

프로그래밍 교육 관련 일본 초등학교 수학 교과서 및 디지털 콘텐츠 분석

권미선(신풍초등학교, 교사)

본 연구는 2020년 발행된 일본 초등학교 수학 교과서 2종의 프로그래밍 교육 관련 특화 차시를 지도 시기 및 주제, 지도 내용의 수학적 구성, 지도 맥락 및 활동 구성의 측면에서 분석하였다. 프로그래밍 교육 관련 특화 차시의 경우, 교과서에 따라 1~6학년 모두에 제시되어 있거나 5~6학년에 집중적으로 제시되어 있었다. 주제는 로봇 이동 시키기, 한붓그리기, 정다각형 그리기, 수 정렬하기 등으로 다양했다. 내용 영역 측면에서는 도형과 측정 영역이 가장 많이 제시되었으며, 수학적 요소로는 순서, 논리, 알고리즘 등의 요소가 가장 많이 구현되었다. 실제 프로그래밍을 구현한 디지털 콘텐츠에는 교과서에 제시된 내용보다 많은 내용들이 포함되어 있었으며, 학생들이 스스로 할 수 있도록 단계별로 매우 상세하게 제시되어 있었다. 마지막으로 코드 블록의 경우 동작, 연산, 선택 블록 등을 많이 사용되었다. 이러한 연구 결과를 토대로 우리나라 초등 수학 교육에서의 프로그래밍 교육에 대한 시사점을 제공하고자 한다.

I. 서론

Wing(2006)에서 컴퓨팅 사고력이 언급된 이후 세계적으로 소프트웨어 교육에 대한 관심이 증가하였다. 우리나라에서도 2015년 소프트웨어 교육을 실시하기 위한 목표, 내용, 방법, 평가 등이 담긴 소프트웨어 운영 지침을 제시하였으며, 2016년에는 초등 소프트웨어 교육과정 표준 모델이 개발되었다(교육부, 2015; 김철외, 2016). 각 교과에서의 소프트웨어 교육을 살펴보면 2015 개정 교육과정에서부터 초등의 경우 실과, 중등의 경우 기술·가정과 정보 교과에서 소프트웨어 교육을 실시하고 있다. 최근에는 소프트웨어 교육에서 더

나아가 AI 교육이 도입되었으며, 교육부에서는 SW·AI 선도 학교 운영, SW·AI 교육 자료 개발 등 여러 측면에서 SW·AI 관련 교육 체계를 구축하고 있다(교육부, 2020). 이처럼 SW·AI 교육은 학교 교육에서 필수적이며 하나의 중요한 학습 주제가 되어가고 있다.

SW·AI 교육은 여러 영역으로 나눌 수 있으며, 그중 소프트웨어의 경우 생활과 소프트웨어, 알고리즘과 프로그래밍의 두 영역으로 구분된다(교육부, 2015). 생활과 소프트웨어 영역은 학생들이 소프트웨어가 생활에 가져온 변화를 알고 정보화 사회를 살아가기 위한 태도를 지니는 것을 목표로 하며, 알고리즘과 프로그래밍 영역은 학생들이 알고리즘과 프로그래밍을 체험해보고 이를 이용하여 실생활의 다양한 문제를 컴퓨팅 사고로 해결할 수 있는 것을 목표로 한다. 즉 생활과 소프트웨어 영역은 학생들이 기술의 소비자로서, 알고리즘과 프로그래밍 영역은 학생들이 기술의 생산자로서 그 역할을 하는 것을 목표로 하고 있다.

두 영역 중 세계 유명 인사들은 프로그래밍 자체의 중요성을 강조하며 교육에서 생산자의 역할에 좀 더 초점을 두어야 한다고 주장하고 있으며, 구체적으로 스티브 잡스의 경우 모든 사람들이 프로그래밍을 배워야 하고 프로그래밍은 생각하는 법을 가르쳐 준다고 이야기하였으며, 마크 저커버그는 다음 세대는 프로그래밍을 읽기와 쓰기처럼 가르치고 있을 것이며, 버락 오바마는 코딩을 배우는 것은 자신의 미래는 물론 조국의 미래에도 매우 중요한 일이라고 제시하였다(YTN 사이언스, 2022). 지금 시대는 컴퓨터와의 협업 없이는 살아가기 힘들며, 이를 위해서는 컴퓨터 언어를 이해하는 것이 필요하다. 흔히 미래 사회를 인공지능과 사람의 대결이라 말하지만, 전문가들은 미래 사회는 인공지능을 이용할 수 있는 인간과 인공지능을 이용하지 못하는 인간 간의 대결이라고 말한다(BArt & Jang, 2017). 이에 다른 사람들이 만들어놓은 프로

* 접수일(2023년 12월 21일), 심사(수정)일(2024년 1월 15일), 게재확정일(2024년 1월 22일)
* MSC2000분류 : 97U20
* 주제어 : 일본 수학 교과서, 프로그래밍 교육, 프로그래밍적 사고, 프로그래밍 교육을 위한 수학적 요소, 디지털 소양

그램을 이용하는 것을 넘어 직접 프로그래밍을 하여 문제를 해결할 수 있도록 지도하는 것이 필요하다.

이제 프로그래밍을 학습하는 것은 선택이 아닌 필수로 되어가고 있다. 최근 실시된 수학 교육의 방향에 대한 연구에서도 미래 사회를 위해 지도해야 할 수학 교육 주제 3가지 중 하나로 프로그래밍을 제시하였다(나귀수 외, 2022). 이웃 나라인 일본 역시 우리나라의 교육과정에 해당하는 2017 개정 소학교학습지도요령(小學校學習指導要領)에서 프로그래밍 교육을 강조하였으며, 수업 개선을 위한 자료 중 하나로 프로그래밍 교육을 위한 교재, 영상 등을 제시하였다. 특히 소학교 학습지도요령 해설(小學校學習指導要領 解説) 산수과에서 프로그래밍을 어떻게 지도해야 할지 구체적인 내용 및 방법을 소개하였으며, 2020년 발행된 수학 교과서에는 이러한 내용들이 반영되어 있다. 이처럼 일본은 소학교 학습지도요령과 그 해설에서 프로그래밍 교육을 제시하고 수학 등의 교과에서 프로그래밍 교육을 구체화하였으며, 이를 통해 프로그래밍적 사고(プログラミング的思考)를 향상시키는 것을 목표로 하고 있다.

또한 핀란드의 경우 초등학교 1학년부터 수학 교과에서 프로그래밍 교육을 실시하고 있으며, 프랑스에서도 중학교부터 수학 교과에서 ‘알고리즘과 프로그래밍’의 내용 영역을 신설하여 교육을 실시하고 있다(권미선, 2023; 이승우, 2020). 하지만 우리나라 수학 교과의 프로그래밍 교육을 살펴보면 초·중학교에서는 프로그래밍 관련 내용을 찾아보기 어려우며, 고등학교에서만 2022 개정 수학과 교육과정에 따라 진로 선택 과목으로 인공지능 수학을 제시하고 있다(권오남 외, 2021). 이에 초등학교 수학에서 프로그래밍 교육을 하나의 수학과 내용 영역으로 지도하기에는 어려움이 있다. 하지만 2022 개정 수학 교육과정 하에서 프로그래밍 교육을 실시하고자 한다면, 수학 교과 역량 측면에서 공학 도구의 사용을 강조하고 있으며 기초 소양 측면에서도 디지털 소양을 제시하고 있어 이러한 측면과 연결하여 지도할 수 있을 것이다. 구체적으로 수학 교과 역량에 대해 교육과정에 제시된 바와 같이 추상적인 수학 내용을 시각화하고 수학의 개념, 원리, 법칙에 대한 직관적 이해와 논리적 사고를 도우며 학생이 주도적으로 공학 도구를 활용하여 탐구할 수 있도록 프로그래밍 교육을 실시한다면 수학과 프로그래밍 교육을 연결하여 제시할 수 있을 것이다. 또한 기초 소양 측

면에서도 수학 내용 특성에 적합한 프로그래밍 도구를 선택하여 효율적인 교수·학습이 이루어지게 한다면 적용이 가능할 것이다.

미래 사회를 준비하기 위해 이제 우리나라에서도 프로그래밍 교육의 도입이 필요하며, 여러 교과 중 논리적 사고를 발달시키며 공학 도구를 활용하여 정보처리 역량을 키울 수 있는 교과인 수학이 이에 적합한 교과 중 하나일 것이다. 또한 2022 개정 수학 교과서에서도 정보처리 역량과 디지털 소양 측면에서 프로그래밍 교육의 도입이 가능하다. 이에 초등학교 수학에서 프로그래밍 교육을 이미 실시하고 있는 일본 수학 교과서를 분석하여 구체적인 지도 방향 및 내용을 살펴보고자 한다. 또한 1학년부터 6학년까지 프로그래밍 교육 내용을 제시한 교과서와 5, 6학년에서만 집중적으로 프로그래밍 교육 내용을 제시한 교과서를 비교하여 분석함으로써 다양한 구성을 살펴보고자 하였다.

II. 이론적 배경

1. 프로그래밍 교육

프로그래밍 교육을 생각할 때, 우선 컴퓨팅 사고를 떠올릴 수 있다. 컴퓨팅 사고(computational thinking)는 Wing(2006)에서 제시되어 세계적인 반향을 일으켰으며 이후 여러 나라에서 소프트웨어 교육이 실시되었다. 컴퓨팅 사고는 ‘컴퓨터 과학의 기본적인 개념과 원리를 기반으로 문제를 효율적으로 해결하도록 생각하는 것’을 의미한다(김대수, 2017; 박성호 외, 2021; Kafai & Burke, 2014). 반면 프로그래밍은 좁은 의미로는 일련의 명령문의 조합으로 알고리즘을 기술하는 것으로 보기도 하며, 넓은 의미로는 해결하는 문제를 이해하고 그 문제를 해결하는 방법인 알고리즘을 설계하고 이를 프로그램으로 작성하고 효율성을 검증하는 과정을 의미한다(한치근, 2018). 이러한 프로그래밍의 의미를 생각해보면, 좁은 의미에서의 프로그래밍 교육은 컴퓨팅 사고력이 기반이 되어야 하며 넓은 의미에서의 프로그래밍 교육은 컴퓨팅 사고력을 포함하는 좀 더 포괄적인 의미를 지닌 발전된 형태의 교육임을 알 수 있다.

