바른 순서로 조립하게 된다. 파슨스 퍼즐은 코드를 실제로 작성하고 오류를 수정하는 활동보다 시간은 짧으면서도 학습 성과는 동일한 것으로 나타났으며(Ericson 외, 2017), 학생들은 비교적 인지 부하가 낮은 객관식 문제 풀이나 인지 부하가 높은 코드 작성보다 파슨스 퍼즐을 푸는 것을 더 선호하는 것으로 나타났다(Ericson 외, 2015). 외국에서는 파슨스 퍼즐을 온라인 상에서 교육에 활용할 수 있도록 다양한 연구들이 진행되고 있다. Kunmar(2018)는 프로그래밍 입문 과정에서 학습자들이 파슨

스 퍼즐을 쉽게 사용할 수 있도록 웹 어플리케이션 도구인 Epplets를 개발하였으며, Karavirta(2012) 등은 퍼스 퍼즐을 활용한 모바일 학습 어플리케이션을 제안하기도 하였다. 이러한 프로그램들은 자동화된 피드백을 제공할 수 있다는 점이 큰 특징이며, 특히 Epplets는 C++, Java, C#뿐 아니라 python으로 등으로 확장할 수도 있으며, 학습자들의 학습데이터와 패턴을 누적하여 프로그램을 발전시킬 수 있다는 점에 큰 장점을 가지고 있다.

본 연구에서는 텍스트형 프로그래밍 언어의 알고리즘을 학습하는 활동을 구성하되 학습자의 수준에 근거하여, 블록형 프로그래밍 언어로부터 텍스트 프로그래밍 언어로의 효과적인 교육 전이를 위한 퍼스 퍼즐 기반의 교육방법을 제시하고자 한다.

Ⅲ. 학습 프로그램 개발

1. 콘텐츠 구성

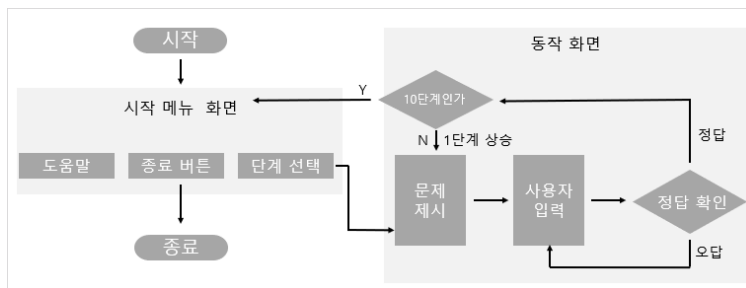
학습 콘텐츠는 Java의 알고리즘 구조 학습 위한 10개의 문제로 구성되어 있다. 학습내용은 Java의 문법 구조를 정확하게 익히는 것이 아니라, Java와 같은 텍스트 프로그래밍 언어가 작동되는 일련의 과정을 학습자가 고민하고 조작해봄으로써 텍스트 프로그래밍의 구조와 알고리즘을 이해할 수 있도록 하였다. 프로그램을 실행하면 학습자는 제시되는 10단계 중 하나의 단계를 선택하게 된다. 그러면 선택된 단계에서의 목표인 출력결과와 맞추어야 할 문제, 즉 코드 조각들이 나타난다. 학습자는 제시된 출력결과를 도출하기 위해 필요한 코드 조각들을 순서 있게 재배치한다. 만일 모든 코드 조각들이 순서있게 재배치되면 현 단계가 종료되고 다음 단계의 문제로 진행된다. 문제에 대한 예시는 <표 2>와 같다. 왼쪽은 학생들에게 처음 제시되는 코드 배치이며, 오른쪽은 올바르게 정렬한 코드이다.

〈표 2〉 예시 문항

목표 : a-b c-d	
if (x==1) { System.out.print("d"); x=x-1; }}}	class Shuffle1 { public static void main(String[] args){
if (x==2) { System.out.print("b c"); }	int x = 3;
class Shuffle1 { public static void main(String[] args){	while(x>0){
int x = 3;	if(x>2){ System.out.print("a"); }
x = x-1; System.out.print("-");	x = x-1; System.out.print("-");
while(x>0){	f (x==2) { System.out.print("b c"); }
if(x>2){ System.out.print("a"); }	if (x==1) { System.out.print("d"); x=x-1; }}}

