

SW 비전공자 대상으로 지능형 데이터 코딩 교육과정 설계 : EZMKER kit교구 중심으로

장승영*

Designing an Intelligent Data Coding Curriculum for Non-Software Majors:
Centered on the EZMKER Kit as an Educational Resource

Seoung-Young Jang*

요약

대학에서는 4차 산업혁명에 맞추어 디지털 융합시대를 이끌어갈 창의·융합 인재를 육성하기 위하여 프로그래밍 언어적 사고를 기반으로 SW교육을 비전공자 대상으로 운영하고 있다. 하지만 학습자들은 프로그래밍 언어의 문법과 생소한 프로그래밍 언어를 습득하는 과정에 난점을 겪고 있다. 본 연구에서는 SW 비전공자들에게 학습과정에서 고충을 해소하기 위해서 소프트웨어 교육 모형을 제안하는 데 목적을 두었다. 프로그래밍 언어사고를 기반으로 EZMKER kit 교구 교육모델 중심을 알고리즘 기술과 다이어그램 기술을 도입하여 프로그래밍 언어와 문법에 대한 학습부족을 극복하고 구조적 소프트웨어 교육모델을 Top-Down시스템 학습모델로 설계하여 구현하게 되었다.

ABSTRACT

In universities, programming language-based thinking and software education for non-majors are being implemented to cultivate creative and convergent talent capable of leading the digital convergence era in line with the Fourth Industrial Revolution. However, learners face difficulties in acquiring the unfamiliar syntax and programming languages. The purpose of this study is to propose a software education model to alleviate the challenges faced by non-major students during the learning process. By introducing algorithm techniques and diagram techniques based on programming language thinking and using the EZMKER kit as an instructional model, this study aims to overcome the lack of learning about programming languages and syntax. Consequently, a structured software education model has been designed and implemented as a top-down system learning model.

키워드

Computational Thinking, Educational Model, Problem Solving, Software Education
computational 컴퓨팅적 사고, 교육 모형, 문제해결, 소프트웨어 교육

* 교신저자: 전남과학교육정보전산원
• 접수일 : 2023. 08. 16
• 수정완료일 : 2023. 09. 13
• 게재확정일 : 2023. 10. 17

• Received : Aug. 16, 2023, Revised : Sep. 13, 2023, Accepted : Oct. 17, 2023
• Corresponding Author : Seung-Young Jang
Dept. Department of Business Information, NAMBU National University
Email : sychaing@cntu.ac.kr

1. 서 론

미래 사회는 지능정보기술을 기반 융·복합 사회구조로 과학적소양을 함양한 창의·융합형 인재양성이 강조된다.

과학소양은 제4차 산업혁명시대의 기초소양으로 강조되고 국가 경쟁력 강화에 핵심교과로 인식되고 있다. 이러한 이유로 2016년 스위스의 다보스(Davos)에서 열린 세계경제포럼(WEF)에서 ‘혁신적인 기술과 디지털화로 인한 과학소양’을 21세기 기초소양으로 제안한 바 있으며, 2011년도 한국교육개발원에서 발표한 ‘21세기 창의적 인재 양성을 위한 교육의 미래전략 연구’ 보고서에서 미래를 전망한 창의적 인재 양성을 위한 전략적인 방향제시를 통해 강조하고 있다[1]. 첨단 과학기술과 융·복합기술의 수요에 대응할 수 있는 미래형과학 교육기반 확대가 필요에 맞게 지능정보기술을 활용한 학습 콘텐츠개발 적용 및 개인의 특성을 고려한 맞춤형 교육강화가 필요하다는 목소리가 높아지고 있다.

교육부, 교육기본법 제22조 및 과학 수학 정보 교육진흥법은 ‘과학교육환경은 과학적 소양, 과학적 지식·탐구능력 및 과학적 창의력을 키울 수 있도록 조성되어야 한다’고 명시하면서 과학교육에 대한 국가의 역할과 책무성을 강조하고 있다[2]. 이는 첨단 기술을 활용하여 학생들이 적극적으로 참여할 수 있는 과학탐구 공간을 조성하고 지원여건을 마련하는 것에 주요한 과제로 공표하고 있다.

