수학·정보 융합교육을 위한 코딩과 연계한 교수학습 자료 개발 연구

신기철 · 서보억[1*]
한국교원대학교 · [1]충남대학교

A Study on Development of Teaching & Learning Materials related to Coding for Convergence Education Integrating Mathematics and Information

Gicheol Shin · Boeuk Suh[1*]
Korea National University of Education · [1]Chungnam National University

Abstract : This study, as an attempt to integrate mathematics and information for convergence education, was conducted to develop teaching-learning materials on mathematics education combined with coding education, which has recently been emphasized. We chose the subject of digital signature for coding education, and used SageMath as a coding program. In this study, we overview mathematics used in the elliptic curve digital signature algorithm, one of the many methods for digital signature, and developed the teaching-learning materials on the algorithm for mathematics education integrated with information education based on coding. The elliptic curve digital signature algorithm utilized in transactions of Bitcoin, which many people recently are interested in, is a good example, showing students that mathematics is applied to problem-solving in the real world and provides an optimal environment for implementation by coding. Accordingly, we expect that a class on algorithm will provide a specific teaching-learning program to achieve the goal of integrated mathematics education. By comprehensively considering the opinions of mathematicians, mathematics teachers and mathematics education experts, we expect that the teaching-learning program will be realized as a meaningful class in science high schools, high school's math clubs, and 'number theory' class in colleges.

keywords : mathematical modeling, convergence education, mathematics and information subject, coding education, mathematics teaching & learning materials, digital signatures

Ⅰ. 서론

최근 4차 산업혁명의 도래와 함께 중등학교 예비교사 교육을 위해 수학적 모델링이 강조되고 있다. 이러한 시대적 흐름으로 인해 예비교사교육, 고등학교 수학교육 등에서도 여러 학제간의 어우러짐에 의한 융합이 강조되고 있다. 또한 교육과정에 포함된 주제 또는 학생들이 관심을 가질만한 주제를 택하여 수학과 여러 학제의 융합을 위한 수업 자료를 개발하는 것은 급변하는 4

*교신저자 : 서보억 (eukeuk@cnu.ac.kr)
**이 논문은 2017년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No.2017R1E1A1A03070849).
***2019년 02월 28일 접수, 2019년 03월 25일 수정원고 접수, 2019년 03월 25일 채택
http://dx.doi.org/10.21796/jse.2019.43.1.17

차 산업혁명의 시대에 적합한 교육활동 중의 하나이다. 특히 최근 이슈가 되고 있는 4차 산업혁명의 도래로 인해, 우리 주변의 모든 영역이 사물인터넷(IOT), 빅테이터(Big Data), 인공지능(Artificial Intelligent, AI) 등과 융합하여 사회 전반에 혁신적 변화가 일어나고 있다. 이러한 급격한 변화는 모바일(Mobile)기기, 휴대폰, 3D프린터, 드론과 같은 물리적 하드웨어뿐 아니라, 이를 제어하고 통제하며 운영하는 소프트웨어까지 중요해지고 있다. 독일의 자동차 회사인 '메르세데스 벤츠'의 CEO는 '자동차는 이제 가솔린으로 움직이는 것이 아니라, 소프트웨어로 움직이고 있다'라고 말할 만큼 소프트웨어의 중요성이 커졌다. 또한 Lee (1996)는 소프트웨어와 같은 컴퓨터 프로그래밍 교육을 통해서 논리적 사고력, 문제해결력 등의 고등사고능력을 길러주어야 한다고 강조하고 있다.

이러한 소프트웨어와 관련된 역량이 많이 있지만, 새로운 세대에 가장 중요하게 여겨는 것은 바로 코딩(coding)이다. 단순히 프로그램을 응용하고 활용하는 차원이 아니라, 스스로 무엇인가를 창조하고 개발하는 수단이 새로운 세대에 가장 중요한 역량이기 때문이다. 이러한 이유로 인해 서구의 많은 나라들은 앞 다투어 코딩 교육을 강화하고 있다. 미국의 경우, 대통령이 직접 나서 '코딩교육은 미국의 미래에 중요하다.'라고 하였고, 영국은 체계적인 코딩교육을 실시하기 위해 영국교육부(British Department for Education, 2013)가 앞장서고 있으며, 핀란드(CSTA, 2011)는 CT(computational thinking)를 바탕으로 하는 프로그래밍 코딩교육을 교육과정에 포함시켰다(Chang, 2017). 실제로 알파고(Alpha Go)의 등장 이래로, 우리나라에서도 인공지능에 대한 관심이 증가되었고, 2015개정 교육과정부터 '정보'를 정규교육과정에 포함시켰다. 이를 통해 볼 때, 우리나라도 '코딩교육'이 교육계의 중요한 이슈가 되었다(Kwon, 2018). 실제로 2018학년도부터 코딩교육이 필수화되었고, 이에 대한 효율적인 운영을 위해 연구학교 1200개를 선정하여 운영하였다(MOE, 2017). 그런데, 4차 산업혁명시대로 급부상하고 있는 코딩교육은 수학교육과는 전혀 무관하지 않다는 점이다(Baek, 2018). 지금까지 수학교육활동에서의 코딩교육은 주로 프로그래밍 교육이라 판단하였고, 이로 인해 코딩교육의 도구로 로고(Logo), 베이직(Basic), C언어 등이 활용되고 연구되어 왔다. 이 연구의 핵심은 이러한 프로그램을 사용한 문제해결력 신장에 초점이 있었다.

이처럼 IT가 발달하고 컴퓨터 프로그래밍 및 코딩교육이 강조되면서, 다양한 교과가 융합한 STEM 교육프로그램의 개발과 더불어 코딩 및 프로그래밍 교육을 강조하는 교육과정 개발의 필요성이 지속적으로 제기되고 있다. 또한 단순한 문제해결력 신장을 위한 도구로써의 코딩교육이 아니라, 수학과 타교과 특히 정보교과와 융합하는 과정에서 특정 개념을 추상화하여 수학을 학습하는 도구로써의 코딩교육 및 수학적 모델링 활동을 통해 정보교과의 현안을 해결하는 도구로써의 코딩교육에 대한 필요성이 제기되고 있다.

이러한 연구의 필요성에 따라서 본 연구에서는 수학적 모델링의 구체적인 사례를 개발하기 위해, 코딩교육과 가장 관련이 깊은 정보교과와 수학이 융합한 교수·학습 자료의 개발을 연구의 목적으로 설정하였다. 이러한 연구의 목적을 달성하기 위해 현재 정보 분야에게 가장 이슈가 되고, 흥미로운 주제를 선정하여 연구를 진행하였다. 바로 비트코인(bitcoin)과 이 비트코인의 결재와 관련된 주제이다. 현대 사회에서 전자거래는 일상이 되었고, '클라이언트-서버' 체계에서의 전통적인 전자거래뿐 아니라 비트코인의 거래와 같은 'P2P (peer-to-peer)' 환경에서의 전자거래도 활발하게 일어나고 있다. 특히 최근에는 비트코인이 주요 화두가 되고 있다. 비트코인은 블록체인 환경에 기반을 두고 있고, 이러한 전자화폐에서 수학이 필수적인 역할을 한다. 즉, 신뢰할 수 있는 전자거래 기술에서 수학은 필수적이다. 실제로, 전자거래에서 이용되는 RSA 알고리즘이나 타원곡선 전자서명 알고리즘 등은 수학의 응용이다. 특히, 비교적 최근에 사회의 큰 관심을 받은 비트코인의 거래는 학생들이 문제 상황에 깊게 몰

입할 수 있는 수학과 정보의 융합 수업의 좋은 소재라고 할 수 있다. 수학에 관심을 가진 학생이라면 관심을 가질만한 그러한 주제임에도 불구하고 전자화폐나 거래에서 어떠한 수학이 어떻게 활용되는지를 소개하려는 시도는 찾아보기 어렵다.

이에 본 연구에서는 수학적 모델링을 활용한 사범대학 예비교사 교육을 위한 '정수론' 강좌, 과학고등학교 또는 일반계 고등학교 수학 동아리에서 구현 가능한 수학과 정보교과의 융합수업을 코딩교육과 연계하여 개념을 획득할 수 있는 수업모형 및 교수·학습 자료 개발을 연구의 내용으로 설정하였다. 수업모형 및 이러한 수업을 통해 실생활에 실제로 활용되는 수학에 대한 진정한 모습을 깨닫고, 수학에 대한 안목을 높일 수 있는 수학교육 실현에 기여하고, 코딩교육 기반 수학교육의 실현을 기대한다.

Ⅱ. 이론적 배경

1. 수학과 융합인재교육

코딩을 기반으로 수학교과와 정보교과를 융합하는 방법 중, 새로운 교육과정을 개발하지 않고 실현할 수 있는 최적의 방법은 STEAM의 맥락에서 융합하는 것이다. 여기서 STEAM이란 과학(Science), 기술(Technology), 공학(Engineering), 예술(Arts), 수학(Mathematic)의 줄임말로 미국에서 시작된 STEM에서 유래되었다(Sanders, 2008). Baek et al. (2010)은 STEM을 기반으로 하는 교육은 현재 교육분야에서 가장 중요한 핵심적인 주제중의 하나이며, 특기 수학 및 과학교육을 개혁하기 위한 중심에 위치하고 있다. 실제로 미국의 과학재단(NSF, 2010; National Science Foundation)에서는 학문적인 접근으로 정규 학교 교육과정에서 공학과 기술을 통합함으로써 수학, 과학 수업을 혁신할 수 있다고 규정하고 있다. 또한 우리나라에서는 STEAM 교육을 '융합인재교육'으로 명명하였고, Baek *et al.* (2010)는 융합인재교육을 '다양한 학문 분야에서의 융합적인 내용을 감성적 체험(Emotional Touch)과 창의적 설계(Creative Design) 등으로 경험함으로써 과학기술 및 공학, 수학 등과 관련된 다양한 분야의 융합 지식, 융합적 과정, 학문적 본성에 대한 흥미와 이해를 높여 종합적이고 창의적이고 문제를 해결할 수 있는 융합 소양(STEAM Literacy)을 갖춘 인재의 양성'으로 정의하고 있다. 더불어, 융합인재교육을 통해 갖추어야할 핵심역량으로는 4C 즉, '창의(Creativity)', '소통(Communication)', '융합(Convergence)', '배려(Caring)'를 제시하였고(Baek *et al.*, 2010), 교육부(MOE, 2015)에서는 '광범위한 기초 학문 지식을 바탕으로 다양한 학문 분야의 지식, 경험, 기술 등을 융합적으로 사용하여 새로운 지식을 창출하는 창의적 사고 역량'으로 규정하여, 적극적으로 융합인재교육을 강조하였다.

2. 정보처리 역량과 CT(Computational Thinking)

2015개정 수학과 교육과정(MOE, 2015)에서 강조하는 교과 역량 중 정보 처리 능력은 '다양한 자료와 정보를 수집·정리·분석·활용하고 적절한 공학적 도구나 교구를 선택·이용하여 자료와 정보를 효과적으로 처리하는 능력'을 의미한다. 즉, 업무를 처리하고 삶을 영위하는 과정에서 직면하는 문제를 해결하기 위하여 다양한 정보와 자료를 수집, 분석, 평가, 분류, 조직함으로써 자료와 정보에 내재된 의미를 올바르게 파악하고, 컴퓨터 등 적절한 매체를 활용하여 정보와 자료를 효과적으로 처리함으로써 합리적으로 문제를 해결할 수 있는 역량으로 볼 수 있다. 정보 처리 능력의 하위 요소와 그 의미, 그리고 이를 구현하는 기능은 Table 1과 같다.

최근 미국 등을 중심으로 강조되고 있던 CT 역량은 국내에서도 매우 높은 관심을 보이고 있다. CT는 궁극적으로는 인간의 사고와 컴퓨터의 정보처리능력을 통합한 사고를 의미하는데, 이는

Table 1. Sub-component and meaning of information processing ability

하위 요소	의미
자료와 정보 수집	실생활 및 수학적 문제 상황에서 적절한 자료와 정보를 탐색 및 생성하여 수집하는 능력
자료와 정보 정리 및 분석	수집한 자료와 정보를 목적에 맞게 분류, 정리, 분석, 평가하는 능력
정보 해석 및 활용	분석한 정보에 내재된 의미를 올바르게 파악하여 해석, 종합, 활용하는 능력
공학적 도구 및 교구 활용	수학적 아이디어와 개념을 탐구하고 문제를 해결하는 데 적합한 공학적 도구 및 교구를 선택하고 이용하는 능력

단편적인 학습에서 벗어나 복합적 사고로 나가는 수단으로, 창의적 문제를 해결하는 핵심 능력으로 주목 받고 있다. 또한, CT는 컴퓨터 과학적 개념 원리가 다른 여러 전문 영역에서 문제해결의 핵심적 도구로 되어가고 있기 때문에 이러한 개념이 등장하게 되었고, 정규 교육에서 수용하여야 할 필요성이 구체적으로 부각되고 있다 (Moon, 2013).

3. 수학교육에서 코딩과 SageMath 프로그램

코딩의 사전적인 의미는 프로그램의 명령문을 사용하여 컴퓨터 작업의 흐름에 따라 프로그램을 작성하는 것이다. 코딩은 컴퓨터와 대화하는 일종의 언어이자 도구로서 컴퓨터가 알아들을 수 있는 형태의 언어로 문장을 작성하는 것을 의미한다. 본 연구에서 코딩은 특정 프로그래밍 언어를 사용하여 프로그램밍을 완성하는 것을 말한다. 본 연구에서 사용하는 프로그래밍 언어는 SageMath 프로그램이다. 본 연구에서 코딩은 컴퓨터 프로그램 작성 단계중 하나로 프로그래밍 언어를 사용하여 프로그램을 기술하는 것으로, 본 연구에서는 수학적 개념 및 정보관련 내용 중 전자서명 관련 개념을 획득해 가는 과정에서 프로그래밍 언어를 사용하여 해당 개념을 추상화하는 과정을 말한다. 따라서 코딩 학습은 단순히 코딩 방법이나 기능만을 학습하는 것이 아니라, 문제의 발견 및 표현, 효율적이고 효과적인 알고리즘의 제작, 사고의 수정 등의 고등 사고 능력을 계발하는 것을 포함한다.