2. 프로그램 설계

프로그램은 학생들의 프로그램 친숙도를 고려하여 엔트리로 개발하였다. 개발한 프로그램의 개념은 <그림 1>과 같다. 프로그램은 시작 메뉴 화면과 동작화면으로 구성되어 있다. 시작 메뉴 화면에서는 학습 단계 선택, 도움말, 종료 등을 할 수 있으며, 단계 선택 시 동작 화면으로 전환되어 학습 활동을 시작하게 된다. 동작화면으로 전환되면 단계에 따른 문제가 주어지고, 이를 사용자가 조작하는 활동을 통해 학습에 참여하게 된다. 정답 여부에 따라 다음 단계가 진행되거나 정답이 될 때까지 수정하게 된다. 만약 최종 단계까지 완료한 경우 시작 메뉴 화면으로 다시 전환되며 종료버튼을 통해 프로그램을 종료할 수 있다.



[그림 1] 프로그램의 구조

퍼즐게임 작성을 위한 코딩 중 핵심 부분 중 일부에 대한 내용은 <그림 2>와 같다. 블록이 정답 칸의 중심 좌표에 근접하면 해당 정답 칸으로 자동으로 위치하도록 코드를 작성하여 자석 같은 효과를 주었다. 이때 블록의 각각의 위치를 변수로 처리하였고, 이 변수들의 조합으로 최종 정답 여부를 확인할 수 있도록 하였다. 예를 들어 마지막 과제는 7개의 블록 자석을 조립하게 되어있으므로 각각의 블록들은 조립된 위치에 따라 1에서 7까지 중 하나의 값을 갖게 되며 그 순서가 정답과 올바르게 배치되어 있다면 정답 메시지가 출력된다. 정답을 확인하는 코드는 <그림 3>과 같다.



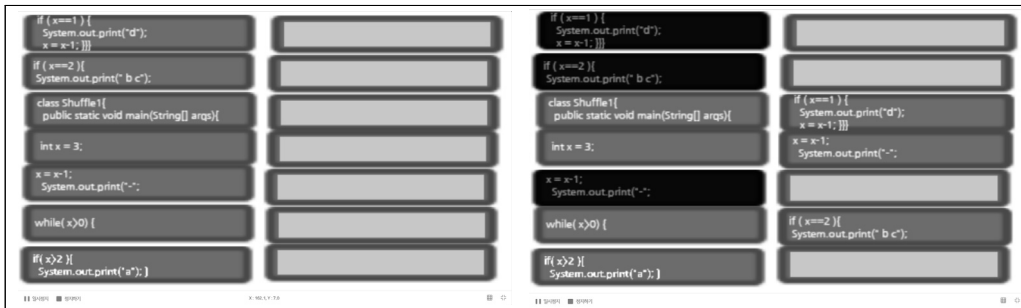
[그림 2] 퍼즐게임 코드



[그림 3] 정답 코드

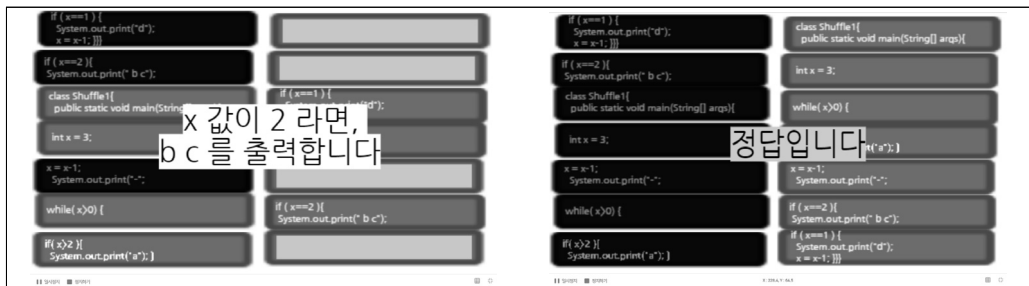
3. 동작 화면

[그림 4]은 프로그램의 첫 번째 단계의 동작 화면을 나타낸 것으로 학생들은 주어진 출력물을 올바르게 출력할 수 있도록 좌측에 제시된 코드들을 Drag & Drop 으로 우측 정답칸에 올바른 순서대로 배치하면 된다. 배치를 잘못하더라도 정답을 맞출 때까지 자유롭게 다시 옮길 수 있다.