본 논문에서는 프로그래밍을 좁은 의미와 넓은 의

미가 절충된 '주어진 문제를 해결해가는 과정에서 정해진 순서나 절차에 따라 논리적으로 명령문을 기술하는 것'으로 정의한다(김대수, 2017; 박성호 외, 2021). 이를 정리하면 프로그래밍은 컴퓨팅 사고와 동일한 것을 의미하지 않으며, 문제 해결 상황에서 살펴보면 해결해야 하는 과제가 주어졌을 때 학생들은 컴퓨팅 사고를 통해 문제를 어떻게 해결할지를 구상하고 이를 실행이 가능하도록 프로그래밍하여 실제 실행해보고 수정하여 문제를 해결하는 것이다.

프로그래밍에서 사용하는 표현과 요소를 구체적으로 살펴보면 학생들은 주어진 문제를 해결하기 위한 프로그램을 기술할 때 대부분 순차적 표현, 반복적 표현, 선택적 표현 등을 사용하며(박성호 외, 2021), 프로그래밍을 하기 위해서는 이러한 표현들을 학습해야 한다. 우리는 어떤 문제를 해결할 때 정해진 순서나 절차에 따라 일을 진행해야 한다. 이때 해야 할 일을 순서대로 나열하는 순차적 표현을 사용할 수 있다. 예를 들어 한 번의 길이가 4cm인 정사각형을 그릴 때, '앞으로 4cm 가기', '왼쪽으로 90° 돌기', '앞으로 4cm 가기', '왼쪽으로 90° 돌기', '앞으로 4cm 가기', '왼쪽으로 90° 돌기', '앞으로 4cm 가기'의 명령어를 차례대로 제시하면 원하는 도형을 그릴 수 있다. 순차적 표현에는 조금 다른 순서로 일을 하여도 결과를 완성하는 데 영향을 미치지 않는 경우도 있으나, 순서가 바뀌면 결과에 영향을 많이 미치는 경우도 있으므로 작업의 선후관계를 반드시 확인하고 지시해야 한다.

문제를 해결하기 위한 순차적 표현에는 일정한 부분의 반복이 있을 수 있다. 이때 일정한 부분이 계속적으로 반복되면 문제 해결을 위한 표현이 너무 길어져서 그 표현을 이해하는 데 어려움을 겪을 수 있다. 이때 표현의 효율성을 위해 반복되는 행동의 묶음을 찾아 '묶음을 몇 번 반복한다'라는 표현을 사용할 수 있으며 이러한 표현이 반복적 표현이다. 예를 들어, 위에서 제시한 한 번의 길이가 4cm인 정사각형을 그릴 때, ['앞으로 4cm 가기', '왼쪽으로 90° 돌기']를 3번 반복하기, '앞으로 4cm 가기'의 명령어를 제시하면 원하는 도형을 그릴 수 있다. 이처럼 프로그래밍에서 간략하고 효율적인 표현의 사용이 중요하며 이를 위해서는 규칙을 찾아 반복적 표현을 사용해야 한다.

선택적 표현을 살펴보면, 문제를 해결할 때 조건에 참·거짓에 따라 다른 행동을 선택해야 하는 경우가 있

다. 이러한 경우 순차적 또는 반복적 표현으로는 해결이 불가능하며, 선택에 따라 다른 행동을 지정해 주어야 한다. 예를 들어, 자율자동차를 신호등 색에 따라 움직이도록 프로그래밍을 할 때, '만약 신호등의 색이 빨간색이면 신호등의 색이 초록색이 될 때까지 기다리기', '만약 신호등의 색이 초록색이면 10cm 움직이기'와 같이 신호등의 색에 따라 선택적 표현을 사용할 수 있다.

이러한 표현들 안에 다양한 프로그래밍 요소들이 사용될 수 있으며, 이러한 것들을 프로그래밍에서는 변수, 연산자, 함수, 디버깅 등으로 이야기한다(Liukas, 2015). 변수는 컴퓨터 프로그램이 숫자나 문자열과 같은 값을 저장하는 곳을 의미하며, 예를 들어 게임에서 플레이어의 이름 같은 정보를 player라는 변수로 저장하여 사용한다. 연산자는 수를 더하거나 빼거나 곱하거나 나누는 표현을 의미하며, 함수는 프로그램 안에서 입력된 독립적인 코드 덩어리로 엑셀의 SUM 함수와 같이 자주 쓰는 코드 조각은 함수로 만드는 것이 유용하다. 또한 프로그래밍에서 문제를 발견하고 해결하는 디버깅 등이 사용될 수 있다.


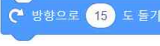







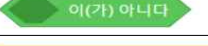


2. 시각적 프로그래밍 도구

시각적 프로그래밍 도구를 이용하면 직접 어려운 프로그래밍 코드를 작성하지 않고 블록을 드래그하여 작성하고 실행할 수 있어 학생들이 쉽고 재미있게 프로그래밍을 할 수 있다. 시각적 프로그래밍 도구에는 스크래치, 아누이노, 앨리스, 앱 인벤터, 로고 등이 있다. 이중 일본 수학 교과서는 스크래치와 유사한 시각적 프로그래밍 도구를 개발하여 활용하고 있으며, 교과서를 소개할 때 스크래치를 이용하여 수업을 할 수 있음을 제시하고 있다. 다만, 프로그래밍을 학습하는 초기에 너무 많은 블록을 제시하면 학습에 방해가 될 수 있어 일본에서 제작한 프로그래밍 도구에는 일부 블록만이 제시되어 있다. 이에 스크래치에 제시된 여러 가지 블록을 알아보고 이러한 블록 중 일본 수학 교과서에서는 어떠한 블록들이 사용되고 있는지 살펴 보았다.

스크래치에서 사용하고 있는 코드 블록은 8종류(동작, 형태, 소리, 이벤트, 제어, 감지, 연산, 변수)가 있으며 이외에 자신이 원하는 블록을 만들 수도 있다([표

1] 참조). 이중 수학과 관련이 있는 코드 블록에는 동작, 형태, 제어, 연산, 변수가 있다. 구체적으로 살펴보면 동작 블록의 경우 이동과 관련되어 있으며, 대표적으로 ○만큼 움직이기, 시계 방향으로 ○도 돌기, 시계 반대 방향으로 ○도 돌기, 좌표점으로 이동하기 등이 있다. 형태 블록의 경우 크기와 모양의 변화와 관련되어 있으며, 대표적으로 크기를 ○만큼 바꾸기, 크기를 ○%로 바꾸기 등이 있다. 제어 블록의 경우 다른 코드를 제어하는 것과 관련되어 있으며 반복하기, 가정문 처리하기 등이 있다. 특히 제어 블록의 경우 동작, 형태, 연산 등의 블록이 그 안으로 포함되도록 모양이 만들어져 있다. 연산 블록의 경우 수와 연산과 관련되어 있으며 더하기, 빼기, 곱하기, 나누기, 수의 크기 비교하기 등의 블록이 있다. 마지막으로 변수 블록의 경우 변수를 일정한 수로 정하거나 수를 바꾸거나 하는 등의 블록이 제시되어 있다.

[표 1] 스크래치의 코드 블록

코드	수학적 내용	블록 예시
동작	점의 이동 방향 각도	 
형태	시각 비율 위치	  
제어	규칙 가능성	 
연산	사칙연산 수의 크기 비교 명제	  
변수	변수	 

3. 프로그래밍 교육 관련 일본 교육과정

교과서를 분석하기에 앞서 일본 교육과정에 제시된 프로그래밍 교육 관련 내용을 살펴보고자 한다. 2017년 개정된 일본의 소학교학습지도요령 해설 수학과에서는 ‘개정의 취지 및 요점’에서 교과 목표로 수학을 통해 지식 및 기능, 사고력·판단력·표현력, 학습하고자 하는 힘·인간성의 세 가지 측면을 키우는 것을 제시하였다. 이를 위해 수학과 내용을 수학의 유용성을 느끼고, STEM 교육과 언어로서의 수학을 고려하며, 통계 교육을 개선하는 것에 초점을 두어 개선하고자 하였다. 이와 더불어 프로그래밍 교육을 도입해야 하는 필요성을 제시하였으며, 그 예로 6학년에서 데이터의 분포에 대한 프로그래밍 교육을 실시할 수 있다고 제시하였다.

‘학년별 교육 내용’에도 프로그래밍 관련 내용을 제시하였으며 구체적으로 살펴보면, 5학년 도형 영역에서 정확한 반복 작업이 필요한 정다각형의 작도 시 프로그래밍을 체험할 수 있다고 제시하였다. 또한 우리나라 교육과정의 ‘교수·학습 및 평가’에 해당하는 ‘지도 계획의 작성과 내용의 반영’에서는 프로그래밍이 생활의 편리함과 풍부함을 가져오고 있음을 학생들이 이해하고 그러한 프로그래밍을 학생들이 의도한 활동에 활용할 수 있도록 하여, 시대를 넘어 보편적으로 요구되며 장래 어떤 직업을 가지더라도 필요한 프로그래밍적 사고를 기르는 교육을 실시해야 한다고 제시하였다.

특히 소학교학습지도요령 해설 수학과에서는 프로그래밍적 사고를 ‘자신이 의도하는 일련의 활동을 실현하기 위해 어떠한 움직임의 조합이 필요하며, 어떻게 개선하면 보다 의도한 활동에 가까워지는 것인가를 논리적으로 생각해가는 힘’으로 정의하고 있으며, 단순히 프로그래밍을 학습하는 것을 넘어 그러한 사고의 학습을 목표로 하고 있는 것이다. 또한 프로그래밍 교육을 교과 학습과 관련시키면서 학생들의 발달 단계에 따라 언제 가르칠지를 정해야 한다고 제시하였다. 수학에서 문제를 해결한 후 해결 방법을 되돌아보고 보다 간결·명료·적정한 것으로 수정하거나 그것을 순서에 따라 정리하는 학습이 많이 이루어지며, 수학을 통해 문제를 해결하는 데 필요한 절차가 있음을 알 수 있으므로 일본 교육과정에서는 수학 교과를 통해 프로그래밍 교육을 실시하는 것이 적합하다고 제시하였다.