코딩과 같은 프로그래밍의 교육적 의의는 여러 가지로 정리할 수 있다. Kang(2004)은 이에 대해 세 가지로 요약하고 있다. 첫째, 학습자의 논리적 사고력과 문제해결능력과 같은 고등인지능력을 향상시킬 수 있다. 둘째, 프로그래밍을 통해서 학습자는 단순히 지식을 수용하기만 하는 것이 아니라, 스스로 자신만의 지식의 체계를 설계하고 구축하는 입장에 놓이게 한다. 셋째, 국가적인 측면에서 고려할 때, 소프트웨어 개발 선진국으로 도약할 수 있는 기반을 마련할 수 있다.

특히 수학교과와 코딩의 연관성은 매우 중요하게 다루어지고 있다. 왜냐하면, 컴퓨터 프로그래밍과 수학적 사고는 깊은 연관이 있기 때문이다. 실제로 수학교육에서 컴퓨터를 이용하면 기초적인 프로그래밍의 알고리즘을 이해하게 된다. 또한 Park(2002)은 추상적인 수학적 개념과 실제적인 표상들 사이의 연관성이 지식망을 형성하여 효과적인 학습 환경의 핵심적인 요소가 된다고 언급하였다. 이러한 이유로 인해 순수 수학 전공과 컴퓨터 과학의 연계가 강화되어 컴퓨터 프로그래밍이 대학교의 수학 전공 교육과정과 연계되고, 포함되는 사례가 늘고 있다. 또한 외국의 사례를 비롯하여 우리나라의 경우에도 이미 LOGO 등의 교육용 프로그래밍 언어가 교과서에 등장하듯이 수학 교과 교육에 컴퓨터 교육을 통합해 가고 있다. 원천적으로 컴퓨터과학의 발전이 순수 수학 학문에 기반을 두고 있듯이, 컴퓨터 프로그래밍 역시 수학적 사고와 밀접한 관련이 있다고 하겠다(Park, 2006). 또한 Park(2002)은 수학교육에서 컴퓨터를 이용하면 기초적인 프로그래밍 알고리즘을 이해하게 되고, 추상적인 수학적 개념과 실제적인 표상들의 연관성이 지식망을 형성하여 효과적인 학습 환경을 구축할 수 있다고 지

적하고 있다. Park(2006) 역시, 컴퓨터 프로그래밍은 수학적 사고와 밀접한 관련이 있다고 말하고 있다. 게다가, 2015개정 교육과정 정보교과에 코딩교육이 초중등학교에서 의무화됨에 따라 코딩교육에 관한 다양한 연구들도 같이 이루어지고 있다. 특히, 기존의 전공자들이 다루는 프로그래밍이 아닌 비전공자들도 자신의 필요 때문에 편하게 다룰 수 있는 각종 코딩에 대한 교육들이 전 영역에 걸쳐 일어나고 있다.

본 연구에서 코딩을 위해 사용하는 프로그램은 SageMath이다. SageMath는 SAGE(System for Algebra and Geometry Experimentation)라고 말하며, 대수학, 조합론, 그래프 이론, 수치해석, 정수론, 미적분학, 통계학 등 수학의 많은 분야를 포괄하는 특징을 가진 컴퓨터 대수 시스템을 의미한다. SageMath의 첫 번째 판은 2005년에 GNU General Public License 버전 2의 조건에 따른 무료 오픈 소스 소프트웨어로서 발매되었으며, 유료 소프트웨어인 Mathematica나 MATLAB의 대안으로 창조되었다. SageMath는 구조나 프로그램의 기본적인 기능은 오픈소스인 Python이다. 따라서 Python언어의 문법을 학습한 학생들이라면 SageMath의 사용법을 빠른 시간 내에 습득할 수 있으며, 그렇지 않은 학생들의 경우도 웹사이트(http://doc.sagemath.org/html/en/tutorial/)에서 제공하는 지침서를 따라서 주제별로 사용법을 학습할 수 있다.

4. 개념이해와 코딩: 추상화에 대한 RBC 이론

전통적인 관점의 추상화에 대한 대안으로 맥락과 상황을 강조하는 상황적 추상화가 새롭게 대두되고 있다. 추상화의 과정은 구체적인 예들로부터 추상화가 이루어지는 한 방향의 수준 상승 과정이 아님을 지적하는 관점이다. 추상화 과정이 계속될수록 맥락이나 상황이 고려되지 않는 탈맥락화 과정이 아니라 구체적인 예나 대상에서 추상으로, 다시 추상에서 구체적인 예나 대상으로 사고의 중심이 이동해가며 서로를 바탕으로 추상화가 변증법적으로 발달하며 이루어진다는 것이다. Noss, Hoyles & Pozzi(2002)는 맥락에 기초한 추상화의 중요성을 강조하면서 컴퓨터 프로그래밍 즉 코딩을 강조하였다. Hershkowitz, Schwarz, Dreyfus(2001)는 Noss와 Hoyles의 상황적 추상화 과정과 Davydov 활동 이론을 바탕으로 오랜 기간 동안 학생들과의 인터뷰 자료를 분석하여 추상화 과정을 연구하였다. 추상화에 대한 연구는 매우 어렵고 도전적인 과제이다. 추상화는 정신 활동을 의미하며 이는 쉽게 파악하거나 관찰하기 어렵다. Hershkowitz, Schwarz & Dreyfus(2001)는 추상화의 발생을 역동적으로 포개어지는 인식론적 행동 모델인 RBC모델로 설명하였다. 여기서 인식론적 행동이란 지식이나 개념, 절차 등을 사용하거나 구성할 때 나타나는 정신적 행동을 의미한다. 본 연구에서는 수학 코딩학습의 계획 단계인 사고 실험(thought experiment) 단계와 실행 단계인 코딩 과정에서 어떤 수학적인 지식이나 개념, 절차 등을 사용하거나 구성하는 정신적 행동을 의미한다. 이러한 인식론적 행동중 상황적 추상화 과정에 나타나는 인식론적 행동은 구조 인식(Recognizing), 구조 확립(Building-with), 구조 구성(Constructing) 세 가지로 세분화하였다. 첫째, 구조 인식(Recognizing)이다. 구조 인식은 학생들이 이미 인식하거나 잘 알고 있는 수학적 대상이나 구조, 지식, 개념, 방법, 절차 등을 현재 주어진 문제 상황에서 인식하고 이를 그대로 적용하는 행동이다. 구조 인식(Recognizing) 행동은 아니라 이전에 경험하고 알고 있는 것과 다른 상황에서 이전에 사용한 구조를 인식하는 행동이다. 익숙한 수학적 구조에 대한 인식은 학생들이 그 구조가 주어진 수학적 상황에 내재되어 있다는 것을 인식할 때 발생한다. 구조 인식 행동은 항상 그런 것은 아니지만 경험적 사고 수준에서 주로 발생한다. 둘째, 구조 확립(Building-with)이다. 구조 확립은 학생들이 이미 인식하거나 잘 알고 있는 수학적 대상이나 구조, 지식, 개념, 방법, 절차 등을 현재 주어진 문제 상황에서 모으거나 결합하여 적용하는 행동이다. 구조 확립(Building-

with) 행동은 주어진 목표를 달성하기 위해 구조적 요소를 결합하는 인식론적 행동이다. 구조 확립 과정에서 학생들은 새롭고, 보다 복잡한 구조적 지식을 사용하지는 않는다. 구조 확립은 학생들이 문제를 해결하고, 상황을 이해하고 설명하며, 과정을 반성하는 등의 목표를 달성하기 위해 가장 많이 발생한다. 이러한 목적을 위해 학생들은 전략, 공식 또는 정리에 의존한다. 구조 확립 행동은 교사가 자원을 상기시키고, 학생들이 그 아이디어를 떠올릴 때 일어날 수 있다. 셋째, 구조 구성(Constructing)이다. 구조 구성은 학생들이 이미 인식하거나 잘 알고 있는 수학적 대상이나 구조, 지식, 개념, 방법, 절차 등을 기반으로 현재 주어진 문제 상황에서 기존과는 다른 새로운 구조나 방법, 절차 등을 구성하는 행동이다. 구조 구성(Constructing) 행동은 세 가지의 인식론적 행동 중 가장 중요한 행동이다.

일반적으로 사람들은 새로운 방법, 전략, 개념을 구성한다. 새로운 구조가 마음속으로 들어왔을 때, 이것은 구성요소들로부터 더 간단한 구조로 인식되거나 연결되어야 한다. 구조 구성 행동은 새로운 구조를 만드는 단계로 지식의 수직적 재조직을 의미한다. 이 행동은 추상화의 핵심적인 행동으로 다른 두 행동보다 쉽게 일어나지 않는다. 구조 구성을 관찰하는 것은 방법론적 문제이다. 왜냐하면 구성은 상대적으로 매우 드물게 드러나기 때문이다. 새로운 구조 구성과 구조 확립의 중요한 차이는 목표 달성을 위한 동기와 그 동기를 실행하는 행동의 참신성에 있다고 할 수 있다. 구조 구성 행동은 문제를 풀거나 해답을 정당화하거나 가설을 만드는 등 목표를 달성하기 위해 학생들은 새로운 수학적 구조를 사용한다. 구조 확립 행동에서 학생들은 목표를 달성하기 위해 기존의 구조들을 결합한다. 학생들이 가지고 있는 목표나 주어진 목표가 구조 구성 행동을 할 것인지, 구조 확립 행동을 할 것인지에 중요한 영향을 미친다. 만약 학생들이 표준 문제를 푼다면 이전에 알고 있는 구조를 인식하거나 구조를 확립하는 행동을 할 가능성이 높다. 하지만 만약 학생들이 비표준 문제를 푼다면, 새로운 구조를 구성해 야할 가능성이 높다. 따라서, 세 가지 인식론적 행동은 서로 분리되어서 선형적으로 일어나는 것이 아니라 서로 포개지고 역동적으로 상호작용하면서 관계망을 형성하는 과정을 통해 발달한다. 구조 구성 행동은 종종 구조 확립과 구조 인식의 행동을 포함한다. 다시 말해 구조 구성은 세 가지 인식론적 행동의 조합인 반면에 구조 인식 행동은 다른 둘에 포함하고, 구조 확립 행동은 구조 구성 행동에 포함한다.

Ⅲ. 연구방법 및 절차

본 연구는 다음과 같은 방법과 절차로 수행되었다.

첫째, 수학과 정보교과가 융합교육을 실시하기 위한 한 방법으로 코딩으로 설정하고, 코딩을 기반으로 수학과 정보교과가 융합하여 특정 개념을 할 수 있는 수업모형을 개발한다. 둘째, 개발한 수업모형 적용 가능한 개념 즉, 특정 주제를 선정한다. 본 연구에서는 블록체인과 비트코인을 이해하기 위해 필요한 수학 중에서 학생에게 어렵지 않게 설명할 수 있는 내용이 무엇이지 추출한다(Shin & Shin, 2017, 2019).

셋째, 추출된 수학을 이해하여 비트코인의 원리를 이해할 수 있도록 학습하기 위한 교수·학습 자료의 개발 방향과 원칙을 설정하고, 이 개발 방향과 원칙에 따라 프로그램 시안을 개발한다. 넷째, 개발된 시안에 대한 전문가의 의견을 듣는다. 전문가는 과학고등학교 수학교사 2인, 수학교육학 교수 1인, 수학 교수 2인으로 구성하였다. 전문가의 구성은 Table 2와 같다. 다섯째, 전문가의 의견을 적극적으로 반영하여 수업모형 및 교수·학습 자료를 완성한다. 여섯째, 본 논문이 제시하는 최종 교수·학습 자료의 구현가능성과 효용성은 동일한 전문가 5인에게 설문함으로 검증한다. 설문 내용은 Table 3과 같다. 필요한 경우 교수·학습 자료를 첨삭하여 최종안을 도출한다. 이 논문에 소개된 교수·학습 자료에 대한 요약본은 최종안만을 제시할 것이다.

Table 2. Composition of Experts

전문가	연번	항목3
수학전문가	전문가1	사범대학에서 수학을 전공한 교수
	전문가2	사범대학에서 수학을 전공한 교수
교과교육전문가	전문가3	사범대학에서 수학교육을 전공한 교수
현장교육전문가	전문가4	박사학위를 소지한 과학고 교사
	전문가5	박사학위를 소지한 과학고 교사

Table 3. Contents of questionnaire

문항	설문 내용
1	수업 주제에 관하여 학생들이 흥미를 가지는가?
2	실제 활용 가능한 교수·학습 자료인가?
3	교수·학습 자료는 수학개념의 획득 및 사고력의 신장에 도움이 되는가?
4	프로그램이 사고력을 향상시킬 수 있는 내용과 문제, 방법으로 구성되어 있는가?
5	교수·학습 자료에 나타난 코딩(프로그래밍)은 개념획득과 사고력 신장에 도움이 되는가?
6	개발된 교수·학습 자료는 수학과 전자서명 융합교육에 효과적이도록 구성되었는가?
7	본 교수·학습 자료로 수업을 할 때, 효과 제고를 위해 무엇을 추가로 고려하여야 하는지 기술하시오.
8	본 교수·학습 자료에 대한 의견을 자유롭게 기술해 주세요.

Ⅳ. 연구결과

1. 수업모형의 개발

본 연구에서는 코딩 기반으로 수학과 정보교과의 융합을 통해 수학적 개념을 획득하기 위한 교수·학습 자료의 개발을 목적으로 한다. 특정 교과에 대한 개념을 학습하면서, 동시에 프로그래밍 언어를 기반으로 하는 코딩을 하는 과정에서 특정 교과의 학습내용을 정확하게 이해할 수 있도록 교수·학습 자료가 구현되어야 한다. 일반적으로 코딩(coding)은 문제해결의 한 유형으로 간주되므로, 학습자가 문제를 이해하고, 해결 방안을 고안하며, 직접 프로그래밍을 실행하여 결과를 검증·반성하는 일련의 과정이다(Lee, 2009).

본 연구에서 수학과 정보의 융합교육을 실현하기 위해서 크게 세 가지가 달성되어야 한다. 첫째, 코딩을 위한 프로그램의 이해, 둘째, 융합교육 주제에 대한 수학적 이해 및 이를 기반으로 한 특정 정보주제에 대한 개념 획득, 셋째, 학습결과에 대한 활용과 실제 상황에서의 적용이다. 이 세 가지 목적을 제한되어진 수업시간에 달성하는 것은 간단한 일이 아니므로, 본 연구에서는 복합 수업모형을 개발하였다. 먼저, 코딩을 위한 프로그램인 SageMath 프로그램에 대한 이해는 본시 수업 이전에 이루어질 필요가 있고, 이를 위해서는 사전 수업활동(Pre-Class)의 제공이 바람직하다(Kim *et al.*, 2017). 둘째, 수학적 이해 및 융합주제에 대한 개념 획득은 본시 수업(In-Class) 활동을 통해 달성되어야하고 이를 위한 특별한 수업절차가 제공되어야 한다. 셋째, 학습결과에 대한 활용과 적용은 소집단에서 과제의 해결이라는 협력적 문제해결이 이루어질 필요가 있고, 이를 위해서는 프로젝트 학습 중심의 사후 수업활동(After-Class)이 필요하다(Shin, 2012;

Choi, 2012). 이에 본 연구에서는 교수·학습 자료 개발을 위해 Kim *et al.* (2017)과 Yoo(2015)의 수업모형을 수정하여 Figure 1과 같은 틀을 개발하였다.