[그림 4 동작 화면 1

텍스트 프로그래밍 언어에 어려움을 느끼는 학생들을 위해 조립을 완성한 후 남아있는 좌측의 코드 블록의 잔상을 두 번 누르면 해당 코드의 설명이 [그림 5]와 같이 팝업되도록 하였다. 학생들은 Java의 정확한 문법을 알지 못하더라도 코드의 형태와 팝업된 설명을 통해 코드를 구조적으로 익힐 수 있으며, 해당 코드들을 활용하여 프로그램이 제대로 동작하도록 알고리즘을 설계할 수 있다. 모든 블록의 배치가 올바르게 정렬되면 ‘정답입니다’라는 메시지가 출력되고 다음 문제로 넘어간다.

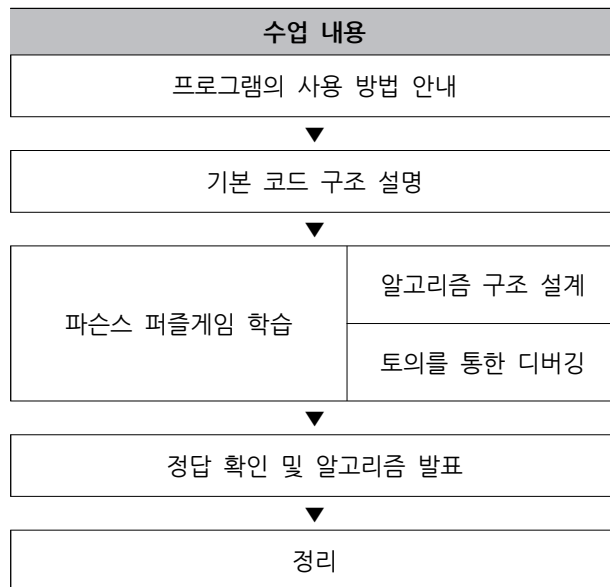


[그림 5] 동작화면 2

IV. 연구의 방법

본 연구는 개발한 학습용 콘텐츠가 학습자의 알고리즘적 사고에 미치는 영향에 대해 연구하고자 하였다. 충북에 위치한 G초등학교의 학생 13명을 대상으로 하였다. 프로그래밍에 대한 경험이 적은 경우 텍스트 프로그래밍 언어만을 보고 학습에 흥미를 잃을 수 있다고 판단하여, 블록 코딩과 다양한 알고리즘 문제해결과정을 많이 경험해본 AI동아리 6학년 학생들에게 적용하였다. 수업에 참여한 학생들은 모두 SW교육 경험이 2년 이상 되는 학생들로 모두 블록형 프로그래밍 언어에 대해 충분한 경험을 가지고 있다. 그러나 Java와 같은 텍스트형 프로그래밍의 경험은 없는 상태이다. 따라서 Java의 문법 자체에 대해 설명하기 보다는 교사와 함께 각각의 코드가 갖는 구조적, 내용적인 의미에 대해 먼저 탐색한 후 알고리즘을 설계하는 콘텐츠 학습을 수행하도록 하였다. 실행한 수업은 <표 3>과 같은 순서로 진행하였다.

〈표 3〉 수업 진행



기본적인 설명 후 학생들이 자유롭게 코드 간의 구조적 관계에 대한 토의를 통해 올바른 알고리즘 구조를 설계하도록 진행하였다. 수업은 태블릿pc를 사용해서 진행하였으며, Drag & Drop

을 편하게 할 수 있도록 터치펜을 제공하였다.



[그림 6] 토의를 통한 알고리즘 구조 설계 및 디버깅

수업을 마친 후 수업과 콘텐츠에 대한 학생들의 만족도 조사를 실시하였다. 수업에 대한 재미, 수업의 수준, 알고리즘 구조에 대한 이해의 3가지 분야에 대해 5점척도 측정하였다. 검사지 내용은 <표 4>와 같다.