III. 연구 방법 및 절차

1. 연구 대상

본 연구는 프로그래밍 교육 관련 특화 차시를 분석하기 위하여 2017년 개정된 일본 교육과정인 ‘소학교 학습지도요령과 그 해설’에 따라 발간된 Gakko Tosho와 Tokyo Shoseki의 일본 수학 교과서를 분석하였다([표 2] 참조). 일본에서 사용하고 있는 여러 가지 교과서가 있으나 Gakko Tosho의 경우 1~6학년 모두에 프로그래밍 교육 관련 특화 차시를 제시하고 있으며, Tokyo Shoseki의 경우 5, 6학년에서만 프로그래밍 교육 관련 특화 차시를 제시하고 있다. 전반적인 프로그래밍 교육의 체계를 살펴보기 위해서 서로 다른 유형의 교과서를 살펴보는 것이 필요할 것이다. 따라서 두 유형의 교과서를 선택하여 일본 수학 교과서에서 제시하고 있는 프로그래밍 교육을 살펴보고 수학 교과에서 어떻게 프로그래밍 교육을 할 수 있는지 알아보았다.

두 교과서의 프로그래밍 교육 관련 특화 차시만을 분석 대상으로 하였으며, 구체적으로 Gakko Tosho의 경우 프로그래밍 교육을 위한 특화 차시를 ‘컴퓨팅 사고’라는 차시 명으로 전 학년의 교과서 마지막 부분에 제시하고 있으며, Tokyo Shoseki의 경우 ‘프로그래밍을 체험하자’라는 차시를 5, 6학년의 각 학기 교과서의 마지막 부분에 제시하고 있다. 단, 6학년 교과서의 경우 학기 구분 없이 한 권으로 제작되어 5학년에서는 두 차시, 6학년에서는 한 차시를 제시하고 있다.

[표 2] 연구 대상

	교육과정	교과서	
발행 년도	2017	2020	
문서명	小學校學習指導要領解説【算數編】	Mathematics for Elementary School	New Mathematics for Elementary School
발행 기관	Ministry of Education, Culture, Sports, Science and Technology	Gakko Tosho (學教圖書)	Tokyo Shoseki (東京書籍)

2. 분석 기준

본 연구에서는 [표 3]의 세 가지 부분에 초점을 두어 분석하였다. 첫째, 지도 시기와 주제에 관련하여 어떠한 학년 학기에 지도되는가, 어떠한 주제에 대한 것인가를 분석하였다. 둘째, 지도 내용의 수학적 구성에 관련하여 어떠한 수학과 내용 영역과 관련되어 있는가, 어떠한 프로그래밍 교육을 위한 수학적 요소가 포함되어 있는가를 기준으로 분석하였다. 이 중 프로그래밍 교육을 위한 수학적 요소의 경우, 권미선(2023)에 제시된 프로그래밍 교육을 위한 수학적 요소인 순서, 규칙, 변수, 가능성, 함수, 선택, 논리, 알고리즘을 활용하여 각 활동이 어떠한 요소와 관련되어 있는지 분석하였다. 구체적으로 순서는 올바른 순서대로 문제를 해결하는 활동인가, 규칙은 반복적으로 해결하는 부분이 포함된 문제를 해결하는 활동인가, 변수는 변수를 사용하거나 변수의 값을 찾는 활동인가, 가능성은 사건이 일어나는 특징에 따른 경우의 수를 파악하는 활동인가, 함수는 어떠한 함수에서 입력값에 대응하는 출력값을 찾는 활동인가, 선택은 조건의 참과 거짓을 판단하여 선택하는 활동인가, 논리는 논리적으로 추론하여 문제를 해결하는 활동인가, 알고리즘은 문제를 해결하기 위한 체계적인 단계를 나열하는 활동인가에 대한 것이다. 마지막으로 지도 맥락 및 활동 구성에 관련하여 어떠한 맥락에서 지도되는가, 어떤 디지털 활동들로 구성되어 있는가, 어떠한 코드 블록을 사용하였는가에 대해 분석하였다.

[표 3] 분석 기준

분석 기준	분석 내용
지도 시기 및 주제	<ul style="list-style-type: none"> 어떠한 학년, 학기에 지도되는가? 어떠한 주제에 대한 것인가?
지도 내용의 수학적 구성	<ul style="list-style-type: none"> 어떠한 수학과 내용 영역과 관련되어 있는가? 어떠한 프로그래밍 교육을 위한 수학적 요소(순서, 규칙, 변수, 가능성, 함수, 선택, 논리, 알고리즘)가 포함되어 있는가?
지도 맥락 및 활동 구성	<ul style="list-style-type: none"> 어떠한 맥락에서 지도되는가? 어떤 디지털 활동들로 구성되어 있는가? 어떠한 코드 블록을 사용하는가?

3. 분석 방법

본 연구에서는 분석 기준 중 지도 맥락 및 활동 구성의 디지털 활동을 제외한 모든 분석 내용은 교과서를 대상으로 실시하였다. 분석 기준 3가지 중 지도 시기 및 주제, 지도 맥락 및 활동 구성에 대한 분석은 명확한 반면, 지도 내용의 수학적 구성 중 프로그래밍 교육을 위한 수학적 요소에 대한 분류는 연구자마다 차이를 보일 수 있다. 따라서 프로그래밍 교육을 위한 수학적 요소에 대해 각 차시별로 초등 수학 교육 박사 학위 소지자 3인이 개별적으로 적절히 구현됨 3점, 부분적으로 구현됨 2점, 구현되지 않음 1점의 3가지 수준으로 분석하였다. 연구 결과에는 분석자 3인의 평균 점수를 제시하고, 평균 점수가 2점 이상인 경우 구현되었으므로 해석하여 기술하였다.

분석자 간 상관관계를 분석한 결과는 [표 4]와 같다. 상관 계수는 0.652~0.950으로 나타났으며, 상관 계수는 0.5~0.7이면 중간 상관관계, 0.8~1이면 강한 상관관계를 나타낸다(강상욱 외, 2022). 따라서 분석자 간의 중간 이상의 상관관계가 있는 것으로 나타났다. 각 요소에 대한 구현 여부는 분석자에 따라 약간의 차이가 있을 수 있으나 어느 정도 일치되었음을 알 수 있다.

[표 4] 분석자간 상관 관계 (N=64)

	분석자A	분석자B	분석자C
분석자A	1.000		
분석자B	0.702	1.000	
분석자C	0.950	0.652	1.000

IV. 연구 결과

1. 전반적인 프로그래밍 교육 관련 교과서 구성

Gakko Toshō의 프로그래밍 교육 관련 차시는 각 학년의 마지막 부분에 제시되어 있으며, 모두 2쪽으로 구성되어 있다. 반면 Tokyo Shoseki의 프로그래밍 교육 관련 차시는 새수학 플러스라는 명칭으로 각 학년의 마지막 부분에 선택형 과제로 제시되어 있으며, 5학년 1학기에 1쪽, 5학년 2학기에 1쪽, 6학년에 2쪽으로 구성되어 있다.

구체적으로 각 교과서에 제시된 지도 시기 및 주제를 살펴보면([표 5] 참조), Gakko Toshō의 경우, 1~2학년에서는 로봇을 목적지까지 이동시키기, 하노이 타워의 원판 옮기기, 3~4학년에서는 무게가 다른 물체 찾기, 한붓그리기, 5~6학년에서는 정다각형 그리기, 수 정렬하기에 대한 프로그래밍 활동을 제시하였다. Tokyo Shoseki의 경우, 5학년에서는 공배수 구하기, 정다각형 그리기, 6학년에서는 수 정렬하기에 대한 프로그래밍 활동을 제시하였다. 대부분의 활동은 전반적인 안내와 세부 활동 몇 개로 구성되어 있다.

Gakko Toshō의 경우 1~6학년에 걸쳐 특별히 선택형이라는 용어를 언급하지 않았으며 다른 단원의 차시와 동일하게 프로그래밍 활동을 제시한 반면, Tokyo Shoseki의 경우 5~6학년에서만 프로그래밍 활동을 제시하였으며, 교과서 차례 부분에 선택하여 학습하는 자료임을 밝혔다. 두 출판사가 제시하는 방식에는 차이를 보였으나 5~6학년에서 제시된 주제는 정다각형 그리기, 수 정렬하기로 동일하다. 이는 ‘소학교학습지도요령 해설 수학과’에서 프로그래밍 활동의 예시로 제시된 정다각형 그리기와 수 정렬하기에 대한 내용을 반영하였기 때문으로 보인다. 각 출판사는 교과서와 연계된 프로그래밍 관련 디지털 콘텐츠를 제작하였으며, 교과서에 제시된 코드 블록과 같은 코드 블록을 제작하여 실제 프로그래밍을 경험할 수 있도록 하였다. 디지털 콘텐츠에서는 교과서에 제시된 내용을 좀 더 상세화하여 기초적인 내용을 다루는 단계에서부터 복잡한 프로그래밍을 수행하는 단계까지 여러 단계로 제시하고 있다. 디지털 콘텐츠는 스크래치와 유사한 형태로 구성되어 있으나 해당 교과서의 과제를 해결하기 위해 새로운 코드 블록들이 제시되기도 하였다.

각 교과서에 제시된 지도 내용의 수학적 구성을 분석한 결과는 [표 6]과 같다. 구체적으로 Gakko Toshō의 교과서를 살펴보면, 1학년의 로봇을 목적지까지 이동시키기 활동의 경우 우리나라 수학과 교육과정의 내용 영역 중 도형과 측정 영역과 관련되어 있으며, 수학적 요소의 경우 순서, 변수, 논리, 알고리즘이 구현되어 있었으며, 규칙은 전혀 제시되지 않았다. 2학년의 하노이 타워의 원판 옮기기 활동의 경우, 변화와 관계 영역과 관련되어 있으며 순서, 규칙, 함수, 논리, 알고리즘이 구현되어 있었으며, 변수, 가능성, 선택은 제시되지 않았다. 3학년의 무게가 다른 물체 찾기 활동의

경우, 도형과 측정 영역과 관련되어 있으며 순서, 규칙, 가능성, 함수, 선택, 논리, 알고리즘이 구현되었으나 변수는 제시되지 않았다. 4학년의 한붓그리기의 경우, 도형과 측정 영역과 관련되어 있으며 순서, 규칙, 가능성, 논리, 알고리즘이 포함되어 있으나 변수, 함수, 선택이 제시되지 않았다. 5학년 정다각형 그리기 활동의 경우, 도형과 측정 영역과 관련되어 있으며, 순서, 규칙, 변수, 논리, 알고리즘이 구현되어 있으나 가능성, 함수, 선택은 제시되지 않았다. 6학년 수 정렬하기 활동의 경우, 자료와 가능성과 관련되어 있으며, 모든 수학적 요소가 포함되어 있다. 이처럼 Gakko Toshō의 교과서의 경우 많은 활동이 도형과 측정 영역과 관련되어 있으며, 변화와 관계 또는 자료와 가능성 영역도 일부

제시되었다. 수학적 요소의 경우 순서, 논리, 알고리즘은 모든 활동에 구현되어 있었으며, 선택이 가장 적게 구현되어 있었다. 이는 프로그래밍의 표현 중 선택적 표현이 가장 난이도가 높은 표현 방식이기 때문으로 볼 수 있다.