과학 기술의 발전속도가 가속화되면서 창의성이 경제적 가치의 주요 추진력이 되어 창의적 인재양성에 초점을 맞춘교육 및 경제정책을 펼치고 있다. 창의적 인재 양성을 위해서는 지식위주 교육보다는 지식을 찾고 활용할 수 있는 역량중심 교육이 커지고 있다. 이로 인해 교육환경 변화는 대학교육에도 영향력이 점차 커지게 되었다. 과거의 암기중심에서 창의력과 협동력 그리고 문제해결 능력을 강조하는 방향으로 전환이 요구되었다. 또한 지식기반 경제에서 요구하는 새로운 패러다임의 ‘교육학습사회’에 관심이 집중되었다. 지식정보화 기반을 두고 혁신 주도적 교육개혁의 전환은 노동 유연화가 가속화하고 평생직업 교육에 대한 관심이 높아지고 있다[3].

2025년부터는 초·중등교육에서 컴퓨터 언어를 사용

하여 프로그램 개발과정의 ‘코딩’ 교육에 대한 의무화 방침을 포함한 정책기조의 틀을 내놓는다. 2024부터는 디지털 및 첨단 기술분야에서 대학입학 후 5년6개월(11학기) 만에 학사, 석사, 박사 학위를 통합하여 취업을 위한 교육 프로그램 연계체계 구축을 2026년까지 인공지능(AI), 사물인터넷(IoT)등 첨단 기술분야에 특화된 인력양성을 위한 제도적 기반조성의 개혁안으로 발표하였다[4].

이러한 맥락에서 본 연구의 주된 관점은 교육정책의 기본방향에 기반을 둔 소프트웨어(Software) 역량 교육 정책기조와 비전공학과를 위한 지능형 SW교육 시스템의 핵심요소를 중심으로 창의적 인재육성을 준비하는데 구체적인 지침을 논의하고자 한다.

현재, 추진중에 있는 지능형 SW교육과 관련된 국내 연구동향을 살펴보면, SW교육과 STEAM교육(STEAM Literacy)의 결합적인 학생의 창의력, 혁신능력 증진에 초점을 둔 연구가 활발히 진행되고 있다. 특히, 초·중·고 학생들을 대상으로 진행하는 코딩 프로그래밍 교육의 효과성에 대한 논의와 연구가 가속화하고 있다. 그러나 교육현장에 지능형 SW교육을 위한 교육과정 설계에 대해 미비하다는 견해가 있다[5]. 전반적인 창의성 함양을 위한 교육과정을 보다 구체적으로 실현하기 위한 방안으로서 지능형 SW교육에 체계적인 교육과정 개발을 용이하게 함에 있어 제안하게 되었다.

첫째, 체계적인 교육과정의 관계성 여부의 개발연구가 필요하다.

둘째, 교사 역량기술을 기반으로 한 창의력 요소 필요하다.

셋째, 상호 협력을 통한 새로운 가치창출에 대한 지식융합 가능성이 필요하다.

넷째, 타당성을 높일수 있는 학과구조의 피드백 설계가 필요하다.

다섯째, 인터넷과 클라우드 기술이 적용된 교육 콘텐츠에 접근 방법이 필요하다.

따라서 이 연구에서는 지능형 SW교육에 적합한 프로그램 언어기반의 학습특성, 도구의 사용, 컴퓨팅 사고력과의 관련성 측면으로 구조적 교육모델을 제안하고 비전공자들의 학습 효과를 극대화할 수 있는 교육모델을 개발·적용할 수 있도록 제시하고자 한다[6].

II. 지능형 소프트웨어 교육의 개념

2.1 프로그래밍 언어적 사고

프로그래밍 언어적 사고란, 컴퓨터 프로그래밍을 활용하여 문제를 해결하는 과정에서 필요한 논리체계의 방식이다.

1. 문제 인력
2. 알고리즘 학습력
3. 코드 문법적 이해력
4. 디버깅 문제해결력
5. 리팩터링 적용력

표 1. 프로그래밍 언어 사고의 구성 요소
Table 1. The components of programming language thinking

Component	Description
Logical thinking	The ability to understand and analyze problems and identify logical structures.
Abstraction	The ability to simplify problems or objects and extract relevant components.
Algorithm design	The ability to design specific procedures or methods to solve problems.
Pattern recognition	The ability to recognize and apply common patterns in similar problems.
Debugging	The ability to find and fix errors that occur during program execution.
Computational thinking	The ability to transform problems or data into a format that computers can understand.
Application	The ability to actively apply learned knowledge to real-world problem solving.

표 1과 같이 프로그래밍 언어적 사고는 인지과학, 환경과학, 시스템 이론, 사회과학, 인문학까지 포함하여 정보통신기술(IT)을 활용한 문제해결을 위한 융합 지식이다[7].