<표 4> 만족도 설문 문항

범주	세부	문항
수업에 대한 재미	집중도	수업에 얼마나 집중에서 참여하였습니까?
	흥미도	수업은 얼마나 재미있었습니까?
	만족도	오늘 한 수업이 얼마나 만족스럽습니까?
수업의 수준	난이도	수업의 난이도는 어떻다고 생각합니까?
	학습 내용	오늘 학습한 내용을 잘 이해할 수 있었습니까?
	적용 가능성	같은 결과를 출력하기 위해 오늘 학습한 내용으로 텍스트 프로그래밍에 적용할 수 있습니까?
알고리즘 이해	구조	알고리즘의 구조를 잘 이해할 수 있었습니까?
	설명 가능성	자신이 조립한 코드의 의미를 친구에게 설명할 수 있습니까?
	과제 달성도	프로그램에 대한 이해를 바탕으로 과제를 달성했다고 생각합니까?

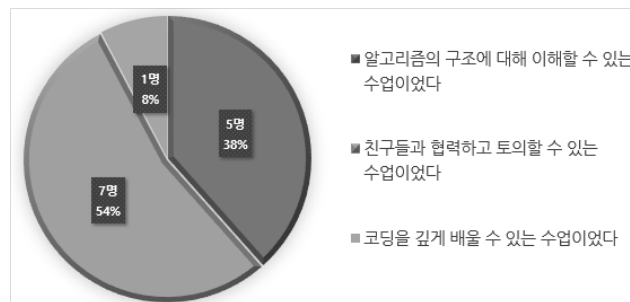
V. 연구의 결과

프로그램을 활용하는 수업을 마친 후 만족도 조사를 실시하였다. 정규성을 확인하기 위하여 정규분포를 따른다는 귀무가설에 대해 Shapiro-Wilk test 결과 수업 재미(유의확률 .000), 수업의 수준(유의확률 .000), 알고리즘의 이해(유의확률 .002)로 $p < .05$ 로 모두 기각되어 정규분포를 따르지 않는 것으로 확인되었다. 이에 따라 모집단의 평균을 중앙값인 3으로 가정하여 Wilcoxon signed rank test를 실시한 결과는 <표 5>와 같다. 그 결과 프로그램의 재미와 알고리즘의 이해 영역에 대해서는 통계적으로 유의한 결과가 나타나 개발한 프로그램이 학생들에게 흥미로웠고 알고리즘을 이해하는데 도움을 줄 수 있었다는 결론을 도출할 수 있었다.

<표 5> Wilcoxon signed rank test결과

범주	M	SD	V	p-value
흥미	4.308	1.251	59.5	0.005
수준	2.231	0.725	5.5	0.997
알고리즘 이해	4.154	0.555	78	0.000

학습자들이 일반적인 SW수업과 비교했을 때 이번수업에서 효과적이었던 측면은 <그림 7>처럼 나타났다. 수업에 참여한 학생들은 응답자의 절반 이상(53.8%)이 친구들과 함께 토론하고 협의하며 문제를 해결할 수 있는 수업이었다고 생각했으며, 본 수업을 통해 알고리즘의 구조에 대해 배울 수 있는 수업이었다고 생각하는 학생도 38.%에 달하였다.



[그림 7] 수업의 효과

학습을 진행한 결과 13명의 학생 중 11명(84.6%)이 과제를 끝까지 완수하였다. 과제를 해결한 학생 중 1명은 정답의 구조에 대해 설명하지 못하였으며, 우연히 배치를 반복하다 보니 풀어

내었다. 그 외에 과제를 완수한 10명의 학생은 알고리즘의 작동 원리를 정교하진 않지만 자신만의 언어로 설명할 수 있었다.

V. 결론 및 제언

본 연구에서는 알고리즘적 사고 향상을 위한 학습용 콘텐츠를 개발하여 적용하였으며, 연구 결과는 다음과 같다.

첫째, 퍼즐게임을 통해 텍스트 프로그래밍 문법구조에 대한 지식이 없거나 어려움을 느끼는 학생들도 흥미를 갖고 수업에 적극적으로 참여할 수 있었다. 이는 학생들이 본 콘텐츠를 학습용이라고 생각하기보다는 게임으로 받아들였기 때문으로 분석된다.

둘째, 프로그래밍 경험이 다소 부족한 학생들도 친구들과 함께 문제 해결 과정을 설계하고 토의하는 과정을 통해 학습에 참여할 수 있었다. 특히 개발한 프로그램은 학습용 태블릿 pc에서도 작동하기 때문에 데스크톱으로 수업을 할 때보다 효과적인 협업이 가능했다.

셋째, 개발한 퍼즐 게임을 통해 학습자들은 알고리즘에 대해 이해할 수 있었다. 설문 결과 내용 수준이 어렵다고 느낀 학생들이 많긴 했으나 알고리즘의 일반적인 구조를 이해하기에는 충분했다고 판단된다.