Tokyo Shōseki의 교과서를 살펴보면, 5학년 1학기 공배수 구하기 활동의 경우, 수와 연산 영역과 관련되어 있으며 5학년 2학기 정다각형 그리기 활동의 경우, 도형과 측정 영역과 관련되어 있으며 모든 수학적 요소가 포함되어 있었다. 6학년 수 정렬하기 활동의 경우도 자료와 가능성 영역과 관련되어 있으며 모든 수학적 요소가 포함되어 있었다.

Tokyo Shōseki의 교과서의 경우 Gakko Toshō의

[표 5] 프로그래밍 관련 내용의 지도 시기 및 주제

출판사	학년학기	활동 주제	활동 구성
Gakko Toshō	1-2	로봇을 목적지까지 이동시키기	지시어 알기 → 활동 1: 지시에 따라 옮겨 움직인 로봇 찾기 → 활동 2: 불이 난 곳을 피해 이동하기
	2-2	하노이 타워의 원판 옮기기	하노이 타워 규칙 알기 → 활동 1: 제시된 지시어에 따라 ㉠ 기둥에서 ㉡ 기둥으로 원판을 옮기기 → 활동 2: ㉢ 기둥으로 원판 옮기기
	3-2	무게가 다른 물체 찾기	활동 1: 저울을 이용하여 무게가 다른 공 찾기 → 무게가 다른 공을 찾는 방법 알기
	4-2	한붓그리기	한붓그리기 알기 → 활동 1: 제시된 도형에서 시작점 찾기 → 활동 2: 기호에 따라 로봇 이동시키기 → 활동 3: 활동 2와 다른 방법으로 로봇 이동시키기 → 활동 4: 제시된 도형에서 한붓그리기에 따라 로봇 이동시키기 → 활동 5: 한붓그리기가 가능한 도형을 찾아 로봇 이동시키기
	5-2	정다각형 그리기	다각형의 성질과 로봇을 이동시키는 방법 알기 → 활동 1: 지시에 따라 정사각형 그리기 → 활동 2: 제시된 격자에 지시에 따라 정사각형 그리기 → 활동 3: 정팔각형을 그릴 때 로봇에게 할 수 있는 지시 생각하기 → 활동 4: 제시된 격자에 활동 3에서 생각한 지시에 따라 정팔각형 그리기
	6-2	수 정렬하기	활동 1: 1~4의 카드 4장을 오름차순으로 정렬하는 방법 생각하기 → 활동 2: 정렬하는 방법을 친구에게 설명하기 → 활동 3: 1~8의 카드 8장을 오름차순으로 정렬하기
Tokyo Shōseki	5-1	공배수 구하기	3의 배수를 찾기 위한 명령문 생각하기 → 활동 1: 3의 배수의 특징 생각하기 → 활동 2: 10까지의 수에서 3의 배수 찾기 → 활동 3: 20까지의 수에서 4의 배수를 찾기 위한 명령문 알아보기
	5-2	정다각형 그리기	다각형을 그리기 위한 코딩 블록 알아보기 → 활동 1: 한 변의 길이가 10cm인 정사각형을 그리는 방법 생각하기 → 활동 2: 한 변의 길이가 10cm인 정삼각형 그리는 방법 생각하기
	6-2	수 정렬하기	컴퓨터에서 수를 정렬하는 예 알아보기 → 활동 1: 4단계에서 마지막에 놓일 수 찾기 → 활동 2: 수를 한 번 더 정렬했을 때 마지막에 자리가 확정될 수 찾기 → 활동 3: 제시된 방법에 따르면 수가 오름차순으로 정렬되는 이유 설명하기 → 활동 4: 그림을 이용하여 수 정렬하기

[표 6] 프로그래밍 관련 내용의 수학적 구성

출판사	학년	활동 주제	관련 수학과 내용 영역	수학적 요소							
				순서	규칙	변수	가능성	함수	선택	논리	알고리즘
Gakko To sho	1	로봇을 목적지까지 이동시키기	도형과 측정	3	1	2.3	1.3	1.7	1.7	2	2.7
	2	하노이 타워의 원판 옮기기	변화와 관계	3	2	1	1	3	1	3	3
	3	무게가 다른 물체 찾기	도형과 측정	3	2.3	1	3	3	3	3	3
	4	한붓그리기	도형과 측정	3	3	1	2.3	1	1	2.3	3
	5	정다각형 그리기	도형과 측정	3	3	3	1	1	1	2.3	3
	6	수 정렬하기	자료와 가능성	3	2.7	3	3	3	3	3	3
Tokyo Shoseki	5	공배수 구하기	수와 연산	3	2.7	3	2.3	3	3	3	3
		정다각형 그리기	도형과 측정	3	2.7	2.3	2.2	2.3	2.0	2.7	3
	6	수 정렬하기	자료와 가능성	3	3	3	3	3	3	3	3

* 3점: 적절히 구현됨, 2점: 부분적으로 구현됨, 1점: 구현되지 않음

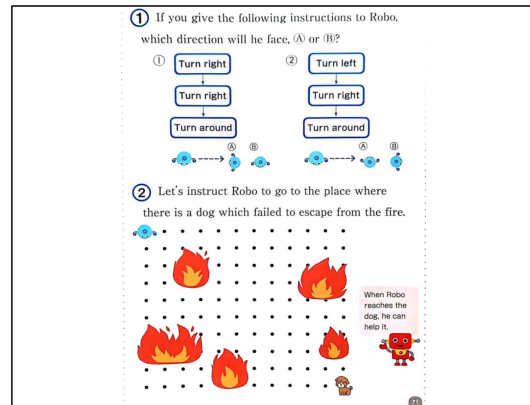
교과서보다 활동은 적게 제시하고 있으나 수학과 내용 영역을 모두 다르게 제시하고 있으며 Gakko Tosho의 교과서에는 제시되지 않은 공배수에 대한 내용을 제시하기도 하였다. 또한 모든 활동에 프로그래밍 교육을 위한 모든 수학적 요소를 포함하고 있었다. 이를 통해 Tokyo Shoseki에서 제시된 활동으로 학습할 경우 학생들이 좀 다양한 수학적 내용과 요소를 학습할 수 있을 것으로 예상된다.

2. 학년별 지도 맥락 및 활동 구성

가. 1~2학년의 지도 맥락 및 활동 구성

1학년은 로봇을 목적지까지 이동시키기, 2학년은 하노이 타워의 원판 옮기기를 주제로 프로그래밍 교육 관련 특화 차시를 구성하였다([그림 1], [그림 2] 참조). 구체적으로 지도 맥락 및 활동 구성을 살펴보면 1학년의 경우 도입, 활동 1, 활동 2로 구성되어 있으며, 도입에서는 로봇이 움직일 수 있는 4가지 지시어(오른쪽으로 돌기, 왼쪽으로 돌기, 뒤로 돌기, 0걸음 앞으로 가기)를 그림과 함께 제시하고 있다. 활동 1에서는 도입 시 제시된 지시어를 이용하여 다음 지시에 따라 움직이는 로봇을 찾도록, 활동 2에서는 도입 시 제시된 지시어를 이용하여 로봇이 불을 피해 강아지에게 갈

수 있도록 지시를 내리는 활동으로 구성되어 있다.

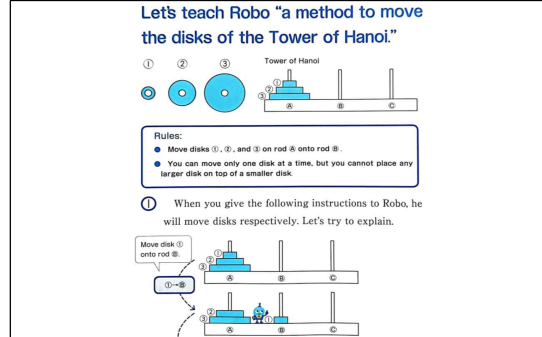


[그림 1] 1학년 교과서 구성(Isoda et al., 2020a, p. 71)

2학년의 경우, 도입에서는 하노이 타워의 규칙인 한번에 한 원판만을 옮겨야 하며, 작은 원판 위에 큰 원판을 놓을 수 없다는 것을 제시하고 있다. 활동 1에서는 로봇에게 원판을 A기둥에서 B기둥으로 옮기는 지시를 제시하고 있으며, 학생들에게 그것을 설명하도록 요구하고 있다. 이때 로봇에게 주는 지시는 총 7개로 교과서에서는 그림과 함께 이를 구체적으로 제시하고

있다. 활동 2에서는 원판을 A기둥에서 C기둥으로 옮기도록 로봇에게 가르칠 것을 요구하고 있다.