첫째, 사전 수업활동 단계이다. 기초단계로 디딤 영상을 제작하고 코딩을 위한 프로그램 언어를 사전에 학습하는 단계이다. 교사가 직접 학습내용을 선정하고, 제작한 동영상 자료를 활용하여 본시 수업활동에서 필요로 하는 프로그램 언어를 동영상 학습으로 사전에 습득하게 된다. 둘째, 본시 수업활동 단계이다. 활동단계로 관찰 및 탐구, 토의·토론, 모둠 학습, 공학도구 및 매체 활용 등 다양한 활동을 기반으로 수학과 정보의 융합인재교육을 실시하여 개념을 획득하는 단계이다. 이 단계가 학습의 가장 중요한 단계로 Shin(2015)와 Shin, Boo & Suh(2015)가 제시한 이론을 수정하여 다음과 같이 구체적인 절차를 개발하였다. 셋째, 사후 수업활동 단계이다. 적용단계로 사전 수업 및 본시 수업활동에서 습득한 프로그램 언어, 수학 및 정보관련 개념 등을 실제로 활용할 수 있는 수준으로 성장하는 단계이다. 이 단계에서는 소집단이 구성되어 교사가 제시한 과제를 학습자가 주체적으로 수행할 수 있는 프로젝트를 부여한다(Shin, 2012; Choi, 2012).

Figure 1. Process of Teaching & Learning

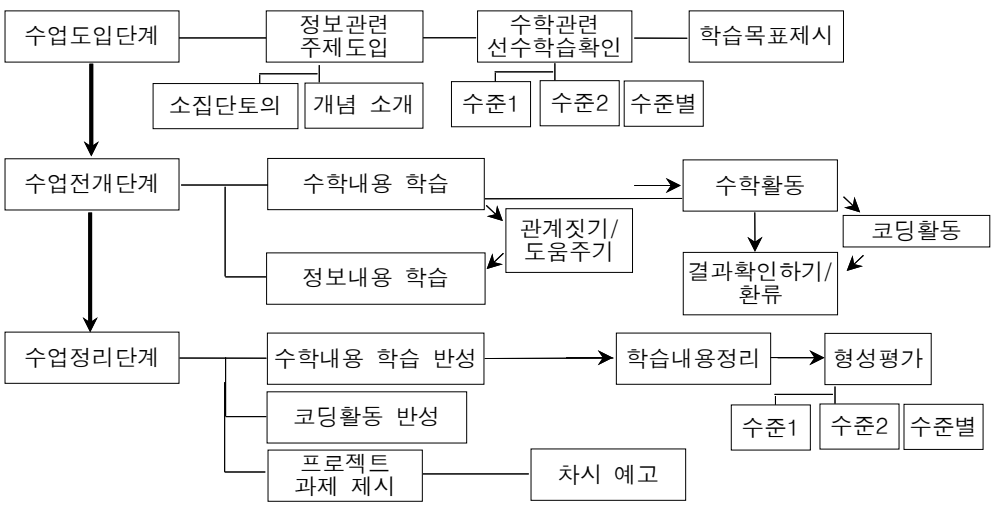


Figure 2. In-Class Teaching & Learning Model

2. 교수학습 자료의 개발

수학·정보 융합교육을 위한 코딩과 연계한 교수·학습 자료의 개발 목표는 크게 세 가지로 구분할 수 있다. 첫째, 수학이 정보교과와 융합할 수 있고, 수학이 전자서명에 결정적인 역할을 하고 있다는 수학적 안목을 가지도록 한다. 둘째, 전자서명에 담긴 수학적 원리와 개념을 이해하는 과정에서 추상화 및 논리적 사고력을 신장시킨다. 셋째, 전자서명과 관련된 수학적 개념을 SageMath 프로그램에서 코딩하는 경험을 통해 추상화 및 논리적 사고력을 신장시킨다.

이를 효과적으로 달성하기 위해 교수·학습 자료의 개발 방향은 다음과 같다. 이러한 개발 방향은 Koo(2004)등의 연구를 고려하였다. 첫째, 전자서명에 관계된 수학 중, 논리적 사고력 및 개념에 대한 추상화에 최적의 내용을 선정한다. 둘째, 코딩을 위해 사용하는 프로그래밍 명령어 및 문법의 난이도를 분석하여 사범대학 수학교육과 예비교사 및 과학고등학교 학생이 학습하기에 적절한 명령어 및 문법 내용을 선정하여 프로그램을 개발한다. 더불어 문제해결에 필수적인 명령어 및 문법만을 사용할 수 있도록 한다. 셋째, 교수·학습의 순서는 수학적 문제 상황의 난이도와 프로그래밍 내용의 난이도를 모두 고려하여 점진적으로 심화하는 것으로 구성한다. 넷째, 학습자가 전자서명에 담긴 수학뿐만 아니라, SageMath 프로그램 코딩에도 흥미를 가질 수 있도록 난이도를 구성하고, 다양한 맥락과 연계한 문제 상황을 제시한다. 다섯째, 사범대학 기초정수론 강의, 과학고등학교 및 일반고등학교 수학동아리에서 실제로 활용할 수 있도록 구성한다.

또한, 본 교수·학습 자료의 개발을 위한 세 가지 원칙을 설정하였다. 이러한 원칙은 Han(2006)이 제시한 수학교수학적 원리를 기초로 본 연구의 수행에 부합되도록 선정하였다. 첫째, 개념학습을 위한 교수·학습의 원칙이다. 수학과 정보교과에 필요한 개념학습이 이루어지도록 자료를 개발한다. 개념의 정의와 설명이 이루어지고, 이러한 설명이 학생들이 이해할 수 있는 수준에서 이루어지며, 다루는 개념과 관계되어진 다양한 속성들을 설명함으로써 학생들의 개념학습에 도움을 주도록 한다. 또한 학생들이 개념을 학습했다고 해도, 그 개념을 실제로 적절히 사용하는 일은 쉽지 않으므로, 학생들이 자신이 획득한 개념을 다양한 상황에서 적절하게 사용할 수 있도록 개발한다. 둘째, PERC에 부합된 교수·학습의 원칙이다. 수학교육에 컴퓨터 프로그래밍 교육을 접목시키는 방법으로 Camp & Marichionini(1984)는 학습내용에 초점을 맞춘 PERC(Programming Exercise Related to Contents)를 제안하였다. Lee(1992)는 이에 대해 프로그래밍을 통한 교육으로 학생들의 올바른 수학적 개념의 이해와 수학적 인 사고의 발달, 문제해결력의 신장 등과 같은 효과를 얻기 위해서는 프로그래밍이 수학적 학습 내용과 부합되어야 한다고 언급하였다. 즉, 수학이 정보와 융합하여 코딩교육과 연계하기 위해서는 학생들은 프로그래밍 언어를 알고 있어야 하고 이를 활용할 수 있어야 하는데, 이 프로그램이 수학공부를 방해하는 주객전도가 일어나서는 안 된다는 입장이다. PERC 구성 형태를 어떻게 할 것인가의 문제는 수업의 중심이 수학이냐 프로그래밍이냐의 문제로, 너무 어렵고 복잡한 프로그램 요소가 필요한 수학 문제라면 다루지 않도록 하였다. 셋째, 활동학습에 기반한 교수·학습의 원칙이다. 활동학습은 특정한 공동체 내에서 인간 행위와 관련된 다양한 요소들의 상호 관계를 파악하여 학습하는 이론이다. 사전 수업활동에서는 플립러닝을 통해 교사와 학생사이의 상호관계를 바탕으로 학습자가 자기 주도적으로 학습활동이 이루어지도록 하고, 본시 수업활동에서는 소집단내에서의 토의활동, 수학적 문제를 해결하는 활동, 컴퓨터프로그램을 통한 협력하는 활동을 통한 학습활동이 이루어지도록 하면, 사후 수업활동에서는 특정과제와 관계한 프로젝트 수행을 통한 학습활동이 이루어지도록 하였다.

3. 교수·학습 자료의 검증 및 수업지도안에 관한 전문가 의견 반영

본 연구에서 개발한 교수·학습 자료에 대해 그 타당성과 현실 적용가능성, 내용의 적절성 등을 파악하기 위해 전문가 설문을 실시하였다. Lee(2012)의 설문 문항을 참고로 하였으며, 설문 문항은 총 8문항으로, 2문항은 자유서술형이고 6문항은 5점 척도 문항으로 Table 3과 같다.

각 설문지에 대한 반응의 결과를 5점 척도로 하여 평균을 구하면 Table 4와 같다. 1번과 6번 문항에 대해 높은 반응 점수를 보였는데, 이는 전문가들이 본 연구에서 개발한 주제를 학생들이 매우 흥미롭게 인식할 것으로 판단함을 의미한다. 반면, 3번 문항과 4번 문항에 대해서는 낮은 인식을 보였다.

구체적으로 각 문항별 전문가들의 반응과 이에 따른 교수·학습 자료의 개선 및 반영한 부분을 제시하면 다음과 같다.

첫째, 문항2의 경우, 수학전문가와 현장교육전문가는 서로 다른 상반된 반응을 보였다. 전문가 1, 2, 3은 실제 활용가능성에 매우 긍정적이었지만, 전문가 4, 5는 그렇지 않았다. 실제로 전문가 4는 '일반고에서는 활용하기가 곤란하다.'라고 하였고, 전문가 3도 '좀 더 정교화, 세분화되어야'라는 의견을 제시하였다. 이러한 의견에 따라 교수·학습 자료를 일반계 고등학생들이 친숙하게 느끼도록 수학내용 및 정보관련 내용을 수정하였다. 또한 전문가 4와 5의 의견을 고려하여 고등학교 R & E에서 활용하도록 수정하였다.

둘째, 문항3에 대한 전문가 1, 2, 3의 의견을 요약하면 다음과 같다.

> 수학적 개념의 획득보다는 미적분, 정수론, 대수학에 관한 수학 지식을 이미 학습한 후에 이런 수학적 지식들이 실생활에서 활용되는 사례로서 '수학의 힘'을 느끼고 사고력을 좀 더 심화시키는 주제에 더 가깝습니다. 또한, 타원곡선에서의 연산의 개념과 전자서명 활용의 이해라는 수학적 개념의 획득은 가능하나 획득된 개념을 다른 부분에 활용하는 것은 쉬운 일이 아니므로 사고력 신장에 어려운 측면이 있습니다.

위 의견을 수용하여, 수학적으로는 타원방정식에 대한 개념, 무한원점의 개념, 집합 $\mathbb{Z}_{17}$에서 덧셈 및 곱셈 연산의 개념 획득에 초점을 맞추어 자료를 수정하였다. 또한 정보와 관련해서는 비트코인의 전자서명의 개념, 인증의 개념에 초점을 맞추어 자료를 완성하였다.

셋째, 문항4에 대한 전문가의 반응은 전체 문항 중에서 가장 낮게 나타났다. 전문가 3은 '코딩을 위한 기본적인 언어와 문법을 숙지한다면 코딩의 속성상 사고력 향상에 크게 도움이 될 것으로 사료됨'이라고 한 것을 제외하면, 전문가 4는 '투입하는 문제와 방법 하나하나를 재음미해

Table 4. Result of questionnaire

문항	설문 내용	평균 (5점 만점)
1	수업 주제에 관하여 학생들이 흥미를 가지는가?	4.8
2	실제 활용 가능한 교수·학습 자료인가?	4.4
3	교수·학습 자료는 수학개념의 획득 및 사고력의 신장에 도움이 되는가?	4.2
4	프로그램이 사고력을 향상시킬 수 있는 내용과 문제, 방법으로 구성되어 있는가?	4.2
5	교수·학습 자료에 나타난 코딩(프로그래밍)은 개념획득과 사고력 신장에 도움이 되는가?	4.4
6	개발된 교수·학습 자료는 수학과 전자서명 융합교육에 효과적이도록 구성되었는가?	4.6

보고 잘 구현되는지를 지속적으로 수정/보완'을 요구하였고, '정보 코딩에 싫증을 느낄지 않을까' 라고 하여 사고력에 대해 회의적 시각을 보였다. 이에 본 연구에서는 플립러닝(flipped learning)을 통해 프로그램을 사전에 학습하고, 본시 수업활동에서는 사고력 향상을 위한 코딩 및 수학개념 학습에 전념할 수 있도록 개선하였고, 최종적으로 사후 수업활동의 프로젝트 수행을 통해 사고력 향상이 극대화될 수 있도록 개선하였다.

넷째, 문항7에 대해서 각 전문가들이 다양한 반응을 보였다. '화폐의 역할을 하는 것이 무엇인지 분명하게 할 필요가 있다.', '보다 많은 수업시간을 확보하고, 더 친절한 설명과 간단한 여러 예를 들어 많은 계산을 할 수 있다면 보다 효과적인 수업이 될 것 같다.', '타원곡선 $y^2=x^3+7$에서 덧셈연산의 정의 과정과 전자서명의 서명절차와 인증절차 사이의 난이도 차이가 있으며, 둘 사이의 관계가 좀 더 구체적으로 연결되고 학습자 입장에서 좀 더 쉽게 설명되었으면 좋겠다.', '코딩을 Sagemath에 국한하지 않고 다른 언어로 코딩하도록 하는 것도 필요하다.' 등의 의견이었다. 이에 본 연구에서는 화폐의 역할은 사전 수업활동을 통해 보완하고, 난이도의 차이가 있는 것은 서로 유사한 난이도가 유지될 수 있도록 보완하였다. SageMath 이외의 부분은 수학 이외의 비본질적 부담감의 증가로 고려하지 않았다.

4. 교수학습 자료의 확정

본 연구에서 설정한 방향과 원칙에 따라 개발한 자료는 전문가 의견을 거쳐 수정되었다. 최종 교수·학습 자료를 요약하여 제시하면 다음과 같다.