본 연구에서 개발한 게임을 발전시켜 적용하기 위해서는 다음과 같은 후속 연구가 필요하다. 먼저 본 연구를 통해 만족도 조사를 실시하였으나 실질적인 효과성 분석을 위해 향후 필요한 요소들을 추가로 제시하고 검증된 평가도구를 사용한 분석이 수반되어야 한다. 또한 이번 연구에서는 참여한 학생이 너무 적고 SW교육의 경험이 충분히 많은 학생들을 대상으로 하여 특수한 집단에 대한 교육으로 이루어졌다는 점에 한계가 있다. 따라서 보다 보편적인 교육 적용을 위해 적용 대상의 범위를 늘려 그 효과를 살펴볼 필요가 있다. 다음으로 초등학교에서는 블록형 프로그래밍 교육을 주로 하는 만큼 블록형 프로그래밍 교육과 텍스트형 프로그래밍 교육에 전환기에 있는 중등학생을 대상으로 적용하는 연구가 수행될 필요가 있다. 마지막으로 Java 이외에도 C, Python 같은 다양한 언어들을 다룰 수 있고 자동화된 채점 기능을 덧붙일 수 있다면 보다 범용성 있는 교육 도구로 확장 가능할 것으로 기대한다.

※ 논문 투고일: 2021. 7. 12. ※ 논문 수정일: 2021. 8. 6. ※ 게재 확정일 : 2021. 8. 30.

〈참고문헌〉

- 김용천, 김자미, & 이원규(2012). 초등학생의 프로그래밍 학습에서 활동지를 사용한 성찰에 대한 사례 연구. **정보교육학회논문지**, 16(1), 21-31.
- 김은정(2019). 프로그래밍 언어 학습 시스템에서 객관식 문제의 난이도 균등화 알고리즘에 대한 연구. **컴퓨터교육학회 논문지**, 22(3), 55-65.
- 김은지, 이태욱(2018). 초등학생의 컴퓨팅 사고력 향상을 위한 게임 프로그래밍 수업 설계. **한국 컴퓨터정보학회 학술발표논문집**, 26(2), 265-268.
- 김정량(2020). 초등학교 소프트웨어교육에서 학습자의 알고리즘 구성 패턴 연구. **정보교육학회 논문지**, 24(1), 11-19.
- 박주연, 류은진(2021). 초등 프로그래밍 교육을 위한 퍼즐게임의 선정 근거 개발. **학습자중심교과교육연구**, 21(9): 463-479.
- 마대성(2021). SW 교육에 대한 초등교사와 학생의 인식, 태도 비교 분석 연구-SW 교육 선도학교와 일반 학교를 대상으로. **정보교육학회논문지**, 25(1), 185-193.
- 문미예, 김갑수(2018). 초등학생을 위한 Python 프로그래밍 언어 교육방안 연구. **한국정보교육학회 2018년 정보교육학회 학술논문집**, 33-41.
- 서성원(2010). **TPL과 VPL을 활용한 로봇프로그래밍 교육이 정보과학적 사고능력에 미치는 영향**. 한국교원대학교 교육대학원 석사학위청구논문.
- 심재권(2019). 초중등 학습자의 알고리즘적 사고 수준 측정 연구. **창의정보문화연구**, 5(3), 237-243.
- 이영석(2020). 문제해결학습의 알고리즘 교육의 효과성 연구. **융합정보논문지(구 중소기업융합학회논문지)**, 10(8), 173-178.
- 장정아, 차승은, 이원규(2009). 정보과학적 사고 기반의 퍼즐 교재 개발. **한국컴퓨터교육학회 학술발표대회논문집**. 13(1), 181-185.
- 정구원(2015). **마이크로컨트롤러 교육을 위한 비주얼과 텍스트 프로그래밍 언어 적용 분석**. 서울교육대학교 교육전문대학원 석사학위청구논문.
- 정의준, 이혜림(2013). 게이미피케이션에 기반한 게임 3.0 시대의 변화 양상에 관한 연구. **한국 컴퓨터게임학회논문지**, 26(4): 87-97.
- 최정원, 이영준(2015). 초등 정보 영재의 알고리즘 학습을 위한 퍼즐의 교육적 활용. **한국컴퓨터 정보학회 논문지**, 20(5), 151-159.
- 최정원, 이영준(2014). 프로그래밍 학습에서 학습자의 어려움 분석. **컴퓨터교육학회논문지**, 17(5), 89-98.
- 한안나(2018). 국내 게이미피케이션 연구동향에 관한 체계적 문헌고찰. **한국콘텐츠학회논문지**, 18(5), 566-578.
- 허미선, 김종혜, 이원규(2009). 텍스트추얼 및 비주얼 프로그래밍 교육의 비교, 분석. **한국컴퓨터 교육학회 학술발표대회논문집**. 13(1), 123-127.
- Adamuthe, A. C., & Mane, S. U.(2020). PBL Based Teaching-Learning Strategy for Inculcating Research Aptitude in CS/IT Students. **Journal of Engineering Education Transformations**, 33, 585-594.
- Brad Myers.(1990). Taxonomies of visual programming and program visualization. **Journal of Visual Languages & Computing** 1.1.