대학은 혁신적인 교육내용과 진보적인 교육방법을 도입하는 등의 ‘창의적 교육과정 확립’을 목표로 체계화된 거버넌스 구조(Governance Structure)를 확립이

중요한 목표가 되어야 한다. 대학교육의 새로운 패러다임으로 제도적 제약을 벗어난 폭넓은 주도학습(Project-Based Learning)이 주어지는 방안이 마련되어야 한다.

현행 주입식 교육 시스템체계를 벗어나 주도학습과 프로그래밍 언어사고를 유도하는 교육과정의 융합지향적 패러다임으로 전환하기 위한 방도가 필요하다. 이를 위해서 논리적이고 체계적방식으로 개방·공유·협업의 복합적 구성과 지식의 활용, 새로운 가치의 창출, 다양성의 네트워크 내에서 효과적으로 지원할 수 있는 역할이 수행해야 한다[8].

2.2 대학교육에서 프로그래밍 언어적 사고

2021년 교육부에서 발표한 ‘정보교육 종합계획안’은 SW정규교육을 위한 인력 및 인프라구축 등 체계적인 SW 교육환경 조성사업 기본계획을 발표했다. 이와 관련하여 21세기 세계화·지식기반사회의 도래와 더불어 4차 산업혁명의 도래로 인해 컴퓨터 프로그래밍 언어사고력 중요성이 강조되고 대학교육의 내용과 방법에 변화가 나타나고 있다. 소프트웨어 교육을 교양필수 교과목으로 편성·운영하여 비전공자들에게 창의, 인성, 지식의 핵심 역량을 겸비하기 위한 가치능력을 확대하고 있다.

기존연구에서 언급한 ‘SW기초과목 개설에 관한 수강 의향이 있는지를 묻는 질문’에서 응답자의 65.3% 학생들이 수강 의사에 답변하고 있다[9]. 이러한 통계적 관계가 나타나는 이유는 대학 졸업자들의 취업난 문제의식에 대한 변화로 분석하고 있다. 대학은 프로그래밍 언어적 사고중심의 교육을 활용하여 각 수준에 맞는 개별화된 교육내용을 개편하고 교육전반의 교육시스템을 재구현할 수 있는 주제중심의 ‘융합과학’ 과목을 도입하고 있다. A대학은 ‘컴퓨터와 문제해결’과 같은 교양 교과목을 편성하여 전 교생 대상으로 운영하고 있으며, B대학은 ‘사물 인터넷의 이해와 활용’ 등의 프로그램을 운영하고 있다. 또한, 교양필수 대한 추이분석을 통해 사례연구에서도 긍정적인 효과 검증이 이루어졌다.

지식과 기술을 가르치고 배우는 과정에서 의사소통 능력, 협동능력, 리더십, 문제해결력 등이 취업에 영향력을 미치는 요인이며, 교육 프로그램 통해서 해소되고 산업과 사회의 요구에 대응할 수 있는 연구결과를 확인하게 된다[10].

2.3 학습 고려사항

프로그래밍 언어적 사고학습은 학습수준에 맞게 교육과정 운영제시와 창의성 신장을 위한 수업이 가능하도록 교육과정 운영의 다양화를 구축해야 한다. 학습평가 면에서도 시험점수 중심의 평가방식을 탈피하고 문제해결력과 창의력, 분석력, 잠재력 등을 종합적으로 평가지표에 포함하도록 구성한다.

1. 사용자중심 설계(User-Centered Design, UCD)
2. 표준화모듈화(Standardization-Modularity)
3. 단계적 구조(Hierarchical Structure)
4. 반복적 개선(Iterative Improvement)

표 2. 프로그래밍 언어적 사고학습 고려요소
Table 2. Considerations for learning programming languages thinking

Item	Description
Understanding learner's level and background knowledge	Selecting suitable programming language and educational content
Setting educational objectives	Understanding learning objectives and designing lessons
Emphasizing practical content	Enhancing knowledge and skills applicable to real tasks
Project-based learning	Engaging in projects for developing collaboration and problem-solving skills
Importance of self-directed learning	Establishing an educational environment that supports independent learning and skill acquisition
Providing continuous feedback	Utilizing feedback to support learners' growth through evaluation and improvement

소프트웨어 교육의 효율성을 높이기 위해서는 맞춤형 교육 프로그램개발, 인프라구축 방법제시, 메가트렌드분석(Mega-trend Analysis)방법 등을 중심으로 학습프로그램 모델을 개발한다.