1) 수학·정보 융합교육을 위한 학습주제: Bitcoin 거래를 위한 전자서명(digital signature)속 수학

2) 학습목표

○ '전자서명'의 개념을 이해하고, 타원곡선 $y^2=x^3+7$위에서 정의된 (덧셈)연산을 이해하고, 삼차방정식의 근과 계수와의 관계, 접선의 방정식, 음함수 미분법 등을 이용하여 구체적인 연산을 직접 계산할 수 있다.

○ 특정 집합 위에서의 타원곡선을 이해하고 구체적인 연산을 계산할 수 있다.

○ '기저점'의 개념을 이해하고 이와 관련된 계산을 할 수 있다.

○ 비트코인 거래에서 이용되는 ECDSA의 서명절차와 인증절차를 이해하고 인증절차의 정당성에 대하여 설명할 수 있다.

○ 수학에서 활용가능한 SageMath 프로그램을 이용하여 전자서명과 관련된 다양한 코딩을 할 수 있고, 그 과정을 논리적으로 설명할 수 있다.

3) 'Bitcoin 거래를 위한 전자서명 속 수학'에 대한 교수·학습 자료

○ **준비 단계: 코딩 융합학습을 위한 준비사항이 무엇인지 소개한다.**

- 교사는 학생들의 학습수준, 수업환경 등을 고려하여 아래 내용을 재구성하여 학습 자료를 준비한다. 또한 수업 시간 전에 '화폐의 기능' 수업을 수강하는 학생들에게 학습 자료를 제시하여 학생들이 미리 학습 자료를 읽을 수 있도록 한다.
- SageMath 프로그램을 활용한 코딩 활동을 수행하기 위해 인터넷에 접속이 가능한 컴퓨터 혹은 SageMath가 설치된 컴퓨터를 준비한다.
  · 인터넷에 접속이 가능한 경우 학생들이 웹사이트(http://cocalc.com)에 접속하여 계정을 미리 만들도록 한다. 이 웹사이트에서 Sage Worksheet를 생성하고 SageMath 프로그램을 이용할 수 있다.
  · SageMath는 웹사이트(http://www.sagemath.org/download-windows.html)에서 다운로드하여 설치할 수 있으며 https://wiki.sagemath.org/SageWindows에서 설치 방법을 확인할 수 있다.

- SageMath 프로그램을 위한 명령어 사전학습 내용을 동영상으로 사전에 학생에게 제공한다.

○ **도입 단계: 융합학습에 대한 동기부여와 기초적인 개념을 소개한다.**

- 수학과 정보의 융합인 전자서명에 대한 학습 동기를 부여하기 위한 활동을 전개한다.

· 정보와 수학 융합수업의 첫 번째 단계로서 '비트코인', '블록체인'의 개념과 역할에 대해 소개한다.

· 소집단별로 자유롭게 토의를 진행한다.

(토의주제) 실생활에서의 접하는 '비트코인'이 무엇인지 간단히 토의하고, 이를 토대로 '블록체인'이 수행해야 할 기능으로 기대할 수 있는 것이 무엇인지 조사하고, 조별로

Table 5. Basic syntax of SageMath

구 분	명령어	설 명
파이썬 (Python)	def/return	def는 사용자가 함수를 만들 때 이용되는 명령어이다. `def 함수명(매개변수): <수행할 내용>`
	if/else	if와 else는 주어진 조건문의 참, 거짓에 따라 수행할 일을 명령할 때 사용한다. `if <조건문>: <수행할 내용1> else: <수행할 내용2>`
	for	for는 리스트의 모든 요소를 차례로 대입하여 수행할 일을 명령할 때 사용한다. for문의 기본 구조는 다음과 같다. `for (변수) in (리스트): <수행할 내용>`
	while	while은 주어진 조건문이 참인 동안 계속 반복하여 수행할 일을 명령할 때 사용한다. `while <조건문>: <수행할 내용>`
	append	'[1, 2, 3]'과 같은 자료의 형태를 리스트(list)라고 한다. 리스트명 뒤에 '.append((추가할 요소))'를 입력하면 새로운 리스트를 결과 값으로 돌려받는다. 예를 들어, `L = [1, 2, 3]` `L.append(4)` 를 입력하면 변수 L에는 리스트 [1, 2, 3, 4]가 대입된다.
	(리스트명)[(숫자)]	인덱싱(indexing)은 리스트가 주어졌을 때 특정 위치의 요소를 가리키는 것을 말한다. 예를 들어 명령 `L = [10, 25, 33]`을 통해 변수 L에 리스트 [10, 25, 33]을 대입한 뒤 `L[0]`, `L[1]`, 또는 `L[2]`를 입력하면 각각 0번, 1번, 2번 위치에 있는 10, 25, 33이 출력된다.
	import/from	이미 저장된 모듈(module)이 있다면 `import (모듈명)`으로 모듈을 불러올 수 있다. 모듈의 특정 함수만 불러올 경우에는 다음과 같다. `from (모듈명) import (함수명)`
SageMath I: 그래프	point	point는 순서쌍 (a, b)를 입력 받아 좌표평면 위의 있는 점 (a, b)을 결과 값으로 준다. 예를 들어 `point((2, 3))`과 같이 입력할 수 있다.
	plot	plot은 함수 $f(x)$와 정의역을 입력 받아 좌표평면 위의 $y=f(x)$의 그래프를 결과 값으로 준다. 예를 들어, 닫힌구간 $[-1, 1]$에서 함수 $y=x^2$의 그래프를 얻기 위해서 `plot(x^2, (x, -1, 1))`를 입력하면 된다.
	list_plot	list_plot은 순서쌍들을 요소로 가지는 리스트를 입력 받아 각각의 요소에 대응되는 좌표평면 위의 점들을 결과 값으로 준다. 예를 들어 두 점 $(2, 3)$, $(1, 5)$는 `list_plot([(2, 3), (1, 5)])`를 입력하면 된다.
	implicit_plot	implicit_plot은 두 변수 x, y의 관계식 $F(x, y)=0$과 x의 범위, y의 범위를 입력 받아 좌표평면 위의 $F(x, y)=0$의 그래프를 결과 값으로 준다. 예를 들어, 방정식 $y^2=x^3+7$의 그래프를 얻기 위해서 `implicit_plot(y^2 == x^3 + 7, (x, -3, 3), (y, -6, 6))`를 입력한다.
	parametric_plot	parametric_plot은 매개화된 도형을 얻을 때 이용한다. 예를 들어, 단위원을 얻기 위해서는 다음과 같다. `t = var('t')` `parametric_plot((cos(t), sin(t)), (t, 0, 2*pi))`
SageMathII: 대수적구조	Integers	`Integers(5)`를 입력하면 결과 값으로 체(field)의 구조를 가지는 $\mathbb{Z}_5$를 얻는다. 한편, `Integers()`를 입력하면 정수 전체를 얻는다.
	inverse_of_unit	`R = Integers(5)`를 입력하여 변수 R에 $\mathbb{Z}_5$를 대입한 뒤 `R(3)`을 입력하면 결과 값으로 $\mathbb{Z}_5$의 원소 3을 돌려받는다. 이때 `R(3).inverse_of_unit()` 과 같이 $\mathbb{Z}_5$의 원소 뒤에 '.inverse_of_unit()'을 입력하면 그 원소의 곱셈의 대한 역원을 결과 값으로 돌려준다.
	EllipticCurve	EllipticCurve는 체와 두 개의 수로 구성된 리스트를 입력 받아 타원곡선을 결과 값으로 돌려준다. 예를 들어 `EllipticCurve(Integers(5), [1, 3])`을 입력하면 $\mathbb{Z}_5$위에서 정의된 $y^2=x^3+x+3$을 결과 값으로 준다.

그 결과를 정리하시오.

· 왜, 비트코인이 현재 사회에게 중요한 이슈가 되고 있고, 이 비트코인의 핵심기술이 미래사회에서 왜 중요한지 다양한 자료를 검색하여 학생들의 실생활에서 수학이 얼마나 다양하게 사용되고 있는지 교사가 학생에게 설명을 한다.

- 전자서명에 대한 정의를 제시한다.

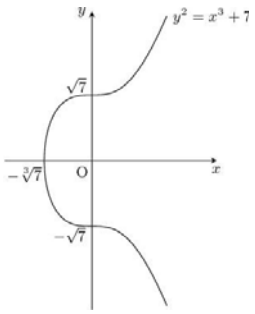
실생활에서 '서명(signature)'은 문서의 진실성(integrity)을 그 문서에 서명한 사람의 신분에 의하여 증명하는 방법으로 이용한다. 따라서 전자문서에서 '전자서명(digital signature)'은 일반 문서에서 서명이 담당한 기능을 수행해야 한다. 우리나라 전자서명법(법률 제14839호)에는 전자서명을 '서명자를 확인하고 서명자가 당해 전자문서에 서명을 하였음을 나타내는데 이용하기 위하여 당해 전자문서에 첨부되거나 논리적으로 결합된 전자적 형태의 정보'로 정의한다.

- 전자서명 알고리즘의 종류에 대해 설명을 한다.

전자서명(digital signature)의 개념은 1976년 디피(W. Diffie)와 헬만(M. Hellman)에 의해 정립되었다. 그들은 전자서명 알고리즘을 실제로 제안하지는 않았지만 기본적인 구상을 제시하였다. 전자서명의 실제 구현은 1977년 라이베스트(R. Rivest), 샤미르(A. Shamir), 애들만(L. Adleman) 세 명에 의해 이루어졌다. 'RSA 알고리즘'이다. 이후 다양한 전자서명 기법이 소개되었다. 1984년 엘가말(T. ElGamal)에 의한 서명기법(El-Gamal signature scheme)과 골드바서(S. Goldwasser), 미칼리(S. Micali), 라이베스트

(활동1) $y^2=x^3+7$의 모든 해 (x,y)를 좌표평면에 나타내시오.

(설명) $y^2=x^3+7$의 모든 해 (x,y)를 좌표평면에 나타내면 아래 그림과 같은 곡선을 얻을 수 있다. 이 곡선은 '타원곡선(elliptic curve)'의 특별한 예이다.



(코딩1) SageMath 프로그램을 이용하여 다음 방정식의 모든 해 (x,y)를 그래프로 그리시오.

① $2x-y=0$ ② $2x^2-3x-3-y=0$ ③ $y^2=x^3+7$

(코딩과정) 두 변수 x와 y를 선언한 뒤, 'implicit_plot' 함수를 이용하여 주어진 방정식의 그래프를 얻는다. 이 함수는 'implicit_plot(INPUT1, INPUT2, INPUT3)'와 같이 세 개의 입력 값을 가지는 함수로 첫 번째 입력 값으로는 방정식, 두 번째와 세 번째 입력 값으로는 각각 변수 x와 y의 값의 범위를 가진다.

```
sage: x = var('x'); y = var('y')
sage: implicit_plot(2*x - y == 0, (x, -1.5, 1.5), (y, -3, 3))
```

위와 같이 입력하면 첫 번째 방정식의 그래프를 얻는다.

마찬가지로 다음을 각각 입력하여 두 번째와 세 번째의 방정식의 그래프를 얻을 수 있다.

```
sage: implicit_plot(2*x^2 - 3*x - 3 - y == 0, (x, -2, 3.5), (y, -5, 7))
sage: implicit_plot(y^2 == x^3 + 7, (x, -3, 3), (y, -6, 6))
```

Figure 3. An example of Activity 1 and Coding 1

(R. Rivest)에 의해 제안된 GMR 서명 기법은 대표적인 예이다.

- 비트코인 전자서명 알고리즘에 대한 학습을 한다.

· 타원곡선 전자서명 알고리즘이 무엇인지 설명하고, 활동으로 타원곡선의 그래프를 그린다.

타원곡선 $y^2=x^3+7$은 비트코인의 거래에서 중요한 방정식이다. 전자서명의 방법 중 하나로 타원곡선 전자서명 알고리즘(elliptic curve digital signature algorithm, ECDSA)이 있다. 실제로 ECDSA는 비트코인(bitcoin)의 거래(transaction) 문서에 전자서명을 하는 방식으로 채택되었다.

○ **전개 단계의 학습: 수학적인 기초내용 및 이를 기초로 한 수학적인 원리를 소개한다.**

- 타원곡선 알고리즘을 이해하기 위한 수학적인 기초내용을 학습한다.

· 방정식 $y^2=x^3+7$을 이용한 집합 C의 정의와 집합 C에서 덧셈연산을 정의한다.

집합 C는 방정식 $y^2=x^3+7$의 해 전체의 집합으로 정의한다. 즉, $C=\{(x,y)|y^2=x^3+7\}$이다. 집합 C는 방정식 $y^2=x^3+7$위의 점을 원소로 하고, 덧셈연산을 다음 두 가지의 경우로 나누어 정의한다.

첫째, x좌표가 같지 않은 두 점 $\mathrm{P}(x_\mathrm{P},y_\mathrm{P}),\mathrm{Q}(x_\mathrm{Q},y_\mathrm{Q})\in C$에 대하여 두 점의 합 $\mathrm{P}+\mathrm{Q}$는 다음과 같다.

(ⅰ) 두 점 P, Q를 지나는 직선과 곡선 $y^2=x^3+7$은 세 점 P, Q, R에서 만난다.

(ⅱ) 점 R을 x축 대칭시켜 얻은 점 R′은 다시 곡선 $y^2=x^3+7$ 위의 점이다.

이때, $\mathrm{P}+\mathrm{Q}$를 R′으로 정의한다.

둘째, $y_\mathrm{P}\neq 0$인 점 $\mathrm{P}(x_\mathrm{P},y_\mathrm{P})\in C$에 대하여 $\mathrm{P}+\mathrm{P}$는 다음과 같다.

(ⅰ) 점 P에서 곡선 $y^2=x^3+7$의 접선과 곡선 $y^2=x^3+7$은 두 점 P, R에서 만난다.

(ⅱ) 점 R을 x축 대칭시켜 얻은 점 R′은 다시 곡선 $y^2=x^3+7$ 위의 점이다.

이때, $\mathrm{P}+\mathrm{P}$를 R′으로 정의한다.

· 무한원점의 정의, 새로운 집합 E와 집합 E에서 덧셈연산을 정의한다.