- Dicheva, D., Dichev C., Agre G., & Angelova G.(2015). Gamification in education: A systematic mapping study. **Educational Technology & Society**, **18(3)**, 75-88.
- Durak, H. Y., Yilmaz, F. G. K., & Yilmaz, R.(2019). Computational thinking, programming self-efficacy, problem solving and experiences in the programming process conducted with robotic activities. **Contemporary Educational Technology**, **10(2)**, 173. <https://doi.org/10.30935/cet.554493>.
- Ericson, B. J., Guzdial, M. J., & Morrison, B. B.(2015). Analysis of interactive features designed to enhance learning in an ebook. In **Proceedings of the Eleventh Annual International Conference on International Computing Education Research** (pp. 169-178).
- Ericson, B. J., Margulieux, L. E., & Rick, J.(2017, November). Solving parsons problems versus fixing and writing code. In **Proceedings of the 17th Koli Calling International Conference on Computing Education Research** (pp. 20-29).
- Jenkins, T.(2002). On the difficulty of learning to program. In **Proceedings of the 3rd Annual Conference of the LTSN Centre for Information and Computer Sciences** (Vol. 4, No. 2002, pp. 53-58).
- Karavirta, V., Helminen, J., & Ihantola, P.(2012). A mobile learning application for parsons problems with automatic feedback. In **Proceedings of the 12th Koli Calling International Conference on Computing Education Research** (pp. 11-18).
- Kumar, A. N.(2018). Epplets: a tool for solving parsons puzzles. In **Proceedings of the 49th ACM Technical Symposium on Computer Science Education** (pp. 527-532).
- Parsons, D., & Haden, P.(2006). Parson's programming puzzles: a fun and effective learning tool for first programming courses. In **Proceedings of the 8th Australasian Conference on Computing Education-Volume 52** (pp. 157-163).
- Tuyano Syoda.(2016). **프로그래밍 가이드북-프로그래머가 되기 전에 꼭 읽어야 할 필독서**. 이영란 역. 정보문화사

〈Abstract〉

Development and Application of Parsons Puzzle Contents to Improve Algorithmic Thinking

Cho, HyunChul¹, Lee, Sihoon²

The study aims to propose puzzle game content for algorithmic thinking enhancement. The Korean 2015 Revised Curriculum presents learning understanding of algorithms through SW education. Puzzle take the form of games, which are suitable for attracting students' interest and participation, and are effective in understanding the underlying algorithms.

In this work we develop and apply content based on Parsons puzzle. The puzzle game operates by Drag & Drop method, which is also easy to use on tablets. The content is composed of text programming contents, but the class was organized to look at the structure of the algorithm with the teacher to prevent excessive grammar-oriented education, and the puzzle game allowed students to check the meaning of the code on the screen.

As a result of applying the developed program to elementary school students with sufficient SW education experience, it was shown that puzzle games could be effectively used to understand text programming algorithms, showing high interest in classes and understanding algorithms. On the other hand, the contents of text programming have been found to be difficult, and it is necessary to expand it to middle school students in the future to conduct follow-up research.

Keywords : Algorithms, text programming, puzzle games, gamification, Parsons puzzle, SW Education

1. Teacher, Cheongju Shinheung High school, pran@naver.com (Lead author)

2. Teacher, Galwon Elementary school, shoon1984@gmail.com (Corresponding author)