첫째, 학생들의 동기부여와 학습사항을 파악하고 이를 토대로 교육과정과 내용을 수정·보완의 과정을

통해 교육의 질을 개선해야 한다.

둘째, 교육지원 체계의 구축은 소프트웨어 교육을 연계할 수 있는 융합교육 프로그램을 개발·보급하고 교육연구 지원을 확대할 계획을 수립한다.

셋째, 교육과정에 스마트기술을 적극 도입하여 학습 프로세스를 구축을 통해 학생 참여형 수업이 이루어지도록 한다.

넷째, 소프트웨어교육의 차별화로 인해 야기되는 교육의 형편성 문제해결과 새로운 가치를 창출하는 네트워크형 인재를 양성하는 목표로 설정한다[11].

표 2와 같이 학부 교양과정 역할은 교수 주체로 운영되는 폐쇄적인 교육과정이 아닌 학생의 참여와 개발적 교육과정 구성체제로 전환하고 학문의 기본지식을 교육과정 연계의 입체적인 구축, 도입, 개발, 전달을 통해 창의 능력을 함양하고 다양한 관점수용에 대한 이해의 폭을 넓히는 공동체적 가치를 실현하려는 지식양성을 교육목표로 삼아야 할 것이다.

III. 교육과정 설계

본 연구에서 제안하는 교육과정 설계를 위한 SW 기초소양 중심으로 실제 수업을 설계하는 데 프로그래밍 언어사고의 방향성을 정립하고 학습설계 모델의 가이드라인을 제시한다. 단계 정립을 위해서 기존 학습모델의 핵심이 되는 과정을 기본에 두고 프로젝트 중심의 내용을 집중하여 알고리즘, 프로그래밍, 문제해결에 대한 산출을 주체로 제안한다. 이들 과정을 통해 학습자 중심으로 진행하여 전체 프로세스에 따른 분해, 패턴인식, 추상화, 알고리즘의 내용을 포함하며 프로토타입 형태로 설계과정을 제시한다.

3.1 도구모델

SW교육의 개발과정에서 데이터 요구분석은 근본적인 알고리즘을 비교·평가할 수 있는 기초적인 프로그래밍 교육 실행도구적 기능을 한다. 도구모델은 문제를 해결하는 과정을 돕는다[12]. 도구 구성요소는 크게 요구분석(Needs), 디자인(Design), 구현(Implementation), 공유(Share)를 구성도구로 사용하여 기초적 프로그래밍과정과 처리과정까지를 포함하여 작동하고 프로그래밍 언어사고를 구현한다.

본 연구에서는 도구모형을 EZSTEAM Kit 교구를 도입기술을 기준으로 학습모형을 설계하고 있다. SW 교육 도구를 수업에 활용하는 과정에서 문제해결 능력에 학습중점을 둔다. 프로그래밍 언어사고를 적용하기 위해서는 Top-Down 진행방식의 방법론이 필수적이다. 또한 SW교육에서 스크래치, 엔트리, 파이썬 등의 범용 프로그래밍언어를 취업준비에 필요한 역량을 함양시킬 수 있는 연계과정을 수반되게 된다.

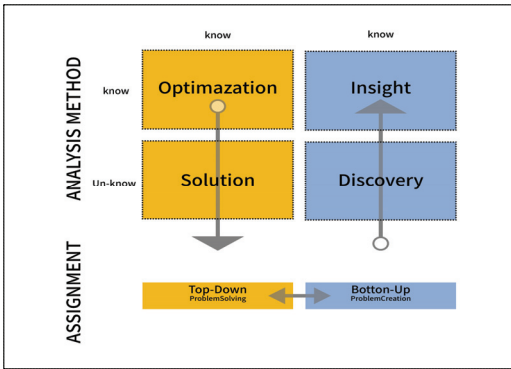


그림 1. EZSTEAM Kit Top-Down 알고리즘 방식
Fig. 1 Top-Down Algorithm Approach for the EZSTEAM Kit

그림 1과 같이 EZSTEAM Kit는 ‘도입-전개-정리’ 단계별 학습활동을 mBlock 프로그래밍 수업방식의 개선과제에 초점을 두고 교육과정 운영을 효과적으로 실효를 거두기 위함이다.