1~2학년의 프로그래밍 교육 관련 차시는 [표 7]과 같이 모두 디지털 콘텐츠로 제작되어 있으며, 학생들은 QR 코드나 활동 번호를 입력하여 연결할 수 있다. 1학년의 로봇을 목적지까지 이동시키기 활동의 경우, 초기 화면의 왼쪽에는 ‘오른쪽으로 돌기’, ‘왼쪽으로 돌기’, ‘뒤로 돌기’, ‘앞으로 0걸음 가기’의 코드 블록이 제시되어 있다. 화면의 오른쪽에는 격자판에 로봇과 강아지가 있으며, 코드 블록으로 코딩을 실시할 경우,



[그림 2] 2학년 교과서 구성(Isoda et al., 2020b, p. 122)

[표 7] 1~2학년의 프로그래밍 관련 디지털 콘텐츠 구성

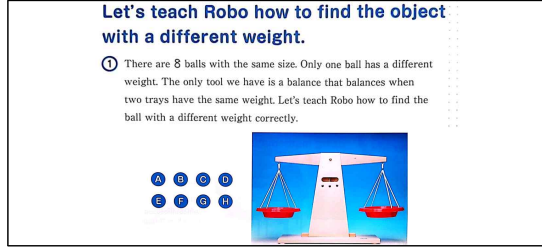
학년	활동 주제	화면 구성	화면 설명
1	로봇을 목적지까지 이동시키기		<ul style="list-style-type: none"> 불을 피해 강아지에게 가도록 프로그래밍하는 초기 화면 화면의 왼쪽에는 ‘오른쪽으로 돌기’, ‘왼쪽으로 돌기’, ‘뒤로 돌기’, ‘앞으로 0걸음 가기’ 블록이 제시됨 화면의 오른쪽에는 지도가 제시됨
			<ul style="list-style-type: none"> ‘앞으로 5걸음 가기’, ‘왼쪽으로 돌기’, ‘앞으로 5걸음 가기’ 블록을 순차적으로 코딩한 후 실행한 화면 로봇이 움직이면, 해당하는 코딩 블록의 테두리가 검정색으로 바뀜
			<ul style="list-style-type: none"> 로봇을 목적지까지 이동시키기 프로그래밍과 관련하여 제시된 6단계의 지도 화면 각 지도를 클릭하면 해당 지도가 있는 화면으로 이동함
2	하노이 타워의 원판 옮기기		<ul style="list-style-type: none"> 모든 원판을 A기둥에서 B기둥으로 이동하는 프로그래밍 초기 화면 화면의 왼쪽에는 ①원판을 O기둥에서 O기둥으로 옮기기, ②원판을 O기둥에서 O기둥으로 옮기기, ③원판을 O기둥에서 O기둥으로 옮기기 블록이 제시됨 화면의 오른쪽에는 하노이 타워의 기둥과 원판이 제시됨
			<ul style="list-style-type: none"> ①원판을 A기둥에서 B기둥으로 옮기기, ②원판을 A기둥에서 C기둥으로 옮기기, ①원판을 B기둥에서 C기둥으로 옮기기, ③원판을 A기둥에서 B기둥으로 옮기기, ①원판을 C기둥에서 A기둥으로 옮기기, ②원판을 C기둥에서 B기둥으로 옮기기, ①원판을 A기둥에서 B기둥으로 옮기기 블록을 순차적으로 코딩한 후 실행한 화면

오른쪽 격자판에서 로봇이 움직이도록 구성되어 있다. 처음 제시된 활동의 경우, ‘앞으로 5걸음 가기’, ‘왼쪽으로 돌기’, ‘앞으로 5걸음 가기’ 블록을 순차적으로 코딩한 후 실행할 수 있으며, 로봇이 움직이면 해당하는 코드 블록의 테두리가 검정색으로 바뀐다. 이때 사용된 코드 블록은 스크래치의 동작 블록과 유사하나 아직 각도를 학습하지 않았기 때문에 ‘90도 돌기’ 블록의 경우 ‘오른쪽으로 돌기’ 또는 ‘왼쪽으로 돌기’ 블록으로 제작하였다. 디지털 콘텐츠는 교과서보다 더 상세하게 여러 단계로 제시되어 있으며, 교과서에는 불이 난 지점이 5개 있는 지도만 제시되었다면, 디지털 콘텐츠에서는 격자의 수와 불의 수가 다양하게 제시된 여러 지도가 제공되었다.

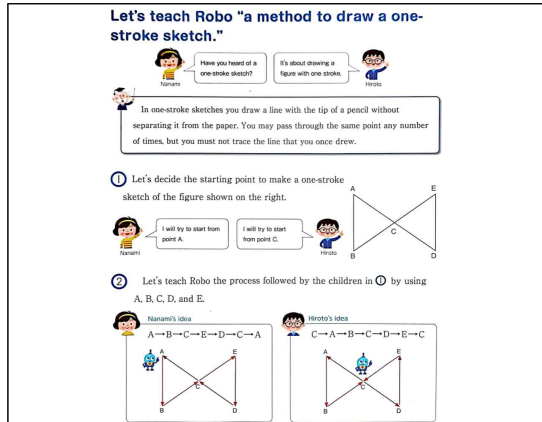
2학년 하노이타워의 원판 옮기기의 경우, 초기 화면의 왼쪽에 세 개의 코드 블록이 제시되어 있다. 구체적으로 ①원판을 ○기둥에서 ○기둥으로 옮기기, ②원판을 ○기둥에서 ○기둥으로 옮기기, ③원판을 ○기둥에서 ○기둥으로 옮기기 블록이 제시되어 있으며 이를 이용하여 모든 원판을 A기둥에서 B기둥으로 이동할 수 있도록 구성되어 있다. 화면의 오른쪽에는 하노이 타워와 원판이 있어 프로그램을 실행하면 제시된 블록에 따라 이동하게 된다. 기존에 있는 시각적 프로그래밍 도구인 스크래치를 이용하여 원판을 ○기둥에서 ○기둥으로 옮기는 것은 하나의 간단한 블록으로 구현하기에는 어려움이 있어, 새로운 블록을 제작하여 학생들이 쉽게 원판을 움직일 수 있도록 제시한 것으로 보인다.

나. 3~4학년의 지도 맥락 및 활동 구성

3학년은 무게가 다른 물체 찾기, 4학년은 한붓그리기를 주제로 프로그래밍 교육 관련 특화 차시를 구성하였다(그림 3, [그림 4] 참조). 구체적으로 지도 맥락 및 활동 구성을 살펴보면 3학년의 경우 하나의 활동으로 구성되어 있으며, 같은 크기의 공 8개 중 하나만 무게가 다를 때, 접시가 두 개인 저울을 이용하여 다른 무게의 공을 찾는 활동으로 구성되어 있다. 학생들은 공을 세 개의 그룹으로 만든 후 두 그룹의 무게를 저울로 비교하여 무게가 다른 공을 찾을 수 있다. 교과서 하단에 다른 무게의 공을 찾는 방법을 수형도로 제시하여 학생들의 이해를 돕고 있으며, 이를 활용하여 디지털 콘텐츠에서 실제 프로그래밍을 쉽게 할 수 있다.



[그림 3] 2학년 교과서 구성(Isoda et al., 2020c, pp. 138-139)



[그림 4] 4학년 교과서 구성(Isoda et al., 2020d, p. 142)

4학년의 경우, 도입에서는 붓을 한 번도 종이 위에서 떼지 않고, 같은 점은 지날 수 있으나 같은 선을 두 번 지나지 않으면서 도형을 완성하는 한붓그리기의 규칙을 제시하고 있다. 활동 1에서는 제시된 도형에서 시작점을 정하는 활동을, 활동 2에서는 기호를 사용하여 제시된 방법으로 로봇을 움직이는 활동을, 활동 3에서는 활동 2와 다른 방법으로 로봇을 움직이는 활동을, 활동 4에서는 제시된 2가지 도형에서 한붓그리기로 로봇을 움직이는 활동을, 활동 5에서는 한붓그리기가 가능한 도형을 찾고, 그 도형에서 로봇을 움직이는 활동을 제시하고 있다.

3~4학년의 프로그래밍 교육 관련 차시의 디지털 콘텐츠는 [표 8]과 같다. 3학년의 무게가 다른 물체 찾기 활동의 경우, 초기 화면의 왼쪽에는 저울 그림, 공 그림(무게를 모르는 공을 담은 칸, 무게가 같은 공을 담은 칸), 정답 확인하기 버튼, 공 다시 모으기 버튼이

[표 8] 3~4학년의 프로그래밍 관련 디지털 콘텐츠 구성

학년	활동 주제	화면 구성	화면 설명
3	무게가 다른 물체 찾기		<ul style="list-style-type: none"> 무게가 다른 물체를 찾으도록 프로그래밍하는 초기 화면 화면의 왼쪽에는 저울 그림, 공 그림(무게를 모르는 공을 담은 칸, 무게가 같은 공을 담은 칸), 정답 확인하기 버튼, 공 다시 모으기 버튼이 제시됨 화면의 오른쪽에는 저울과 공을 선택하는 버튼이 제시됨
			<ul style="list-style-type: none"> 저울이 수평을 이룰 때는 빨간색 선으로, 저울이 수평을 이루지 않을 때는 파란색으로 표시 수형도를 이용하여 무게가 다른 물체를 찾는 활동
			<ul style="list-style-type: none"> 무게가 다른 물체를 찾는 전체 경우의 수를 모두 확인했을 경우 성공이라는 문구가 제시됨
4	한붓그리기		<ul style="list-style-type: none"> 한붓그리기를 할 수 있도록 프로그래밍하는 초기 화면 화면의 왼쪽에는 A, B, C, D, E, F, G, H, I, J와 같이 점의 이름이 블록으로 제시됨 화면의 오른쪽에는 한붓그리기를 할 도형이 제시됨
			<ul style="list-style-type: none"> 제시된 블록의 순서대로 로봇이 이동하여, 별이 만들어지면 완성이라는 문구가 제시됨
			<ul style="list-style-type: none"> 제시된 도형뿐만 아니라 모두 6개의 도형을 그릴 수 있으며, 이 중 2개의 그림이 교과서에 제시되어 있음

제시되어 있으며, 오른쪽에는 저울과 공을 선택하는 버튼이 제시되어 있다. 구슬을 선택하여 각 저울의 접시에 올려놓고 이를 이용하여 무게가 다른 공을 찾는다. 저울이 수평을 이룰 때는 빨간색 선으로, 저울이 수평을 이루지 않을 때는 파란색 선으로 표시하며 수형도를 이용하여 무게가 다른 물체를 논리적으로 판별한다. 3학년의 경우 블록을 제시하지 않았으며, 선택을 통해 다양한 가능성을 파악해 보는 활동으로만 제시하고 있다.