집합 C위에서 덧셈연산이 정의되지 않는 것은 심각한 문제이다. 이 문제를 해결하기 위하여 '무한원점(無限遠點, point at infinity)'의 개념을 도입한다. 여기서는 무한원점을 I로 나타내기로 한다. 무한원점은 좌표평면 위의 점이 아님을 주의하자. 즉, I는 (x,y) (x,y는 실수) 꼴로 나타낼 수 없다. 무한원점은 무한히 먼 곳에 존재하는 한 점이라고 판단할 수 있다. 대수적(algebraic) 관점에서 무한원점이 필요한 이유는 덧셈연산이 잘 정의되도록 하는 특별한 점의 필요에 의해서 도입된 점이다.

무한원점 I를 이용하여 앞에서 정의가 되지 않았던 $\mathrm{P}+\mathrm{Q}$와 $\mathrm{P}+\mathrm{P}$를 정의할 수 있다.

(ⅰ) x좌표가 같은 서로 다른 두 점 $\mathrm{P}(x,y),\mathrm{Q}(x,-y)\in C$에 대하여 $\mathrm{P}+\mathrm{Q}=\mathrm{I}$로 정의한다.

(ⅱ) $\mathrm{P}(-\sqrt[3]{7},0)$에 대하여 $\mathrm{P}+\mathrm{P}=\mathrm{I}$로 정의한다.

(ⅲ) 임의의 점 $\mathrm{P}\in C$에 대하여 $\mathrm{P}+\mathrm{I}=\mathrm{P}$, $\mathrm{I}+\mathrm{P}=\mathrm{P}$로 정의한다.

(ⅳ) $\mathrm{I}+\mathrm{I}=\mathrm{I}$로 정의한다.

위와 같이 정의한 덧셈에 대하여 일반적으로 $(x_\mathrm{P},y_\mathrm{P})+(x_\mathrm{Q},y_\mathrm{Q})=(x_\mathrm{P}+x_\mathrm{Q},y_\mathrm{P}+y_\mathrm{Q})$이 성립하지 않는다.

이제 무한원점 I와 방정식 $y^2=x^3+7$의 모든 해 (x,y)로 구성된 집합을 E로 나타내자. 즉, $E=C\cup\{\mathrm{I}\}$라 한다. 정의로부터 다음과 같은 대수적인 성질을 확인할 수 있다.

(ⅰ) 임의의 $\mathrm{P},\mathrm{Q}\in E$에 대하여 $\mathrm{P}+\mathrm{Q}\in E$이다.

(ⅱ) 임의의 $\mathrm{P},\mathrm{Q}\in E$에 대하여 $\mathrm{P}+\mathrm{Q}=\mathrm{Q}+\mathrm{P}$이다.

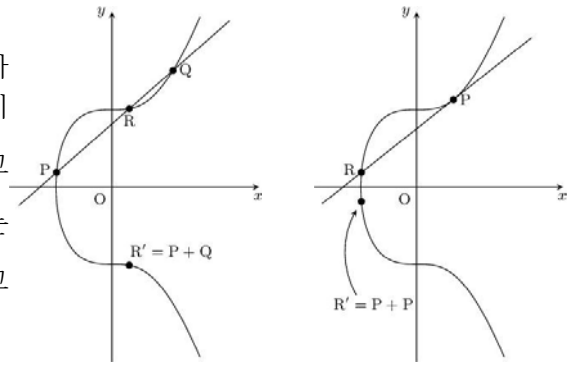
(ⅲ) 임의의 $\mathrm{P}\in E$에 대하여 $\mathrm{P}+\mathrm{I}=\mathrm{I}+\mathrm{P}=\mathrm{P}$이다.

(ⅳ) 임의의 $\mathrm{P}\in E$에 대하여 $\mathrm{P}+\mathrm{Q}=\mathrm{Q}+\mathrm{P}=\mathrm{I}$인 $\mathrm{Q}\in E$가 유일하다. 이때 $\mathrm{P}+\mathrm{Q}=\mathrm{Q}+\mathrm{P}=\mathrm{I}$인 Q를 $-\mathrm{P}$로 나타낸다. 더욱이 $(x,y)\in C$에 대하여 $(x,-y)\in C$이며 $(x,y)+(x,-y)=(x,-y)+(x,y)=\mathrm{I}$이다. 즉,

(활동2) 집합 C위의 두 점 $\mathrm{P}(x_\mathrm{P}, y_\mathrm{P}), \mathrm{Q}(x_\mathrm{Q}, y_\mathrm{Q})$에 대하여 P+Q연산과 P+P연산 결과를 그래프로 표현하시오. 또한, $x_\mathrm{P} \neq x_\mathrm{Q}$인 두 점 $\mathrm{P}(x_\mathrm{P}, y_\mathrm{P}), \mathrm{Q}(x_\mathrm{Q}, y_\mathrm{Q})$에 대하여 P+Q와 P+P을 구하시오.

(설명) 집합 C에서의 덧셈연산을 그래프로 표현하면 아래와 같다.

또한, $x_\mathrm{P} \neq x_\mathrm{Q}$인 두 점 $\mathrm{P}(x_\mathrm{P}, y_\mathrm{P}), \mathrm{Q}(x_\mathrm{Q}, y_\mathrm{Q}) \in C$에 대하여 P+Q는 $(m^2 - x_\mathrm{P} - x_\mathrm{Q}, m(x_\mathrm{P} - x_{\mathrm{P+Q}}) - y_\mathrm{P})$이다. 여기서 $m = \dfrac{y_\mathrm{Q} - y_\mathrm{P}}{x_\mathrm{Q} - x_\mathrm{P}}$이고 $x_{\mathrm{P+Q}} = m^2 - x_\mathrm{P} - x_\mathrm{Q}$이다. 그리고 $y_\mathrm{P} \neq 0$인 점 $\mathrm{P}(x_\mathrm{P}, y_\mathrm{P}) \in C$에 대하여 P+P는 $(m^2 - 2x_\mathrm{P}, m(x_\mathrm{P} - x_{\mathrm{P+P}}) - y_\mathrm{P})$이다. 여기서 $m = \dfrac{3x_\mathrm{P}^2}{2y_\mathrm{P}}$이고 $x_{\mathrm{P+P}} = m^2 - 2x_\mathrm{P}$이다.



(코딩2) SageMath 프로그램을 이용하여 집합 C위의 두 점 $\mathrm{P}(x_\mathrm{P}, y_\mathrm{P}), \mathrm{Q}(x_\mathrm{Q}, y_\mathrm{Q})$에 대하여 다음 물음에 답하시오.

① P+Q를 출력하는 함수와 이를 그래프로 나타내는 함수를 구현하시오.

② $\mathrm{P}(\sqrt[3]{2}, 3)$와 $\mathrm{Q}(-\sqrt[3]{2}, \sqrt{5})$에 대하여 P+Q의 값을 구하고, 그래프로 구현하시오.

③ $\mathrm{P}(\sqrt[3]{2}, 3)$에 대하여 P+P(=2P)의 값을 구하고, 그래프로 구현하시오.

(코딩과정) 두 점 P와 Q를 입력하였을 때 P+Q를 출력하는 함수 'sum'과 이를 그래프로 나타내는 함수 'graph'를 정의한다. 다음과 같이 if문을 이용하여 두 점이 같은 경우와 다른 경우로 나누어 직선의 기울기인 m을 정의할 수 있다.

```
sage: x = var('x'); y = var('y')
sage: def sum(P, Q):
....:     if P != Q:
....:         m = (Q[1] - P[1])/(Q[0] - P[0])
....:     else:
....:         m = (3 * P[0]^2) / (2 * P[1])
....:     x = m^2 - P[0] - Q[0]
....:     y = m * (P[0] - x) - P[1]
....:     return (x, y)
....:
sage: def graph(P, Q):
....:     C = implicit_plot(y^2 == x^3 + 7, (x, -3, 3), (y, -6, 6))
....:     A = point(P, color='red', pointsize=30)
....:     B = point(Q, color='red', pointsize=30)
....:     if P != Q:
....:         m = (Q[1] - P[1]) / (Q[0] - P[0])
....:     else:
....:         m = (3 * P[0]^2) / (2 * P[1])
....:     L = plot(m*(x - P[0]) + P[1], (x, -3, 3), color='gray')
....:     S = point(sum(P, Q), color='green', pointsize=30)
....:     return C + A + B + L + S
```

이제, 위와 같이 정의한 두 함수 'sum'과 'graph'를 이용하여 다음을 계산할 수 있다.

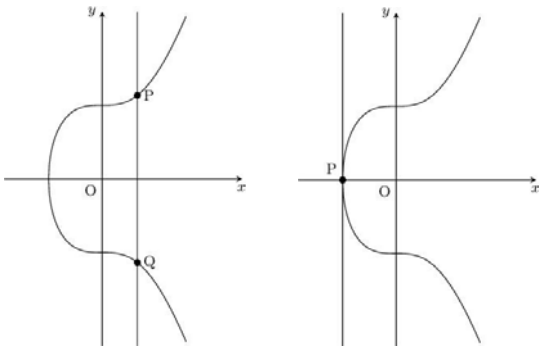
```
sage: sum(P, Q)
(1/8*2^(1/3)*(sqrt(5) - 3)^2,
 1/32*2^(2/3)*(2^(1/3)*(sqrt(5) - 3)^2 - 8*2^(1/3))*(sqrt(5) - 3) - 3)
sage: graph(P, Q)

sage: sum(P, P)
(-3/2*2^(1/3), -1/2)
sage: graph(P, P)
```

'sage:'로 시작하는 줄은 'sage:' 뒤에 내용을 사용자가 입력한 것을 의미하고, 마지막 줄 '(1/8*2^(1/3)……'과 같이 'sage:'로 시작하지 않는 줄은 결과 값을 의미한다.

Figure 4. An example of Activity 2 and Coding 2

(활동3) 집합 C위의 두 점 $\mathrm{P}(x_\mathrm{P}, y_\mathrm{P}), \mathrm{Q}(x_\mathrm{Q}, y_\mathrm{Q})$에 대하여 $\mathrm{P}+\mathrm{Q}$는 x좌표가 같지 않다는 조건, $\mathrm{P}+\mathrm{P}$에서는 $y_\mathrm{P} \neq 0$이라는 조건이 필요한 이유를 설명하시오.

(설명) x좌표가 같은 서로 다른 두 점 $\mathrm{P}(x, y), \mathrm{Q}(x, -y) \in C$의 경우 위와 같은 방법으로 $\mathrm{P}+\mathrm{Q}$를 정의할 수 없다. 두 점 P, Q를 지나는 직선과 곡선 $y^2 = x^3 + 7$은 오로지 두 점 P, Q에서 만나기 때문이다. 비슷한 이유로 점 $\mathrm{P}(-\sqrt[3]{7}, 0)$의 경우 $\mathrm{P}+\mathrm{P}$를 위와 같은 방법으로 정의할 수 없다. 접선이 x축과 수직이기 때문이다.



(코딩3) $\mathrm{P}(\sqrt[3]{2}, 3)$와 $\mathrm{Q}(\sqrt[3]{2}, -3)$에 대하여 (코딩2)에서 구현한 함수를 이용하여 $\mathrm{P}+\mathrm{Q}$를 구하면 어떠한 결과 값이 출력되는지 확인하고, 이 상황을 그래프로 표현하시오.

(코딩과정) $\mathrm{P}(\sqrt[3]{2}, 3)$와 $\mathrm{Q}(\sqrt[3]{2}, -3)$에 대하여 'sum(P, Q)'를 입력하면 다음과 같은 에러 메시지가 출력된다.

```
sage: P = (2^(1/3), 3)
sage: Q = (2^(1/3), -3)
sage: sum(P, Q)
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
<ipython-input-8-cf47e37cae30> in <module>()
----> 1 sum(P, Q)

<ipython-input-1-b2becc43e952> in sum(P, Q)
      1 def sum(P, Q):
      2     if P != Q:
----> 3         m = (Q[Integer(1)] - P[Integer(1)])/(Q[Integer(0)] - P[Integer(0)])
      4     else:
      5         m = (Integer(3) * P[Integer(0)]**Integer(2)) / (Integer(2) * P[Integer(1)])

... (생략) ...

ZeroDivisionError: Symbolic division by zero
```

두 점 P와 Q의 x좌표가 같기 때문에 m을 구하는 과정에 0으로 나누는 계산을 포함하기 때문이다. 위 에러 메시지의 마지막 줄에 'ZeroDivisionError'는 이를 나타낸다.

한편, 다음과 같이 입력하면 이 상황을 그래프로 표현할 수 있다.

```
sage: P = (2^(1/3), 3)
sage: Q = (2^(1/3), -3)
sage: y = var('y')
sage: C = implicit_plot(y^2 == x^3 + 7, (x, -3, 3), (y, -6, 6))
sage: A = point(P, color='red', pointsize=30)
sage: B = point(Q, color='red', pointsize=30)
sage: L = parametric_plot((2^(1/3), y), (y, -6, 6), color='gray')
sage: C + A + B + L
```

Figure 5. An example of Activity 3 and Coding 3

$-(x, y) = (x, -y)$이다. 또한 $\mathrm{I} \in E$에 대하여 $\mathrm{I}+\mathrm{I}=\mathrm{I}$이므로 $-\mathrm{I}=\mathrm{I}$이다.

또한, 다음이 성립함이 알려져 있다.

(v) 임의의 $\mathrm{P}, \mathrm{Q}, \mathrm{R} \in E$에 대하여 $(\mathrm{P}+\mathrm{Q})+\mathrm{R}=\mathrm{P}+(\mathrm{Q}+\mathrm{R})$이다.

타원곡선 전자서명 알고리즘에서는 점 $\mathrm{P} \in E$에 대하여 P, P+P, P+P+P, P+P+P+P, … 등의 계산이 등장한다. 편의상 양의 정수 n에 대하여 $\underbrace{\mathrm{P}+\mathrm{P}+\cdots+\mathrm{P}}_{n\text{개}}$을 $n\mathrm{P}$로 나타낸다.

· 집합 $\mathbb{Z}_p$의 정의와 $\mathbb{Z}_p$위에서의 덧셈과 곱셈의 정의를 제시한다.

집합 $\mathbb{Z}_p$에서 p는 소수(prime number)이고, 집합 $\mathbb{Z}_p=\{0,1,2,\cdots,p-1\}$라고 정의한다. 일반적인 정수의 덧셈과 곱셈을 이용하여 집합 $\mathbb{Z}_p$위에서의 덧셈과 곱셈을 다음과 같이 정의할 수 있다.