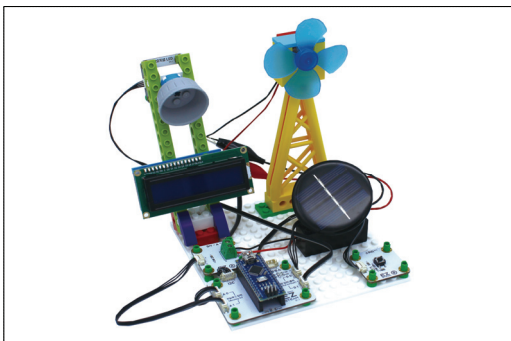


그림 2. EZSTEAM Kit 학습 구현 모델
Fig. 2 EZSTEAM Kit Learning Implementation Model

그림 2와 같이 EZSTEAM Kit는 시각적, 체험적, 상호작용적인 학습 경험을 제공하기 위한 하나의 구성품으로 되어있다. EZSTEAM 메인보드와 각 센서 타입으로 구성하고 있다. SW수업에서 하드웨어, 알고리즘, 프로그래밍 3가지 영역에서 발생하는 문제를 해결하기 위한 도구모듈(Tool Module)이다[13].

EZSTEAM Kit은 Top-Down의 상향식 교육모형을 제시한다. Top-Down모들의 반복패턴을 학생 주도적 활동으로 이끄는 단계(분석, 설계, 구현, 시험, 운용)로 역량을 키울수 있다. 또한 제작과정에서 알고리즘을 통한 프로그래밍 언어사고의 자주적 학습능력·창의적사고력·귀납적사고력 등을 함양하는 장점을 가지고 있다.

표 3. 코딩 학습 이해 난이도
Table 3. Coding learning comprehension difficulty

Level	Rang	Description
Beginner	0-30%	Basics of Python syntax: variables, conditions, loops.
Intermediate	31-70%	Functions, modules, object-oriented concepts, file I/O, exceptions.
Advanced	71-100%	Advanced syntax: OOP advanced topics, libraries, multithreading, parallel processing.

또한 스크래치는 블록 코딩지원과 파이썬의 프로그래밍 언어기반을 지원하여 초보자들도 쉽게 접근할 수 있는 PBL(Problem-Based Learning) 학습환경을 조성하여 문제해결력, 창의적사고력을 증진시킬수 있다[14].

3.2 과정모형

교육과정 개발에 있어 ICT(Information Communication Technology) 활용중심에서 알고리즘을 중심으로 SW교육과정을 설계하고자 한다.

중점사항으로 탐구, 프로젝트, 문제해결을 바탕으로 설계·모형 개발을 통해 소프트웨어를 개발하기 위한 기초 설계과정은 제한된 모듈이 아닌 학습자가 비판

적으로 사고하고 자기주도적으로 학습하는 과정을 통해 스스로 내용지식을 생성할 수 있도록 주도방식의 프로그램을 운영에 핵심을 두고 있다[15].

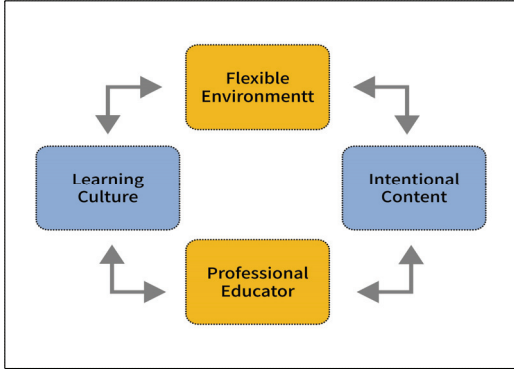


그림 3. ICT 교육과정 알고리즘 방식
Fig. 3 TICT Curriculum Algorithm Method

교육과정 설계는 그림 3에 나타난 바와 같이 학습 주제, 학습목표, 전체적인 수업진행, 학습활동에 대해 명확하게 제시하고 학습자의 고등사고능력, 의사소통·협업능력, 내재적 동기개발, 자기주도적 학습능력을 목표로 단계적 구성요소를 제시한다.

1. 분석 단계
 - 교육목적, 대상분석, 교육환경, 제한사항 분석
2. 설계 단계
 - 교육내용, 교육방법, 교육자료 분석
3. 개발 단계:
 - 교육자료 제작, 교육방법 구체화
4. 실행 단계
 - 교육 제공, 상호적 관리
5. 평가 단계
 - 교육효과 평가, 교육과정 개선

분석단계는 프로토타입 모델(Prototype Model) 통해 요구분석을 수행한다. 학습자료를 제공하여 요구사항 분석, 환경 디자인, 피드백 수집·수정과 같은 경험 과정을 분석한다.