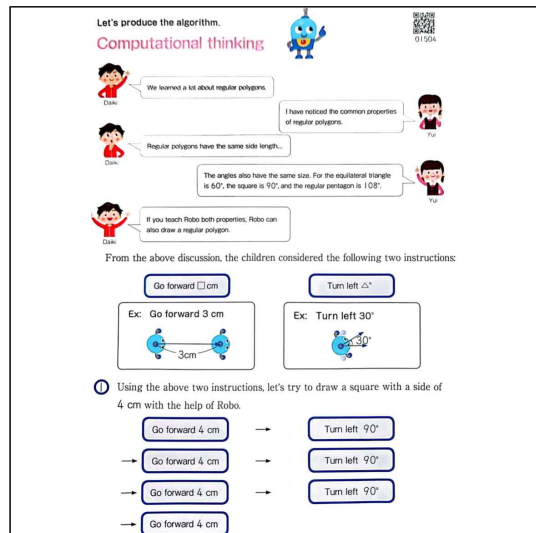
4학년의 한붓그리기 활동의 경우, 초기 화면의 왼쪽에는 A, B, C, D, E, F, G, H, I, J와 같이 점의 이름이 블록으로 제시되어 있으며, 화면의 오른쪽에는 도형이 제시되어 있다. 제시된 블록의 순서대로 로봇이 이동하여, 별이 만들어지면 완성이라는 문구가 제시된다. 디지털 콘텐츠에는 교과서에 제시된 2개의 도형뿐만 아니라 새로운 도형 4개가 더 제시되어 총 도형 6개를 제시하고 있다. 이때 사용된 블록은 스크래치에 제시된 블록보다 더욱 간단하게 각 꼭짓점의 이름을 블록으로 제시하고 있다. 이를 통해 시각적 프로그래밍 도구를 이용해본 경험이 없는 학생들도 쉽게 활용할 수 있을 것으로 보인다.

다. 5~6학년의 지도 맥락 및 활동 구성

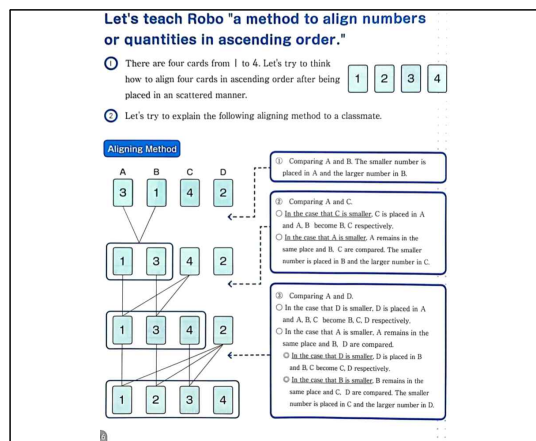
5~6학년의 경우, Gakko Tosho와 Tokyo Shoseki 모두에서 프로그래밍 관련 특화 차시를 제시하고 있다. 우선 1~6학년까지 모두 프로그래밍 교육 관련 차시를 제시하고 있는 Gakko Tosho의 활동 구성을 살펴보고자 한다. Gakko Tosho의 경우, 5학년은 정다각형 그리기, 6학년은 수 정렬하기를 주제로 프로그래밍 교육 관련 특화 차시를 구성하였다([그림 5], [그림 6] 참조). 구체적으로 지도 맥락 및 활동 구성을 살펴보면 5학년의 경우 도입, 활동 1~4로 구성되어 있으며 도입의 경우 학생들이 정다각형의 성질을 이야기하는 장면을 제시한 후, 로봇을 이용하여 정다각형을 그릴 때 사용할 수 있는 명령문을 제시하고 있다. 또한 활동 1에서는 로봇을 이용하여 한 변이 4cm인 정사각형 그리기, 활동 2에서는 로봇이라고 생각하고 격자에 활동 1에서 제시된 명령문에 따라 정사각형 그리기, 활동 3에서는 정육각형을 그리기 위해 로봇에게 내릴 명령문 생각하기, 활동 4에서는 로봇이라고 생각하고 격자에 활동 3에서 생각한 명령문에 따라 정육각형 그리기 활동을

제시하고 있다.

6학년의 경우, 활동 1~3을 제시하고 있으며 활동 1에서는 4가지의 수를 정렬하는 방법에 대해 생각하는 활동을, 활동 2에서는 교과서에 제시된 수를 정렬하는 방법을 친구에게 설명하는 활동을, 활동 3에서는 8가지의 수를 오름차순으로 정렬하는 활동을 제시하고 있다.



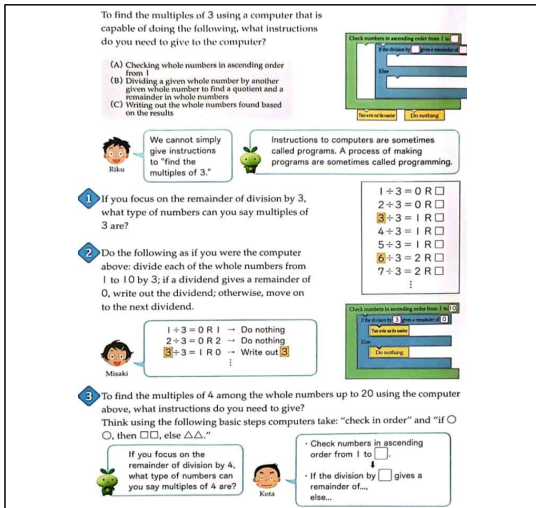
[그림 5] 5학년 교과서 구성(Isoda et al., 2020e, p. 144)



[그림 6] 6학년 교과서 구성(Isoda et al., 2020f, p. 216)

Tokyo Shoseki의 경우, 5학년 1학기에는 공배수 구하기, 5학년 2학기에는 정다각형 그리기, 6학년은 수

정렬하기를 주제로 프로그래밍 교육 관련 특화 차시를 구성하였다([그림 7], [그림 8], [그림 9] 참조). 구체적으로 살펴보면 5학년 1학기 공배수 구하기의 경우 도입, 활동 1~3으로 구성되어 있으며 도입의 경우 프로그래밍을 이용하여 3의 배수를 찾는 방법과 사용할 수 있는 코드 블록 알기, 활동 1에서는 3으로 나눈 나머지가 얼마이면 그 수가 3의 배수가 될지 생각해보기, 활동 2에서는 10까지의 수 중 3으로 나누었을 때 나머지가 0인 수를 적도록 프로그래밍하기, 활동 3에서는 20까지의 수 중 4의 배수를 찾도록 프로그래밍하기를 제시하고 있다.

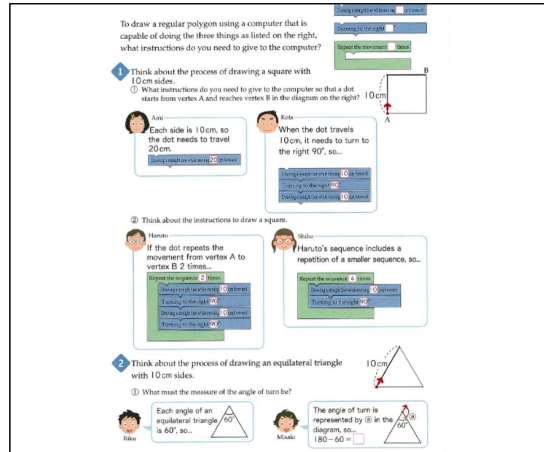


[그림 7] 5학년 1학기 교과서 구성(Fujii et al., 2020a, p. 126)

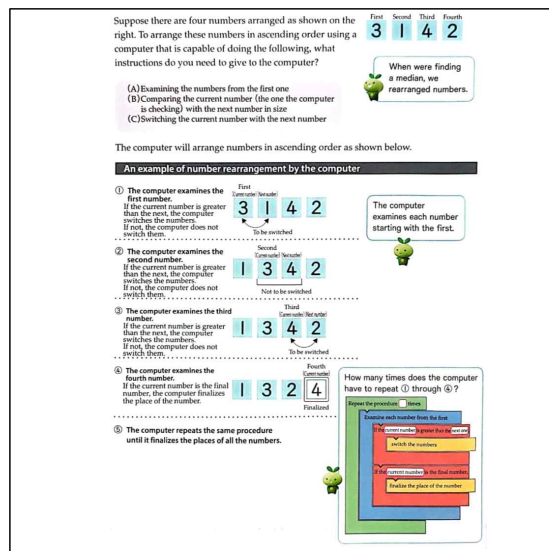
5학년 2학기의 정다각형 그리기의 경우 도입 및 활동 1~2로 구성되어 있으며, 도입의 경우 정다각형을 그리기 위해 필요한 명령어 이야기하기, 활동 1에서는 한 변의 길이가 10cm인 정사각형을 그리도록 프로그래밍하는 다양한 방법 이야기하기, 활동 2에서는 한 변의 길이가 10cm인 정삼각형을 그리도록 프로그래밍하는 과정 생각하기의 활동을 제시하고 있다.

6학년의 경우, 도입과 활동 1~4를 제시하고 있으며 도입에서는 컴퓨터로 수를 정렬할 수 있는 방법 알기, 활동 1에서는 자리가 확정된 수에 대해 이야기하기, 활동 2에서는 다시 단계를 진행할 때 자리가 확정될 수에 대해 이야기하기, 활동 3에서는 수가 오름차순으

로 정렬되는 이유 이야기하기, 활동 4에서는 제시된 그림을 이용하여 수를 오름차순으로 정렬하기 활동을 제시하고 있다.





[그림 8] 5학년 2학기 교과서 구성(Fujii et al., 2020b, p. 134)



[그림 9] 6학년 교과서 구성(Fujii et al., 2020c, p. 242)

5~6학년의 Gakko Toshō와 Tokyo Shoseki의 프로그래밍 교육 관련 특화 차시의 디지털 콘텐츠는 [표 9]와 같다. 우선 Gakko Toshō의 디지털 콘텐츠를 살펴보면 5학년의 정다각형 그리기 활동의 경우, 초기 화