(ⅰ) $a,b\in\mathbb{Z}_p$에 대하여 $a+b$=(두 정수 a와 b의 합을 p로 나누었을 때 나머지)

(ⅱ) $a,b\in\mathbb{Z}_p$에 대하여 ab=(두 정수 a와 b의 곱을 p로 나누었을 때 나머지)

집합 $\mathbb{Z}_p$ 위에서 정의된 덧셈과 곱셈에 대하여 다음이 성립함을 확인할 수 있다.

(ⅰ) 임의의 $a,b,c\in\mathbb{Z}_p$에 대하여 $(a+b)+c=a+(b+c)$이다.

(ⅱ) 임의의 $a\in\mathbb{Z}_p$에 대하여 $a+0=0+a=a$이다.

(ⅲ) 임의의 $a\in\mathbb{Z}_p$에 대하여 $a+b=b+a=0$인 $b\in\mathbb{Z}_p$가 존재한다. 이때 b를 $-a$로 나타내고 이를 a의 덧셈에 관한 역원이라고 한다. 예를 들어 $\mathbb{Z}_{17}$에서 $-0=0$이고 $-3=14$이다. 한편, 덧셈에 관한 역원을 이용하여 뺄셈을 정의할 수 있다. 자세히 말해서, $x,y\in\mathbb{Z}_p$에 대하여 $x-y=x+(-y)$로 정의한다.

(ⅳ) 임의의 $a,b\in\mathbb{Z}_p$에 대하여 $a+b=b+a$이다.

(ⅴ) 임의의 $a,b,c\in\mathbb{Z}_p$에 대하여 $(ab)c=a(bc)$이다.

(ⅵ) 임의의 $a,b,c\in\mathbb{Z}_p$에 대하여 $a(b+c)=ab+ac$이고 $(a+b)c=ac+bc$이다.

(ⅶ) 임의의 $a\in\mathbb{Z}_p$에 대하여 $a\cdot 1=1\cdot a=a$이다.

(ⅷ) 임의의 $a,b\in\mathbb{Z}_p$에 대하여 $ab=ba$이다.

(ⅸ) 임의의 $a\in\mathbb{Z}_p-\{0\}$에 대하여 $ab=ba=1$인 $b\in\mathbb{Z}_p-\{0\}$가 존재한다. 이때 b를 a^{-1}로 나타내고 이를 a의 곱셈에 관한 역원이라고 한다.

(활동4) $p=17$일 때, 집합 $\mathbb{Z}_p$를 구하고, 위의 정의에 따라 합 $\mathbb{Z}_{17}$에서 정의된 덧셈과 곱셈의 연산표를 구하시오.

(설명) $\mathbb{Z}_{17}=\{0,1,2,\cdots,16\}$이다. 집합 $\mathbb{Z}_{17}$에서 정의된 덧셈과 곱셈의 연산표는 다음과 같이 구할 수 있다.

+	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	0
2	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	0	1
3	3	4	5	6	7	8	9	10	11	12	13	14	15	16	0	1	2
4	4	5	6	7	8	9	10	11	12	13	14	15	16	0	1	2	3
5	5	6	7	8	9	10	11	12	13	14	15	16	0	1	2	3	4
6	6	7	8	9	10	11	12	13	14	15	16	0	1	2	3	4	5
7	7	8	9	10	11	12	13	14	15	16	0	1	2	3	4	5	6
8	8	9	10	11	12	13	14	15	16	0	1	2	3	4	5	6	7
9	9	10	11	12	13	14	15	16	0	1	2	3	4	5	6	7	8
10	10	11	12	13	14	15	16	0	1	2	3	4	5	6	7	8	9
11	11	12	13	14	15	16	0	1	2	3	4	5	6	7	8	9	10
12	12	13	14	15	16	0	1	2	3	4	5	6	7	8	9	10	11
13	13	14	15	16	0	1	2	3	4	5	6	7	8	9	10	11	12
14	14	15	16	0	1	2	3	4	5	6	7	8	9	10	11	12	13
15	15	16	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
16	16	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

·	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2	0	2	4	6	8	10	12	14	16	1	3	5	7	9	11	13	15
3	0	3	6	9	12	15	1	4	7	10	13	16	2	5	8	11	14
4	0	4	8	12	16	3	7	11	15	2	6	10	14	1	5	9	13
5	0	5	10	15	3	8	13	1	6	11	16	4	9	14	2	7	12
6	0	6	12	1	7	13	2	8	14	3	9	15	4	10	16	5	11
7	0	7	14	4	11	1	8	15	5	12	2	9	16	6	13	3	10
8	0	8	16	7	15	6	14	5	13	4	12	3	11	2	10	1	9
9	0	9	1	10	2	11	3	12	4	13	5	14	6	15	7	16	8
10	0	10	3	13	6	16	9	2	12	5	15	8	1	11	4	14	7
11	0	11	5	16	10	4	15	9	3	14	8	2	13	7	1	12	6
12	0	12	7	2	14	9	4	16	11	6	1	13	8	3	15	10	5
13	0	13	9	5	1	14	10	6	2	15	11	7	3	16	12	8	4
14	0	14	11	8	5	2	16	13	10	7	4	1	15	12	9	6	3
15	0	15	13	11	9	7	5	3	1	16	14	12	10	8	6	4	2
16	0	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

(코딩4) SageMath 프로그램을 이용하여 집합 $\mathbb{Z}_{17}$에서 정의된 덧셈과 곱셈의 연산표를 출력하시오.

(코딩과정) 모듈(module) 'sage.matrix.operation_table'에 정의된 함수 'OperationTable'를 불러온 뒤 이를 이용하면 $\mathbb{Z}_{17}$에서 정의된 덧셈과 곱셈의 연산표를 출력할 수 있다. 여기서 'Integers(17)'은 SageMath에 내장된 클래스(class)로서 $\mathbb{Z}_{17}$을 뜻한다.

```
sage: from sage.matrix.operation_table import OperationTable
sage: R = Integers(17)
sage: OperationTable(R, operation=operator.add, names='elements')
```

Figure 6. An example of Activity 4 and Coding 4

(활동5) $\mathbb{Z}_{17}$에서 3의 곱셈에 관한 역원 3^{-1}을 구하시오.

(설명) $\mathbb{Z}_{17}$에서 3의 곱셈에 관한 역원 3^{-1}를 찾기 위해 <표 2>를 이용할 수 있다. '3'행에서 1은 '6' 열에 나타나므로 $3 \cdot 6 = 1$, 즉 $3^{-1} = 6$이다.

(코딩5-1) SageMath 프로그램을 이용하여 $\mathbb{Z}_{17}$에서 3의 곱셈에 관한 역원 3^{-1}을 구하시오.

(코딩5-2) 실제 비트코인 전자거래에서는 소수 $2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$를 이용한다. SageMath 프로그램을 이용하여 $2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$을 십진법으로 나타내시오.

(코딩과정) SageMath에 내장된 클래스(class) 'Integers(17)'과 이 클래스 안에 구현된 메서드(method) 'inverse_of_unit'을 이용하면 $\mathbb{Z}_{17}$에서 3의 곱셈에 관한 역원을 계산할 수 있다.

```
sage: R = Integers(17)
sage: R(3).inverse_of_unit()
6
```

두 번째 줄에서 3의 곱셈에 관한 역원을 정수 전체의 집합 $\mathbb{Z}$가 아닌 $\mathbb{Z}_{17}$에서 고려한다는 의미에서 '3'이 아닌 'R(3)'으로 입력해야 함을 주의하여 한다.

또한 $2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$를 입력하면 십진법으로 나타낸 결과를 얻는다.

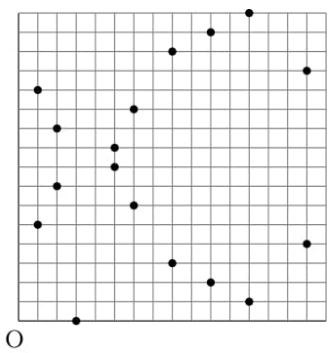
```
sage: 2^256 - 2^32 - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1
115792089237316195423570985008687907853269984665640564039457584007908834671663
```

Figure 7. An example of Activity 5 and Coding 5

(활동6) $\mathbb{Z}_{17}$에서 타원곡선 $y^2 = x^3 + 7$의 해집합을 구하시오.

(설명) 예를 들어 소수 $p = 17$의 경우 다음과 같이 17 개의 해가 존재한다.

$(1,5),(1,12),(2,7),(2,10),(3,0),(5,8),(5,9),(6,6),(6,11),(8,3),(8,14),(10,2),(10,15),(12,1),(12,16),(15,4),(15,13)$



(코딩6) SageMath 프로그램을 이용하여 $\mathbb{Z}_{17}$에서 타원곡선 $y^2 = x^3 + 7$의 해집합을 모두 구하고, 그 결과를 좌표평면위에 나타내시오.

(코딩과정) for문과 if문을 이용할 수 있다. 임의의 원소 (x, y)(단, $x, y \in \mathbb{Z}_{17}$)가 방정식 $y^2 = x^3 + 7$의 해인지 판단하여 해집합을 구한다.

```
sage: R = Integers(17)
sage: S = cartesian_product_iterator([R, R])
sage: C = []
sage: for P in S:
....:     if P[1]^2 == P[0]^3 + 7:
....:         C.append(P)
....:
sage: list(C)
[(1, 5), (1, 12), (2, 7), (2, 10),
 ...
```

두 번째 줄 'S = cartesian_product_iterator_iterator([R, R])'는 S를 $\{(x, y) \mid x, y \in \mathbb{Z}_{17}\}$으로 정의한다는 뜻이다. 'list_plot' 함수를 이용하여 해집합을 좌표평면위에 나타낼 수 있다.

```
sage: list_plot(C)
```

Figure 8. An example of Activity 6 and Coding 6

- 타원곡선 알고리즘을 구체적으로 학습한다.
  · $p=17$일 때, 집합 $\mathbb{Z}_p$위에서의 타원곡선 $y^2=x^3+7$의 해집합을 구하는 방법을 학습한다.

$\mathbb{Z}_p$ 위에서 정의한 덧셈과 곱셈에 관하여 방정식 $y^2=x^3+7$의 해 (x,y) $(x,y\in\mathbb{Z}_p)$를 생각할 수 있다.

  · 집합 E_p의 정의와 덧셈 연산을 정의에 대해 학습한다.

방정식 $y^2=x^3+7$의 해 (x,y) 전체의 집합에 무한원점 I를 추가한 집합인 E 위에서 덧셈을 정의하였다. 여기서는 집합 $E_p=\{(x,y)\,|\,y^2=x^3+7,\ x,y\in\mathbb{Z}_p\}\cup\{\mathrm{I}\}$로 정의한다.

집합 E_p 위에서 덧셈을 다음과 같이 정의할 수 있다.

(ⅰ) $x_\mathrm{P}\neq x_\mathrm{Q}$인 두 해 $\mathrm{P}(x_\mathrm{P},y_\mathrm{P}),\mathrm{Q}(x_\mathrm{Q},y_\mathrm{Q})$에 대하여 $\mathrm{P}+\mathrm{Q}$는 $(m^2-x_\mathrm{P}-x_\mathrm{Q},\ m(x_\mathrm{P}-x_{\mathrm{P}+\mathrm{Q}})-y_\mathrm{P})$로 정의한다. 여기서 $m=(y_\mathrm{Q}-y_\mathrm{P})(x_\mathrm{Q}-x_\mathrm{P})^{-1}$이고 $x_{\mathrm{P}+\mathrm{Q}}=m^2-x_\mathrm{P}-x_\mathrm{Q}$이다.

(ⅱ) $y_\mathrm{P}\neq 0$인 해 $\mathrm{P}(x_\mathrm{P},y_\mathrm{P})$에 대하여 $\mathrm{P}+\mathrm{P}$는 $(m^2-2x_\mathrm{P},\ m(x_\mathrm{P}-x_{\mathrm{P}+\mathrm{P}})-y_\mathrm{P})$로 정의한다. 여기서 $m=3x_\mathrm{P}^2(2y_\mathrm{P})^{-1}$이고 $x_{\mathrm{P}+\mathrm{P}}=m^2-2x_\mathrm{P}$이다.

(활동7) 집합 E_{17}에 속하는 모든 원소를 구하고, $\mathrm{P}(6,6)$을 거듭하여 더하는 결과를 구하시오.

(설명) 소수 $p=17$의 경우 다음과 같이 17개의 해가 존재한다. E_{17}에서 $\mathrm{P}(6,6)$을 거듭하여 더하는 활동을 하면, $\mathbb{Z}_{17}$위에서의 곱셈 연산표를 활용하여 아래와 같이 결과를 얻을 수 있다.

$1\mathrm{P}=(6,6)$,	$7\mathrm{P}=(15,4)$,	$13\mathrm{P}=(10,2)$
$2\mathrm{P}=(1,5)$,	$8\mathrm{P}=(12,1)$,	$14\mathrm{P}=(2,7)$
$3\mathrm{P}=(8,14)$,	$9\mathrm{P}=(3,0)$,	$15\mathrm{P}=(8,3)$
$4\mathrm{P}=(2,10)$,	$10\mathrm{P}=(12,16)$,	$16\mathrm{P}=(1,12)$
$5\mathrm{P}=(10,15)$,	$11\mathrm{P}=(15,13)$,	$17\mathrm{P}=(6,11)$
$6\mathrm{P}=(5,9)$,	$12\mathrm{P}=(5,8)$,	$18\mathrm{P}=\mathrm{I}$

(코딩7) SageMath 프로그램을 이용하여 집합 E_{17}에 속하는 모든 원소를 구하는 과정과 그 결과를 출력하시오. 또한 E_{17}에서 $\mathrm{P}(6,6)$을 거듭하여 더하는 결과를 출력하시오.

(코딩과정) 먼저 $\mathbb{Z}_{17}$ 위에서 $y^2=x^3+7$의 모든 해와 무한원점 I로 구성된 E_{17}을 정의하자. SageMath에 내장된 패키지를 이용하여 유한체 위에서의 연산과 E_{17} 위의서의 연산을 할 수 있다.

```
sage: F = Zmod(17)
sage: E = EllipticCurve(F, [0, 7])
sage: E
Elliptic Curve defined by y^2 = x^3 + 7 over Ring of integers modulo 17
```

'list' 함수를 이용하여 E_{17}의 모든 원소를 다음과 같이 얻을 수 있다.

```
sage: list(E)
[(0 : 1 : 0), (1 : 5 : 1), (1 : 12 : 1), (2 : 7 : 1), (2 : 10 : 1),
(3 : 0 : 1), (5 : 8 : 1), (5 : 9 : 1), (6 : 6 : 1), (6 : 11 : 1),
(8 : 3 : 1), (8 : 14 : 1), (10 : 2 : 1), (10 : 15 : 1), (12 : 1 : 1),
(12 : 16 : 1), (15 : 4 : 1), (15 : 13 : 1)]
```

E_{17}에서의 연산은 아래와 같이 얻을 수 있다. 예를 들어 $(6,6)$과 $(5,9)$의 합, $12(6,6)$은 다음과 같다.

```
sage: E(6,6) + E(5, 9)
(15 : 4 : 1)
sage: 12*E(6,6)
(5 : 8 : 1)
```

이때 E_{17} 위에서의 계산임을 나타내기 원소를 '(6, 6)'꼴이 아닌 'E(6, 6)'꼴로 입력해야 함을 주의하여 한다. 실제로 '(6, 6) + (5, 9)'를 입력하면 '(6, 6, 5, 9)'와 같은 결과를 얻는다. 이는 덧셈(+)을 연결(concatenation) 연산으로 인식하기 때문이다.