설계단계는 선행학습과 교실수업을 운영하기 위한 수업 내용방법을 결정한다. 프로그램 목표와 기능을 정의하고 추상된 알고리즘 모듈을 프로그래밍 언어기반으로 구현하여 ‘교수-학습’ 활동을 중심으로 설계한다.

실행단계는 수업계획을 실행한다. 수업목표에 따라 교실수업의 형태를 다양하게 변형하고 단계별 수업 방법·자료를 개발한다. 또한 학생들의 선행학습 수준에 따라 토론식 수업, 팀 구성 기반학습 등 다양한 형태로 진행한다[16].

평가단계는 수업 실행과정 및 결과에 대한 평가를 바탕으로 수업 설계·모형에 대한 평가가 언급된다. 교사가 계획했던 수업 전략·수업 자료, 도구·매체의 적절한 활용여부, 학생의 흥미·집중도, 개별화 학습발생 여부 등을 교수학습 설계의 문제분석을 통한 개선점 도출한다.

학습설계는 의도적이고 명확한 계획을 수립하는 과정을 통해서 수업방식을 제시하는 것에 있다. in-class 학습에서 ‘무엇을 탐구하고 심화할 것인가?’에 대해 고민하고 학습내용을 선정한 후 학습매체와 액티브러닝(Active Learning)단계를 넘어 사고의 확장과 창의적 산출지원으로 방안이 모색되어야 한다[17].

IV. 지능형 소프트웨어 교육 시연 모델

본 연구에 목적에 따라 주어진 프로젝트 중심의 수업모델을 다음과 같이 설계하였다.

- 1) 주제
 - 풍력 발전을 만들기
- 2) 전략
 - 스토리텔링, 협동학습
- 3) 도구
 - EZSTEAM Kit
- 4) 웹코딩
 - 파이썬 기반의 블록 프로그래밍언어
- 5) 중점요소
 - 패턴인식, 추상화, 알고리즘, 자동화

4.1 학습목적

본 연구에서 제시한 수업모델에 대한 학습목적 프로그래밍 언어사고의 알고리즘 설계와 프로그래밍 사고를 함양하는데 그 목적에 있다.

첫째, 절차적 사고에 의한 문제의 이해와 구조화를 통해 정보를 표, 다이어그램 등 다양한 형태로 구조화하여 표현한다.

둘째, 간단한 알고리즘 체험을 통해 구상, 계획, 설계 단계를 기호를 이용하여 해결방안을 모색할 수 있다. 셋째, 프로그래밍 요소와 구조를 효율적으로 정보를 제공하고 처리하기 위해서 리스트, 계층, 테이블, 다이어그램 등과 같은 다양한 형식으로 정보를 표현한다.

넷째, 프로그래밍의 조건문과 반복문, 선택구조, 다중 반복구조, 제어구조 등을 이해하고 이를 이용하여 자료의 입출력 프로그램을 작성할 수 있다.

프로그래밍 언어사과는 창의적 사고력, 비판적 사고력, 문제해결력, 협동능력 등을 목표로 한다. 제시된 문제를 분석하고 정보를 처리하며, 알고리즘 설계를 통한 절차적 사고와 프로그래밍을 이해하며, 프로그램의 흐름을 이해하고 문법에 부담 없이 학습할 수 있는 방법으로 해결할 수 있다.

4.2 알고리즘 프로그래밍 시연

본 연구에서 시연모델은 프로그래밍 언어적 사고를 기반으로 설계되었으며, 직접교수법을 바탕으로 프로그래밍 언어와 문법을 설명하고 컴퓨팅 사고의 관점에서 접근을 통해 표 4과 같이 수업이 진행된다.

표 4. 알고리즘 프로그래밍 데모 모델
Table 4. Algorithm Programming Demonstration Model

Classification	Procedure	Explanation
Teaching Method	Demonstration-Centered Model	Demonstration, Modeling, Making.
Demonstration	Explanation, Demonstration	Teacher demonstrates strategies to students
Modeling	Imitation, Question and Answer	Students imitate teacher's demo, practice, ask, get answers.
Making Creation	Creation, Repetitive Activities	Students independently create projects based on demos.