[표 9] 5~6학년의 프로그래밍 관련 디지털 콘텐츠 구성

학년	Gakko Toshō		Tokyo Shoseki	
	디지털 콘텐츠 화면	설명	디지털 콘텐츠 화면	설명
5		<정다각형 그리기> 초기 화면의 왼쪽에 '앞으로 4cm 가기', '왼쪽으로 90° 돌기', '왼쪽으로 90° 돌고, 앞으로 4cm 가기' 블록으로 구성		<공배수 구하기> 왼쪽에는 프로그래밍 구현 화면을, 오른쪽에는 코드 블록을 제시하였으며, 공배수를 구하는 과정을 14단계로 제시함
		<정다각형 그리기> '앞으로 4cm 가기', '왼쪽으로 90° 돌고, 앞으로 4cm 가기' 블록을 이용하여 한 변의 길이가 4cm인 정사각형을 그린 장면		<정다각형 그리기> 왼쪽에는 격자와 흰색 선, 캐릭터가 있으며, 오른쪽에는 앞으로 4cm 가기, 90° 오른쪽을 회전, 90° 왼쪽으로 회전 블록이 제시됨
		<정다각형 그리기> '앞으로 4cm 가기', '왼쪽으로 45° 돌고, 앞으로 4cm 가기' 블록을 이용하여 한 변의 길이가 4cm인 정육각형을 그린 장면		<정다각형 그리기> 코드 블록으로 사각형을 그리는 장면, 이후 단계에서 삼각형, 오각형, 육각형, 자유 도형을 그리는 활동도 제시됨
6		<수 정렬하기> 왼쪽의 카드의 수를 정하는 칸과 수를 늘어놓는 버튼이 제시되며, 오른쪽에는 카드가 제시됨		<수 정렬하기> 왼쪽에서 플레이어의 키를 말하도록 프로그래밍하기 위해 말하는 블록과 선수의 번호 블록이 제시됨
		<수 정렬하기> 삽입 정렬과 버블 정렬 방식을 선택할 수 있으며, 정렬 방식을 누르면 자동으로 정렬됨		<수 정렬하기> 키 말하기 블록, 자리 바꾸기 블록, 가정(If) 블록, 반복 블록 등을 이용하여 오른쪽부터 키가 큰 순서대로 정렬하는 프로그래밍을 하는 화면
		<수 정렬하기> 카드 8장을 삽입 정렬을 선택한 후, 실행을 지속적으로 눌렀을 때 자동으로 정렬됨		

면의 왼쪽에는 ‘앞으로 \circ cm 만큼 가기’, ‘왼쪽으로 \circ° 돌기’, ‘왼쪽으로 \circ° 돌고, 앞으로 \circ cm 만큼 가기’ 블록이 제시되어 있다. 이 디지털 콘텐츠에서는 자유롭게 도형을 그리도록 제시하고 있다. 교과서에서 제시된 한 변의 길이가 4cm 인 정사각형은 ‘앞으로 4cm 가기’, ‘왼쪽으로 90° 돌기’, ‘앞으로 4cm 만큼 가기’ ‘왼쪽으로 90° 돌기’, ‘앞으로 4cm 가기’, ‘왼쪽으로 90° 돌기’, ‘앞으로 4cm 가기’ 블록을 이용하거나 ‘왼쪽으로 \circ° 돌고, 앞으로 \circ cm 만큼 가기’ 블록을 이용하여 ‘앞으로 4cm 가기’, ‘왼쪽으로 90° 돌고, 앞으로 4cm 만큼 가기’, ‘왼쪽으로 90° 돌고, 앞으로 4cm 만큼 가기’, ‘왼쪽으로 90° 돌고, 앞으로 4cm 만큼 가기’로 좀 더 효율적인 알고리즘으로 코딩할 수 있다. 마지막으로 교과서에 제시된 육각형의 한 각의 크기를 이용하여 ‘앞으로 4cm 가기’, ‘왼쪽으로 45° 돌고, 앞으로 4cm 만큼 가기’ 블록을 이용하여 한 변의 길이가 4cm 인 정육각형을 그릴 수 있다. 이때 사용된 블록은 스크래치와 거의 유사한 형태로 제시되었으며, 순차적 표현을 반복적 표현으로 바꿀 수 있도록 하여 표현의 효율성을 학습할 수 있도록 구성되어 있다.

6학년의 수 정렬하기 활동의 경우, 초기 화면의 왼쪽에는 카드의 수를 정하는 칸과 수를 넣어놓는 버튼이 제시되며, 오른쪽에는 카드가 제시된다. 원하는 카드의 수를 선택하면, 삽입 정렬과 버블 정렬 버튼이 제시되고 정렬 방식을 직접 선택할 수 있으며, 원하는 정렬 방식을 누르고 실행을 누를 때마다 한 단계씩 정렬이 된다. 실제 학생들이 수를 정렬하는 프로그래밍을 하지 않으며 수를 정렬하는 방식을 살펴보도록 구성되어 있다. 이는 각각의 정렬 방식을 알게 하는 의도로 생각해 볼 수 있다. 교과서의 경우 종이에 직접 수를 적고 이를 정렬해 보는 활동을 하는 반면, 디지털 콘텐츠에서는 카드 8장을 선택하고 삽입 정렬과 버블 정렬 중 하나를 선택한 후, 실행을 지속적으로 눌러 정렬을 실시한다. 학생들의 수를 직접적으로 정렬할 수는 없지만 이를 통해 두 정렬 방식의 차이점을 알 수 있을 것이다.

5~6학년의 Tokyo Shoseki의 프로그래밍 교육 관련 차시의 디지털 콘텐츠를 살펴보면, 5학년 1학기의 공배수 구하는 활동의 경우 왼쪽에는 프로그래밍 구현 화면을, 오른쪽에는 코드 블록을 제시하였으며, 공배수를 구하는 과정을 상세하게 14단계로 구성하였다. 1단

계는 캐릭터에게 이름을 말하도록 프로그래밍을 하는 것, 2단계는 상대의 이름을 말하도록 프로그래밍을 하는 것, 3단계는 1, 2, 3을 순서대로 말하도록 프로그래밍을 하는 것 등으로 시작하여 가정(IF) 블록을 사용하여 5와 3의 공배수를 찾는 프로그래밍을 하도록 유도하고 있다. 여기서는 스크래치와 매우 유사한 블록들이 제시되어 있으며, 동작 블록뿐만 아니라 제어 블록까지 사용되었다.

5학년 2학기의 정다각형을 그리는 활동의 경우 왼쪽에는 격자와 캐릭터가 있으며, 오른쪽에는 ‘앞으로 \circ cm 만큼 가면서 선을 그리기’, ‘ \circ° 오른쪽을 회전하기’, ‘ \circ° 왼쪽으로 회전하기’ 블록이 제시되었다. 정다각형을 그리는 활동의 경우 자유롭게 도형을 그리고 완성할 수 있었던 Gakko Tosho와 달리, Tokyo Shoseki에서는 총 8단계를 제시하였으며 1단계, 2단계에서는 사각형의 일부 그리기, 3단계에서는 단일 블록을 사용하여 한 변의 길이가 3cm 정사각형 그리기, 4단계에서는 반복 블록을 사용하여 한 변의 길이가 3cm 정사각형 그리기, 5단계는 한 변의 길이가 3cm 정삼각형 그리기, 6단계는 한 변의 길이가 3cm 정육각형 그리기, 7단계에서는 한 변의 길이가 3cm 정오각형 그리기, 8단계에서는 자유 도형을 그리기 활동을 제시하고 있다. 이때 사용된 블록은 스크래치와 유사하며 표현 역시 순차적 표현과 반복적 표현을 모두 사용할 수 있도록 구성되었다. 이는 Gakko Tosho의 경우 1학년년부터 프로그래밍을 학습했으나 Tokyo Shoseki의 경우 5학년 때 처음으로 프로그래밍을 접하므로 좀 더 상세하게 프로그래밍 단계를 제시한 것으로 보인다.

6학년에 제시된 수를 오름차순으로 정렬하는 활동의 경우, 9단계로 구성되어 있으며 초기 화면의 왼쪽에는 플레이어의 키를 말하도록 프로그래밍을 하기 위하여 키 말하기 블록과 선수의 번호 블록이 제시되어 있다. 마지막 단계에서는 키를 왼쪽부터 오름차순으로 정렬하기 위해 키 말하기 블록, 자리 바꾸기 블록, 가정(If) 블록, 반복 블록 등을 이용하여 프로그래밍을 하도록 하고 있다. 이때 정렬은 모두 버블 정렬에 대한 것이다. 이를 통해 Tokyo Shoseki는 학생들이 하나의 정렬 방식으로 스스로 프로그래밍을 하는 것에 초점을 두고 있으며, Gakko Tosho는 삽입 정렬과 버블 정렬의 방식 원리를 모두 경험해 보는 것에 초점을 두고 있음을 알 수 있다.

V. 결론 및 논의

본 연구는 일본 수학 교과서에 제시된 프로그래밍 교육 관련 특화 차시와 그에 따른 디지털 콘텐츠를 분석하였다. 주요 연구 결과를 토대로 다음과 같은 시사점을 제시하고자 한다.

첫째, 프로그래밍 교육을 통해 수학의 개념, 원리, 법칙에 대한 지도가 가능하며, 적절한 주제를 선정한다면 수학과 내용 영역 전체에서 적용이 가능할 것이다. 일본 수학 교과서에서는 점 이동하기, 공배수 구하기, 정다각형 그리기 등의 다양한 수학적 내용을 활용하여 프로그래밍 교육 관련 특화 차시를 제시하고 있다. 이러한 프로그래밍 교육은 수학의 개념, 원리, 법칙에 대한 학생들의 깊이 있는 이해를 도울 수 있다. 또한 2022 개정 수학과 교육과정에 따른 내용 영역별로 살펴보면 수와 연산, 도형과 측정, 변화와 관계, 자료와 가능성의 모든 내용 영역이 일본 수학 교과서의 프로그래밍 관련 특화 차시에 제시되어 있었다. 이처럼 수학의 개념, 원리, 법칙에 대한 이해와 논리적 사고를 도울 수 있으므로 우리나라 초등학교 수학 교과에서도 수학과 전 내용 영역과 연결하여 프로그래밍 교육이 가능할 것이다.

둘째, 수학을 통해 프로그래밍 교육을 위한 기본적인 요소들을 학습할 수 있으며, 수학 교과에서의 연결 역량을 키울 수 있다. 일본 수학 교과서의 분석 결과 제시된 활동들을 통해 프로그래밍 교육을 위한 다양한 수학적 요소를 모두 학습할 수 있었다. 구체적으로 프로그래밍 관련 특화 차시에 제시된 모든 활동에서 순서, 규칙, 변수, 가능성, 함수, 선택, 논리, 알고리즘 요소 중 4개 이상의 여러 수학적 요소를 통합적으로 학습할 수 있었다. 이러한 요소들은 프로그래밍을 하기 위한 기초적이며 필수적인 요소들이므로 수학이 프로그래밍을 학습하기 위한 기본적인 교과의 역할을 할 수 있을 것이다. 다만 연구 결과 같은 주제에 대한 활동이더라도 활동 구성에 따라 포함된 수학적 요소에 많은 차이를 보였으므로 다양한 수학적 요소가 포함될 수 있도록 활동을 적절히 구성해야 할 것이다.