이제 'for문'을 이용하여 $(6,6)$을 거듭하여 더한 결과를 얻을 수 있다.

```
sage: for k in [1..18]:
....:     print k*E(6,6)
(6 : 6 : 1)
(1 : 5 : 1)
...
```

Figure 9. An example of Activity 7 and Coding 7

앞에서와 마찬가지로 특수한 경우에는 다음과 같이 정의한다.

(ⅰ) x좌표가 같은 서로 다른 두 해 $\mathrm{P}(x,y),\mathrm{Q}(x,-y)$에 대하여 $\mathrm{P}+\mathrm{Q}=\mathrm{I}$로 정의한다.

(ⅱ) 방정식 $y^2=x^3+7$의 해 $\mathrm{P}(x_\mathrm{P},0)$에 대하여 $\mathrm{P}+\mathrm{P}=\mathrm{I}$로 정의한다.

(ⅲ) 방정식 $y^2=x^3+7$의 임의의 해 P에 대하여 $\mathrm{P}+\mathrm{I}=\mathrm{P}$, $\mathrm{I}+\mathrm{P}=\mathrm{P}$로 정의한다.

(ⅳ) $\mathrm{I}+\mathrm{I}=\mathrm{I}$로 정의한다.

· 기저점의 정의가 무엇인지 제시하고, 비트코인에 사용되는 실제적인 예를 설명한다.

소수 p에 대하여 E_p의 원소의 개수를 $n(E_p)$로 나타내자. 소수 q가 $n(E_p)$를 나눌 때 다음 조건을 만족시키는 E_p의 원소 B가 존재한다는 사실이 알려져 있다.

(ⅰ) $q\mathrm{B}=\mathrm{I}$이다.

(ⅱ) q개의 원소 $\mathrm{B},2\mathrm{B},3\mathrm{B},\cdots,(q-1)\mathrm{B},q\mathrm{B}$ 서로 다르다.

위 사실은 추상대수학(abstract algebra)에서 '코시(Cauchy)의 정리'로 불리는 사실이다. 위 조건을 만족시키는 E_p의 원소 B를 기저점(base point)이라고 한다.

Certicom(2000)에는 비트코인의 거래에서 사용하는 소수 $p=2^{256}-2^{32}-2^9-2^8-2^7-2^6-2^4-1$에 대하여 $\mathbb{Z}_p$ 위에서의 타원곡선 $y^2=x^3+7$에 관련된 변수(parameters)들인 소수 q, 기저점 B 등을 제시하고 이 체계를 'secp256k1'이라 부른다. 이 체계에서 소수 q는

$$115792089237316195423570985008687907852837564279074904382605163141518161494337$$

으로 주어진다.

E_{17}에서 $\mathrm{P}(6,6)$을 거듭하여 더하면 E_{17}의 모든 원소를 얻을 수 있다. 소수 3이 $n(E_{17})=18$을 나누고 $\mathrm{B}=6\mathrm{P}=(5,9)$에 대하여 $1\mathrm{B}=(5,9)$, $2\mathrm{B}=(5,8)$, $3\mathrm{B}=\mathrm{I}$이므로 $\mathrm{B}(5,9)$는 E_{17}의 기저점이다. 이때 B의 순환성에 주목할 수 있다.

비트코인의 거래에 사용되는 체계인 'secp256k1'에서 Certicom(2000)이 제시하는 기저점을 소개한다. 실제 비트코인의 거래에서 이용되는 소수인 $p=2^{256}-2^{32}-2^9-2^8-2^7-2^6-2^4-1$에 대하여 E_p의 기저점 $\mathrm{B}(x_\mathrm{B},y_\mathrm{B})$은 다음과 같다.

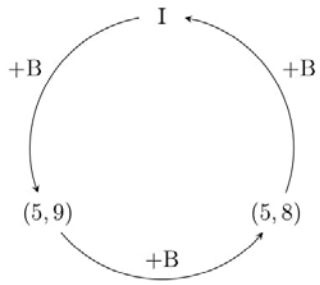


Figure 10. The multiples of the base point B

$$\begin{aligned}(x_\mathrm{B}=&\,550662630222773436695787188951685343262506034537777594175500187360389116729240,\\ y_\mathrm{B}=&\,326705100207588169780830851305070431844712733806592432759389043357573374824 24)\end{aligned}$$

○ **정리 단계의 학습: 수학적 이론을 바탕으로 전자서명의 실제를 소개하고, 학습을 정리한다.**

- 전자서명을 위한 비밀열쇠와 공개열쇠의 의미를 설명하고, 기초 원리를 제시한다.

전자문서에 전자서명을 하려면 '비밀열쇠(secret key, private key)'가 필요하며, 전자서명의 정당성을 인증하려면 비밀열쇠에 대응되는 '공개열쇠(public key)'가 필요하다. 소수 p와 $n(E_p)$를 나누는 소수 q를 고려하자. 타원곡선 전자서명 알고리즘에서 비밀열쇠와 공개열쇠는 다음의 과정에 의하여 생성한다.

(ⅰ) 비밀열쇠 `sk`는 $\mathbb{Z}_q-\{0\}$의 원소를 '임의로' 선택하여 정한다.

(ⅱ) 공개열쇠 `Pk`는 $\mathtt{Pk}=\mathtt{sk}\mathrm{B}=\mathrm{B}+\mathrm{B}+\cdots+\mathrm{B}$ (`sk` 개)로 정의한다. 비밀열쇠는 `sk`는 $\mathbb{Z}_q$의 원소인 반면 공개열쇠 `Pk`는 $\mathbb{Z}_q\times\mathbb{Z}_q$의 원소, 즉 두 개의 성분이 $\mathbb{Z}_q$의 원소인 순서쌍임에 주의하자.

개인의 비밀열쇠는 타인에게 노출되지 않도록 보안에 주의해야 하는 민감한 정보이다. 반면 공개열쇠는 체계 내의 모든 사람들이 접근가능한 정보이다.

- 전자서명을 위한 서명절차를 설명하고, 서명 단계에서 k를 임의로 선택하는 이유를 제시한다.

· 전자서명을 위한 서명절차를 설명한다.

서명할 문서를 msg라고 하자. 이때 msg는 $\mathbb{Z}_q$의 원소로 가정한다. 실제로 비트코인 거래 체계에서는 거래의 내용 혹은 내용의 일부를 SHA256이라고 부르는 해시함수(hash function)를 이용하여 $\mathbb{Z}_q$의 원소로서 msg에 해당하는 값을 얻는다.

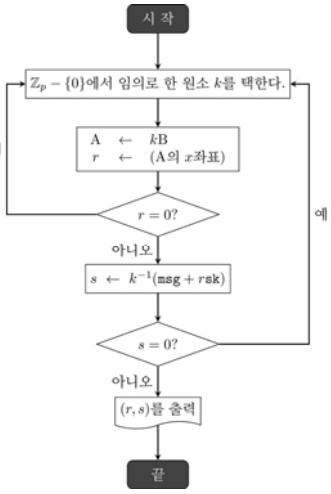
비밀열쇠 sk와 서명할 문서 msg로부터 다음 과정을 따라 전자서명 (r, s)를 얻는다.

(i) $\mathbb{Z}_q - \{0\}$에서 '임의로(randomly)' 원소 k를 뽑는다.

Figure 11. An example of a transformation with SHA256

(활동8) 전자서명 (r, s)를 얻는 과정을 순서도로 표현하시오.

(설명) 전자서명의 4단계 과정을 순서도(flow chart)로 나타내면 다음 그림과 같다. 이러한 순서도를 거치면 문서를 받을 사람에게 msg와 함께 전자서명 (r, s)를 함께 보낸다.



(코딩8) SageMath 프로그램을 이용하여 E_{17}에서 기저점을 $B(5, 9)$로 하였을 때, 전자서명 (r, s)를 계산하는 활동 과정을 수행하고, 그 결과를 제시하시오. 이때, 비밀열쇠가 sk=2, 공개열쇠가 Pk=2B=(5, 8)인 사람이 문서 msg=?에 전자서명을 하는 상황을 가정한다.

(코딩과정) while문을 이용하여 전자서명을 계산하는 함수 'ECDSA'를 정의할 수 있다.

```
sage: R = Integers(17)
sage: S = Integers(3)
sage: E = EllipticCurve(R, [0, 7])
sage: Z = Integers()
sage: B = E(5, 9)
sage: def ECDSA(sk, Pk, msg):
....:     k = 0
....:     r = 0
....:     s = 0
....:     while s == 0:
....:         while r == 0:
....:             while k == 0:
....:                 k = S.random_element()
....:             A = Z(k)*B
....:             r = A[0]
....:         s = k.inverse_of_unit() * (S(msg) + S(r) * S(sk))
....:     print(r, s)
```

위와 같이 정의된 함수 'ECDSA'를 이용하여 비밀열쇠와 공개열쇠가 각각 2, (5, 8)인 사람의 문서 1에 대한 전자서명을 구할 수 있다.

```
sage: ECDSA(2, E(5, 8), 1)
(5, 2)
```

Figure 12. An example of Activity 8 and Coding 8

(ⅱ) 앞에서 뽑은 k와 기저점 B을 이용하여 A와 r를 다음과 같이 정의한다.

$$\mathrm{A}=k\mathrm{B}, \qquad r=(\mathrm{A}\text{의 } x\text{좌표})$$

(ⅲ) $r=0$이면 다시 [1단계]로 돌아가고, $r\neq 0$이면 서명할 문서 msg와 비밀열쇠 sk를 이용하여 s를 다음과 같이 정의한다.

$$s=k^{-1}(\mathtt{msg}+r\,\mathtt{sk})$$

(ⅳ) $s=0$이면 다시 [1단계]로 돌아가고, $s\neq 0$이면 (r,s)를 출력한다.

· 서명 단계에서 k를 임의로 선택하는 이유를 제시한다.

전자서명을 할 때마다 [서명-1단계]에서 k를 임의로 선택해야 하는 이유는 다음과 같다.

(ⅰ) 두 개의 서로 다른 문서 $\mathtt{msg}_1$와 $\mathtt{msg}_2$에 대하여 같은 k를 이용하여 전자서명을 얻었다고 하고 수신자가 이 사실을 안다고 가정하자.

(ⅱ) 같은 k를 이용하였으므로 두 문서의 전자서명의 첫 번째 성분(r의 값)은 같다. $\mathtt{msg}_1$와 $\mathtt{msg}_2$의 전자서명을 각각 (r,s_1), (r,s_2)라고 하자.

(ⅲ) $s_1=k^{-1}(\mathtt{msg}_1+r\,\mathtt{sk})$, $s_2=k^{-1}(\mathtt{msg}_2+r\,\mathtt{sk})$로부터 등식 $s_1-s_2=k^{-1}(\mathtt{msg}_1-\mathtt{msg}_2)$가 성립함을 알 수 있다. 그런데 s_1, s_2, $\mathtt{msg}_1$, $\mathtt{msg}_2$ 모두는 수신자가 알고 있는 정보이므로 이를 이용하여 수신자는 k의 값을 구할 수 있다. 실제로 $k=(s_1-s_2)^{-1}(\mathtt{msg}_1-\mathtt{msg}_2)$이다.

그런데 k의 값이 수신자에게 알려지면 등식 $\mathtt{sk}=r^{-1}(ks_1-\mathtt{msg}_1)$로부터 비밀열쇠가 노출되기 때문이다.

- 전자서명을 위한 인증절차를 설명한다.

· 전자서명을 위한 인증절차를 설명한다.

보낸 사람의 전자서명 (r,s)과 함께 msg를 받은 사람은 보낸 사람의 공개열쇠 Pk와 msg로부터 수신된 문서의 전자서명 (r,s)가 정당한지 인증해야 한다. 다음의 과정을 통하여 전자서명의 정당성을 인증할 수 있다.

(ⅰ) msg와 전자서명 (r,s)를 이용하여 u_1과 u_2를 다음과 같이 정의한다.

$$u_1=s^{-1}\mathtt{msg},\ u_2=s^{-1}r$$

(ⅱ) 기저점 B, 전자서명한 사람의 공개열쇠 Pk와 함께 앞서 구한 u_1, u_2를 이용하여 Q를 다음과 같이 정의한다.

$$\mathrm{Q}=u_1\mathrm{B}+u_2\mathtt{Pk}$$

(ⅲ) $r=$(Q의 x좌표)이면 '인증성공'을 출력하고, $r\neq$(Q의 x좌표)이면 '인증실패'를 출력한다.

· 전자서명을 위한 인증절차의 정당화과정을 설명한다.

인증절차의 예에서 다음을 확인하도록 한다. 수신자는 발신자의 비밀열쇠 sk와 서명절차의 (ⅰ)에서 임의로 선택한 k의 값을 모두 모르지만 오직 발신자의 공개열쇠 Pk와 문서 msg만으로 $\mathrm{A}=k\mathrm{B}$의 x좌표 r를 구할 수 있다. 인증절차에서 $\mathrm{A}=k\mathrm{B}$의 x좌표 r를 구하는 방식이 타당한지 확인하도록 한다.

$$\begin{aligned}\mathrm{Q} &= u_1\mathrm{B}+u_2\mathtt{Pk} = u_1\mathrm{B}+u_2\mathtt{sk}\mathrm{B}\\ &= (u_1+u_2\mathtt{sk})\mathrm{B} = (s^{-1}\mathtt{msg}+s^{-1}r\,\mathtt{sk})\mathrm{B}\\ &= s^{-1}(\mathtt{msg}+r\,\mathtt{sk})\mathrm{B} = k\mathrm{B} = \mathrm{A}\end{aligned}$$

· 소집단별로 전자서명과 일반서명의 차이점을 토의하도록 한다.