또한 모듈을 시연할 때 문제의 단위를 분해하거나 알고리즘의 단계를 분해하여 그 패턴을 이해하고 핵심 개념이나 원리, 공식 등을 소개하며, 컴퓨팅 구성 내용도 직접 설계하고 구성해야 할 수 있는 방안을 제시한다.

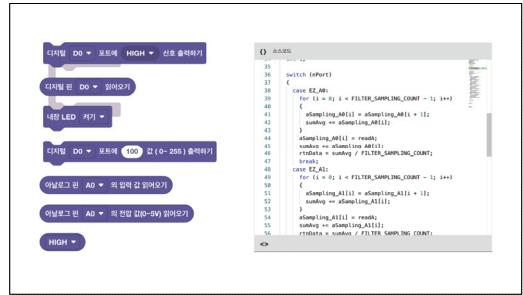


그림 4. EZSTEAM Kit 블랙 코딩 및 텍스트 코딩
Fig. 4 EZSTEAM Kit Black Coding and Text Coding

EZSTEAM Kit의 코드작성 방법은 비전공자를 코딩 교육에 최적의 환경을 그림 4와 같이 제공한다. 텍스트코딩의 파이썬과 자동화 검증을 위한 블록코딩으로 나누어 진행할 수 있다. 프로그래밍 구현하기 위해서는 순서도, 의사(pseudo)코드, 설계표현 등의 제어구조의 흐름을 파악하고 분석하여 그 결과를 언어로 표현하게 된다. Kit는 직접 언어표현을 시각적 블록코딩 기반의 전용 소프트웨어를 제공하여 프로그래밍을 어려워하는 학생에게 인지적 부담을 줄이고 학습 부담을 완화할 수 있는 학습환경을 제공한다.

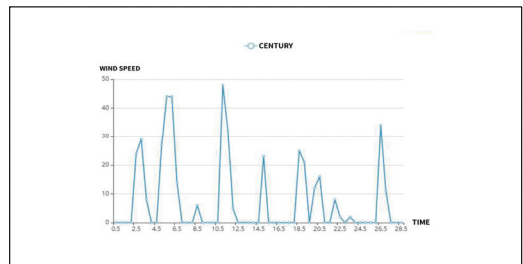


그림 5. 시간당 풍속 데이터 그래프
Fig. 5 Wind Speed Data Graph per Hour

EZSTEAM Kit은 블록조립을 통해서 센서를 연결하는 구조로 활용할 수 있다. 학습과정에서 코딩의 결

과가 센서를 통해 데이터분석이 실시간으로 시각화하여 그림 5와 같이 학습결과로 나타난다. 센서를 활용은 학생들의 수업참여를 유도하며, 코딩의 결과를 통해 학습자 스스로 문제해결 방안을 찾는 학습도구로 활용된다. 또한, 센서측정을 위한 코딩적용은 코딩에 대한 이해도를 높일 수 있으며, 기본적인 코딩에 대한 이해뿐만 아니라 데이터 분석의 중요성을 이해하고 개념이나 원리, 공식 등을 이해하는데 도움을 줄 수 있다.

V. 결 론

‘미래사회=4차산업혁명’으로 여겨질 만큼 미래의 모습을 그리는데 있어 4차산업혁명을 빼놓고 설명하기는 매우 어렵다. 현재 한국대학이 마주하고 있는 초저출산 현상과 재정확보의 문제는 시대가 원하는 인재를 길러내는데 큰 장애물로 작용한다[18]. 선진국의 주요 대학들은 지식생산과 기술이전을 통해 경제적 가치를 창출하고 있다. 스탠퍼드대학은 HP, 구글, 인스타그램 등 4만개가 넘는 기업을 창업해 성공적으로 실리콘밸리를 성장시켰다. 최근 기사에 따르면, 한국의 KAIS T도 기술이전을 통해 연간 수입을 증가시키고 있으며, 국내 대학들도 기술사업화에 힘쓰고 있다. 기술사회는 기술이 우리 일상생활과 업무에 깊숙이 관여하게 되면서 소프트웨어에 대한 이해와 능력이 다각적인 방안을 모색되고 있다.