셋째, 프로그래밍을 통해 일련의 문제 해결 과정을 경험할 수 있으며, 문제 해결 역량을 향상시키는데 도움을 줄 수 있다. 프로그래밍은 해결해야 할 문제를

이해하고 계획하며 이를 실제 프로그래밍하여 디버깅을 통해 반성할 수 있다. 이는 수학에서의 문제 해결의 단계와 동일하며 적절한 과제를 선정하여 수학적 내용을 학습하면서 프로그래밍을 실시한다면 일련의 문제 해결 과정 전체를 경험할 수 있는 것이다. 또한 프로그래밍은 교사의 직접적인 피드백이 없어도 지속적인 실행을 통해 문제에서 잘못된 부분을 찾고 수정할 수 있어 학습에 좀 더 효과적이다. 이에 프로그래밍을 활용할 수 있는 적절한 과제를 구성하여 문제 이해, 계획 수립, 계획 실행, 반성의 단계를 거쳐 지도한다면 문제해결 역량도 키울 수 있을 것이다.

넷째, 프로그래밍에 대한 적절한 교육을 위해서는 디지털 콘텐츠를 제작하는 것이 좀 더 효과적일 수 있으며, 학생들이 프로그래밍 도구를 적절히 활용하기 위해서는 세분화된 단계로 제공해야 할 것이다. 일본 수학 교과서의 경우 2종의 교과서 모두 해당 활동에 따른 디지털 콘텐츠를 직접 제작하여 배포하였으며, 모두 교과서에 제시된 내용보다 상세히 제시하였다. 또한 수학적 내용뿐만 아니라 디지털 콘텐츠를 적절히 활용할 수 있도록 사용 방법에 대한 안내도 제시하였다. 따라서 우리나라에서 프로그래밍 교육을 실시할 경우, 효과적인 교육을 위해서 교과서뿐만 아니라 디지털 콘텐츠도 함께 제시해야 하며 디지털 콘텐츠는 학습 수준에 맞추어 단계로 제시되도록 개발해야 할 것이다.

다섯째, 1학년부터 수학에서 프로그래밍 지도가 가능하며, 지도 내용에 따른 보충 심화 학습이 이루어질 수 있다. 일본 수학 교과서의 경우 1학년부터 로봇을 목적지도 이동시키기에 대한 프로그래밍을 지도하고 있으며, 자체 제작한 간단한 코드 블록을 통해 학생들이 이용하기 쉽게 구성되어 있다. 우리나라의 경우 점의 이동은 3~4학년군에서 학습하므로 일본 수학 교과서에 제시된 내용과 동일하게 구성하기는 어려우나 우리나라 교육과정에 적합한 주제를 찾아 구성한다면 충분히 지도가 가능할 것으로 보인다. 또한 프로그래밍 교육 관련 디지털 콘텐츠의 경우 교과서와 달리 지면의 제한이 없어 다양한 활동이 제시될 수 있다. 실제로 일본에서는 교과서에 제시된 활동보다 디지털 콘텐츠에 더욱 많은 활동을 제시하고 있었다. 따라서 다양한 수준의 디지털 콘텐츠를 제시한다면 보충 심화 학습도 가능할 것으로 보인다.

마지막으로 프로그래밍 교육을 통해 기초 소양 측면에서 디지털 소양을 기를 수 있을 것이다. 2022 개정 수학과 교육과정에 제시된 바와 같이 학생들에게 프로그래밍 교육과 연계할 수 있는 적절한 수학 내용을 선택하여 효율적인 교수·학습이 이루어지게 한다면 디지털 소양을 함양할 수 있을 것이다. 디지털 소양의 의미는 디지털 기술을 잘 다루는 능력에서 최근에는 디지털을 활용하여 디지털 세계에서 서로 공존하는 삶을 살아갈 수 있는 능력으로 그 의미가 확장되고 있다(강현영, 2022). 학생들은 프로그래밍 교육을 통해 디지털 기술을 잘 다룰 수 있게 되며, 디지털 세계에서 문제를 해결하기 위해 스스로 프로그래밍을 할 수 있으며 자신이 만든 프로그램을 다른 사람과 공유하는 삶을 살 수 있다. 이처럼 프로그래밍 교육은 디지털 세계에서 학생들이 자연스럽게 삶의 문제를 해결할 수 있도록 하는 디지털 소양을 키우는 하나의 방법이 될 수 있을 것이다.

본 연구는 일본 수학 교과서에 제시된 프로그래밍 관련 특화 차이를 다양한 측면에서 분석하고자 하였다. 이를 통해 우리나라 수학 교과에서 프로그래밍 교육을 적용할 때 기초적인 자료로 활용되길 기대한다.

참 고 문 헌

강상욱, 강승호, 김일문, 김철웅, 김현중, 김현태, 박상언, 박재우 외 6인(2022). Excel·Spss·R로 배우는 통계학 입문(제4판). 자유아카데미.
 강현영(2022). 미래사회 대비를 위한 디지털·AI 소양 교육을 위한 고찰. 한국디지털콘텐츠학회논문지, 23(6), 1067-1075.
 교육부(2015). 소프트웨어 교육 운영 지침. 교육부.
 교육부(2020). 정보교육 중합계획(2020년~2024년). 교육부.
 권미선(2023). 핀란드 1~2학년 초등 수학 교과서에 제시된 프로그래밍 교육을 위한 수학적 요소 분석. 학교수학, 25(3), 385-406.
 권오남, 이경원, 오세준, 박정숙(2021). <인공지능 수학> 교과서의 ‘관련 학습 요소’ 반영 내용 분석. 수학교육논문집, 35(4), 445-473.
 김대수(2017). 소프트웨어와 컴퓨팅 사고. 생능출판사.

김철, 김갑수, 김현배, 정영식, 정인기(2016). 초등 소프트웨어 교육과정 표준 모델 연구. 커넥트재단.
 나귀수, 박미미, 김동원, 김연, 이수진(2018). 미래 시대의 수학교육 방향에 대한 연구. 수학교육학연구, 28(4), 437-478.
 박성호, 김은정, 문경희, 박미현, 양자영(2021). 인문·사회·예체능 계열 전공자를 위한 컴퓨팅 사고와 인공지능. 부산대학교출판 문화원.
 박주미(2017). 컴퓨팅 사고력을 키우는 이산수학(개정판). 한빛아카데미(주).
 이승우(2020). 프랑스 중학교 수학 교육과정 분석: ‘알고리즘과 프로그래밍’ 영역을 중심으로. 학교수학, 22(1), 125-159.
 한지근(2018). 컴퓨팅 사고력. 배움터.
 文部科學省 (2017). 小學校學習指導要領(平成29年告示)解説 算數編. 文部科學省.
 BArt, 장재용(2017). 코딩시대. 클라우드북스.
 Fujii, T., Majima, H., & New mathematics Editorial Board (2020a). *New mathematics 5A for elementary school*. Tokyo Shoseki Co., Ltd.
 Fujii, T., Majima, H., & New mathematics Editorial Board (2020b). *New mathematics 5B for elementary school*. Tokyo Shoseki Co., Ltd.
 Fujii, T., Majima, H., & New mathematics Editorial Board (2020c). *New mathematics 6 for elementary school*. Tokyo Shoseki Co., Ltd.
 Isoda, M., Murata, A., Rafiepour, A., Matsuzaki, A., Esteley, C., Bautista Jr, G. P., ... Widjaja, W. (2020a). *Mathematics for elementary school 1st grade, Vol. 2*. Gakko Tosho Co., Ltd.
 Isoda, M., Murata, A., Rafiepour, A., Matsuzaki, A., Esteley, C., Bautista Jr, G. P., ... Widjaja, W. (2020b). *Mathematics for elementary school 2nd grade, Vol. 2*. Gakko Tosho Co., Ltd.
 Isoda, M., Murata, A., Rafiepour, A., Matsuzaki, A., Esteley, C., Bautista Jr, G. P., ... Widjaja, W. (2020c). *Mathematics for elementary school 3rd grade, Vol. 2*. Gakko Tosho Co., Ltd.
 Isoda, M., Murata, A., Rafiepour, A., Matsuzaki, A., Esteley, C., Bautista Jr, G. P., ... Widjaja, W. (2020d). *Mathematics for elementary school 4th*

- grade, Vol. 2* Gakko Tosho Co., Ltd.
- Isoda, M., Murata, A., Rafiepour, A., Matsuzaki, A., Esteley, C., Bautista Jr, G. P., ... Widjaja, W. (2020e). *Mathematics for elementary school 5th grade, Vol. 2* Gakko Tosho Co., Ltd.
- Isoda, M., Murata, A., Rafiepour, A., Matsuzaki, A., Esteley, C., Bautista Jr, G. P., ... Widjaja, W. (2020f). *Mathematics for elementary school 6th grade, Vol. 2* Gakko Tosho Co., Ltd.
- Kafai, Y. B., & Burke, Q. (2014). *Connected code*. MIT Press. 최윤희 역(2017). 코딩 혁명: 아이들이 프로그래밍을 배워야 하는 이유. (주)디뷰.
- Liukas, L. (2015). *Hello ruby: Adventures in coding*. Feiwei & Friends Book. 이지선 역(2016). 헬로 루비: 코딩아 놀자. 길벗어린이(주).
- YTN 사이언스(2020). 미래 사회의 또 다른 언어, 코딩. <https://science.ytn.co.kr/program/view.php?mcd=1213 &key=202212091235495459>
- Wing, J. M. (2006) Computational thinking. *Communication of the ACM, 49(3)*, 33-35.

Analysis of Japanese elementary school mathematics textbooks and digital contents on programming education

Kwon, Misun

Shinpoong Elementary School

E-mail : annietj@naver.com

This paper analyzed the programming education specialized lessons presented in two types of elementary school mathematics textbooks according to the revised Japanese curriculum in 2017. First, this paper presented in detail how each activity is connected to Korean mathematics areas, what elements of mathematics can be learned through programming education, how each activity is structured, and how the actual programming according to the textbook activities is structured. In Japanese textbooks, geometry and measurement areas were presented the most among Korean mathematics content areas, and mathematical elements such as sequences, rules, and algorithms were most implemented for learning. Digital learning tools that make up actual programming present more elements than those presented in the textbooks and are presented in great detail so that students can do actual programming. Lastly, in blocks, motion, control, and calculation blocks were used a lot. Based on these research results, this study provides implications when conducting programming-related education in Korea.

* 2000 Mathematics Subject Classification : 97U20

* Key Words : Japanese mathematics textbook, programming education, programmatic thinking, mathematical elements for programming education, digital literacy