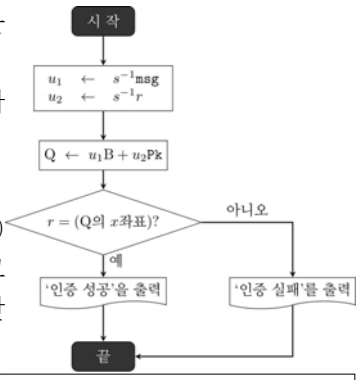
(토의주제) 전자서명과 일반서명의 차이점이 무엇인지 다음 과점에서 논의하시오.

발신자가 문서 msg에 전자서명 (r,s)을 하여 수신자에게 보냈을 때, 수신자는 발신자의 서명 (r,s)를 다른 문서에 발신자의 서명인 척 속여 사용할 수 있을까?

위 질문에 대하여 다음과 같은 답을 할 수 있다. 서명절차에서 전자서명 (r,s)는 msg의 값에 영향을 받는다. 따라서 문서의 내용이 바뀌면 같은 사람의 서명일지라도 전자서명 (r,s)의 값이 바뀐다. 일반서명은 같은 사람이 모든 문서에 같은 모양의 서명을 하는 반면, 전자서명은 재사용이 불가능하다.

(활동9) 앞에서 제시한 전자서명 (r,s)가 올바른지 인증하는 과정을 순서도로 표현하시오.

(설명) 전자서명을 인증하는 과정을 순서도로 나타내면 다음 그림과 같다.



(코딩9) E_{17}에서 기저점을 $B(5,9)$로 하였을 때, 공개열쇠가 $Pk=(5,8)$인 사람으로부터 문서 $msg=1$와 전자서명 $(5,2)$을 받았다고 가정하자. SageMath 프로그램을 이용하여 인증절차 각 단계별 수행하고, 그것이 옳은지 확인하시오.

(설명) 입력 값으로 보내온 사람의 공개키, 문서, 전자서명을 가지는 함수 'Verify'를 다음과 같이 정의한다.

```
sage: R = Integers(17)
sage: S = Integers(3)
sage: Z = Integers()
sage: E = EllipticCurve(R, [0, 7])
sage: B = E(5, 9)
sage: def Verify(Pk, msg, r, s):
....:     u1 = S(s).inverse_of_unit() * S(msg)
....:     u2 = S(s).inverse_of_unit() * S(r)
....:     Q = Z(u1)*B + Z(u2)*Pk
....:     if r == Q[0]:
....:         print '인증 성공'
....:     else:
....:         print '인증 실패'
```

위와 같이 정의된 함수를 이용하여 전자서명을 인증할 수 있다.

```
sage: Verify(E(5, 8), 1, 5, 2)
인증 성공
```

Figure 13. An example of Activity 9 and Coding 9

V. 결론 및 논의

본 연구의 목적은 코딩교육과 관련이 깊은 정보교과와 수학이 융합한 교수·학습 자료의 개발과 수업모형의 개발이다. 이를 위해, 현재 정보 분야에게 가장 이슈가 되고 있는 '비트코인(bitcoin)과 이 비트코인의 전자결재'와 관련된 주제를 선정하여, 코딩과 연계한 수학·정보 교과의 융합 연구를 수행하였다.

본 연구에서는 수학과 정보의 융합교육을 실현하기 위해 달성되어야 할 것으로, 첫째, 코딩을 위한 프로그램의 이해, 둘째, 융합교육 주제에 대한 수학적 이해 및 이를 기반으로 한 특정 정보 주제에 대한 개념 획득, 셋째, 학습결과에 대한 실제적 활용으로 규정하였다. 이 세 가지의 실현을 위해 복합적 수업모형으로 사전 수업활동(Pre-Class) 단계, 본시 수업활동(In-Class) 단계, 사후 수업활동(After-Class) 단계로 구분하여 수업모형을 개발하였다.

이러한 수업모형에 따라 코딩과 연계한 교수·학습 자료에 대한 개발 방향을 설정하였다. 첫째, 전자서명에 관계된 수학 중, 논리적 사고력 및 개념에 대한 추상화에 최적의 내용을 선정하고, 둘째, 코딩을 위해 사용하는 프로그래밍 명령어 및 문법의 난이도를 분석하여 사범대학 예비교사, 일반계 및 과학고등학교 학생 등에 적절한 명령어 및 문법 내용을 선정하여 프로그램을 개발하고, 셋째, 교수·학습의 순서는 수학적 문제 상황의 난이도와 프로그래밍 내용의 난이도를 고려하여 점진적으로 심화하는 것으로 구성하고, 넷째, 학습자가 전자서명에 담긴 수학뿐만 아니라, SageMath 프로그램 코딩에도 흥미를 가질 수 있도록 난이도를 구성하며, 다섯째, 사범대학 수학교육과 기초 정수론 강의, 과학고등학교 및

일반고등학교 수학동아리에서 실제적으로 활용할 수 있도록 설계하는 것이다.

또한, 본 교수·학습 자료의 구체적인 개발을 위한 원칙을 제시하였다. 첫째, 개념학습을 위한 교수·학습의 원칙, 둘째, PERC에 부합된 교수·학습의 원칙, 셋째, 활동학습에 기반한 교수·학습의 원칙이다. 특히, 수학 및 정보교과 관련 학습내용에서 다루어지는 개념이 추상화과정을 통해 분명히 획득되어질 수 있도록 구성하였다. 또한 수학활동과 코딩활동의 병렬식 배치를 통해 다양한 소집단 활동, 토의활동이 이루어지도록 하였다.

개발된 수업모형 및 교수·학습 자료는 수학전문가, 수학교육전문가, 현장교육전문가에 의해 평가되어졌다. 이러한 평가결과는 환류과정을 통해 교수·학습 자료 및 수업모형을 전문가의 요구에 부합되도록 수정하는 이론적 근거로 활용하였다. 평가결과 대부분의 문항에서 높은 점수를 받았지만, 상대적으로 두 문항에서 낮은 반응을 보여, 전문가의 의견을 충분히 반영하여 최종 교수·학습 자료를 완성하였다. 완성된 수업모형 및 교수·학습 자료는 본 연구의 결과에서 요약하여 제시하였다.

결론적으로, 블록체인과 이에 기반을 둔 비트코인에서 핵심적인 기술은 전자서명이다. 서명은 남이 위조할 수 없어야 하거니와 서명한 사람 본인이 자기 서명이 아니라고 부인할 수도 없어야 하는 등 여러 기능을 가지고 있어야 한다. 이 모든 기능을 수학으로 구현하여야 하므로 관련된 수학은 고등학교 수학의 범위를 벗어나는 것은 자연스러운 일일 것이다. 따라서 고등학교 수업에서 전자서명 등 비트코인에 필요한 모든 수학을 엄밀하게 소개하는 것은 용이한 일이 아니다. 그러나 어느 수준의 수학적 능력을 가진 학생이라면 어떤 수학이 어떻게 활용되는지에 관해 궁금할 것이다. 이 논문에서 제시하는 내용과 방식은 그러한 목적을 달성할 것으로 사료된다. 이에 본 연구에서 개발된 수업모형 및 교수·학습 자료는 수학과 정보 교과의 융합교육이 필요한 상황에서 일반계 고등학교의 동아리, 과학고등학교, R & E 활동, 대학교 '정수론' 강좌 등에 유용하게 활용될 것으로 기대된다.

참 고 문 헌

Baek, H. Y. (2018). *Analysis on the situated abstraction process in mathematics coding class* (Master's thesis). Korea National University of Education, Chung-Buk, Korea.

Baek, Y. S., Park, H., Kim, Y., Noh, S. G., Park, J.-Y., Lee, J., Jeong, J.-S., Choi, Y., & Han, H. (2010). STEAM education in Korea. *Journal of Learner-Centered Curriculum and Instruction, 11*(4), 149-171.

British Department for Education (2013). *National curriculum in England: computing programmes of study* (2013. 9. 11.)

Camp, J., & Marchionini, G. (1984), Programming and learning: Implications for mathematics education, In V. P. Hanson (Ed.), *Computers in mathematics Education.* Reston, VA: National Council of Teachers of Mathematics.

Chang, K. Y. (2017). A feasibility study on integrating computational thinking into school mathematics. *Journal of Korea Society Educational Studies in Mathematics School Mathematics, 19*(3), 553-570.

Choi, K. S. (2012). *A study on the development and effects of project learning programs of a home economics subject for creativity & character education -Focusing on a 'residential space utilization'* (Master's thesis). Korea National University of Education, Chung-Buk, Korea.

CSTA Standard Task Force (2011). *CSTA*

K-12 Computer Science Standards. Revised 2011, 1-73.

Han, I. (2006). *수학교육의 기초와 실제* [Fundamental theory and Practice of Mathematics Education]. JinJu: Gyeongsang National University Press.

Hershkowitz, R., Schwarz, B., & Dreyfus, T. (2001). Abstraction in context: epistemic actions. *Journal for Research in Mathematics Education, 32*(2), 195-222.

Kang, H. (2004). *Analysis of children's logical thinking improvement with basic programming ability* (Master's thesis). Sookmyung Women's University, Seoul, Korea.

Kim, J., Shin, J., Park, B., Seo, B., Lim, M., Lim, S., Lee, J. Kim, Y., Kim, J., Suh, B. E., Yang, M., & Kim, K. (2017). *2017 창의교육 거점센터 운영에 관한 연구* [The study of 2017 regional center for creativity education] (AD18020004). Seoul, Korea: Korea Foundation for the Advancement of Science and Creativity.

Koo, D. (2014). Development of digital storytelling education program based on software programming. *The Journal of Korea Elementary Education, 25*(1), 245-260.

Kwon, S. (2018). *Computational Thinking 교육을 위한 수학교과와 정보교과와의 융합인재교육(STEAM) 자료 개발: 2015개정 중학교 1학년 수학을 중심으로* [Development of STEAM program contained mathematics and information for computational thinking education: Focused on mathematics in 2015 revision curriculum middle school first grade] (Master's thesis). YONSEI University, Seoul, Korea.

Lee, H. J. (2012). *A study on development of programming program for elementary gifted children of information in math curriculum* (Master's thesis). Seoul National University of Education, Seoul, Korea.

Lee, K. H. (1996). *The Application and evaluation of computer educational progrmming language 'SAM' in programming education* (Master's thesis). Korea National University of Education, Chungbuk, Korea.

Lee, S. R. (1992). *A study on the development of mathematics worksheets using computer programming for eight graders in Korea* (Master's thesis). Korea National University of Education, Chungbuk, Korea.

Lee, S.-M. (2009). *An analysis on cases of PBL-based computer programming learnings* (Master's thesis). Sookmyung Women's University, Seoul, Korea.

Ministry of Education [MOE]. (2015). *2015 revised mathematics curriculum*. Sejong, Korea: Author.

Ministry of Education [MOE]. (2015). *2017년도 SW교육 연구선도학교 1,200개 선정 결과 발표* [Announcement of the selection result about 2017 SW education 1,200 model schools] (MOE press release 2017.3.8). Sejong, Korea: Author.

Moon, W. (2013). STEAM learning model in elementary schools by applying SCRATCH programming. *Journal of The Korean Association of Information Education, 17*(4), 457-466.

National Science Foundation (2010). *Science and Engineering Indicators 2010, NSF*.

Noss, R., Hoyles, C., & Pozzi, S. (2002).

Abstraction in expertise: A study of nurse' conceptions of concentration. *Journal for Research in Mathematics Education, 33*(3), 204-229.

Park, S. (2002). *Mathematics education for computer science* (Master's thesis). Chung-Ang University, Gyeonggi, Korea.

Park, Y. M. (2006). *Programming curriculum and teaching method related to mathematics education system to enhance problem solving ability* (Master's thesis). Chungbuk National University, Cheongju, Korea.

Sanders, M. (2009). STEM, STEM education, STEM mania. *Technology Teacher, 68*(4), 20-26.

Shin, H., & Shin, K. (2017). *대칭: 갈루아 이론* [Symmetry: theory of Galois]. Cheonju: mathesign.

Shin, H., & Shin, K. (2019). *정수와 대수: 암호, 부호* [Number theory and algebra: Cryptographs, and coding theory]. Cheonju: mathesign.

Shin, J. (2012). *A study of project method in technology education for developing students' creativity and personality - Centering on the lesson for electricity and electronic technology* (Master's thesis). Kyungpook National University, Daegu, Korea.

Shin, J. (2015). *A study on the development of mathematical teaching-learning materials for character education in middle school mathematics classes* (Unpublished doctoral dissertation). Korea National University of Education, Chung-Buk, Korea.

Shin, J. K., Boo, D. H., & Suh, B. E. (2015). A study on the development of teaching and learning materials for character education in middle school. *Communications of Mathematical Education, 29*(2), 255-279.

Yoo, J. H. (2015). *Development of coding education program based on mathematics content* (Master's thesis). Seoul National University of Education, Seoul, Korea.

국 문 요 약

이 연구는 수학과 정보교과의 융합교육을 위한 시도로, 최근 강조되고 있는 코딩교육을 수학교육에 접목한 교수학습자료의 개발연구이다. 코딩교육을 위한 수학주제로 전자서명을 선택하였고, 코딩을 위한 프로그램으로는 SageMath이다. 본 연구에서 전자서명의 다양한 방법 중 타원곡선 전자서명 알고리즘에서 이용되는 수학을 조명하고, 이를 소재로 정보와 수학교과의 융합 교수·학습 자료를 코딩기반으로 개발하였다. 최근에 많은 사람들이 관심을 가진 비트코인의 거래에서 실제로 활용되는 타원곡선 전자서명 알고리즘은 수학이 응용되는 실례로서 학생들에게 보여주기 좋은 소재이고, 코딩으로 구현하기에도 최적의 환경을 제공해 주고 있다. 따라서 이를 소재로 한 수업은 수학중심의 융합교육을 실현할 수 있는 구체적인 교수·학습 프로그램을 제공할 것으로 기대된다. 또한 이 연구에서 제시된 교수·학습 프로그램은 수학자, 현장수학교사, 수학교육 전문가의 의견을 종합적으로 고려하여 수정 보완하여 완성함으로써 과학고등학교, 수학동아리, 대학교 '정수론' 강좌 등에서 유의미한 수업으로 구현될 것으로 기대된다.

주제어: 수학적 모델링, 융합교육, 수학과 정보, 코딩교육, 수학교수-학습자료, 전자서명