본 연구에서 EZSTEAM Kit 교구기반의 프로그래밍 수업을 통해서 비전공자 학습의 특성을 파악하고 이를 위한 프로그래밍 언어적 사고를 기반으로 학습 모델을 제시하였다. 제안한 학습모델은 ICT(Information Communication Technology) 활용중심으로 알고리즘 기술의 언어적 사고에 대해 이해하고 다이어그램 기술의 분석단계를 프로토타입 모델(Prototype Model) 통해 비전공자 학습을 유기적으로 연계되도록 학습(Seamless Learning)을 구성하였다. 학습자가 컴퓨팅 사고력을 이용하여 문제 해결을 Top-Down 진행방식을 적용한 후 코딩에 대한 문법 이해의 어려움을 시스템 학습설계를 통해 해결하였다.

향후 미래사회의 교육 변화에 효과적으로 대응하기 위해서는 SW중심사회를 이끌어 갈 미래 인재 양

성하는 과정에서부터 이러한 미래 교육의 환경에 대처할 수 있도록 다양한 학습 자료를 제작하고 이에 맞는 학생들의 교육 니즈를 SW교육 교과과정에 반영하기 위한 수업설계가 현장교육의 기회를 제공할 수 있어야 한다. 또한 대학은 비판적 사고(Critical Thinking), 창조력(Creativity), 소통(Communication), 협력(Collaboration) 등의 역량을 지닌 인재를 길러내야 한다.

References

- [1] S. Choi, J. Kim, S. Ban, K. Lee, S. Lee and H. Choi, "Education Strategy to Foster Creative Talent for the 21st Century," *Report*, Dec, 2011.
- [2] O. Hong, K. Kim, and J. Lee, "Concept and Characteristics of Intelligent Science Lab," *Journal of the Korean Association for Science Education*, vol. 42, no. 02, Apr. 2022, pp. 177-184.
- [3] J. Park, M. Jang, M. Kwak and E. Chang, "Analysis of Research Trends in University Lifelong Education-Based on the Articles Published in Korean Journals from 1999 to 2019," *Korean Society for the Study of Vocational Education*, vol. 38, no. 06, Apr. 2019, pp. 89-114.
- [4] Y. Son and K. Lee, "Computational Thinking Teaching Model Design for Activating IT Convergence Education," *Journal of the KIECS*, vol. 11, no. 05, Apr. 2016, pp. 511-522.
- [5] J. Lee and E. Kang, "Development of Artificial Intelligence basic Liberal Arts Education Program for Non-majors," *Journal of Digital Contents Society*, vol. 22, no. 09, Apr. 2021, pp. 1431-1440.
- [6] H. Jho, "The Directions and Challenges of Science Education Based on the Prediction of Future Education and Schools," *Journal of Research in Curriculum & Instruction*, vol. 25, no. 01, Apr. 2021, pp. 61-78.
- [7] Y. Kim, J. Choi, D. Kwon and W. Lee, "Development of Algorithm Design

- Worksheets using Algorithmic Thinking-based Problem Model in Programming Education for Elementary School Students," *Journal of The Korean Association of Information Education*, vol. 25, no. 01, Apr. 2013, pp. 233-242.
- [8] E. Bae, "The Case Study on the Features of Youth Self-directed learning," *Studies on Korean Youth*, vol. 15, no. 02, Apr. 2004, pp. 247-279.
- [9] J. Nah, "TSoftware Education Needs Analysis in Liberal Arts," *Korean Journal of General Education*, vol. 11, no. 03, Apr. 2017, pp. 63-89.
- [10] W. Kim, "A Study on the Students' Perceptions Trend for Software Essentials Subject in University," *Korean Journal of General Education*, vol. 13, no. 04, Apr. 2019, pp. 161-180.
- [11] G. Park and Y. Choi, "Exploratory study on the direction of software education for the non-major undergraduate students," *Journal of Education & Culture*, vol. 24, no. 04, Apr. 2018, pp. 273-292.
- [12] T. Kim, D. Cho and S. Park, "Design and Implementation of a Concentration-based Review Support Tool for Real-time Online Class Participants," *Korea Institute of Electronic Communication Sciences*, vol. 18, no. 03, Apr. 2023, pp. 521-526.
- [13] Y. Lee and J. Cho, "Development and Application of STEAM Education Program based on Topic-specific Project," *Journal of the Korea Academia-Industrial cooperation Society*, vol. 13, no. 12, Apr. 2012, pp. 5770-5775.
- [14] Y. Son and K. Lee, "Computational Thinking Teaching Model Design for Activating IT Convergence Education," *Korea Institute of Electronic Communication Sciences*, vol. 11, no. 05, Apr. 2016, pp. 511-522.
- [15] K. Kim, H. Lee and H. Song, "What Competencies are Required for Teachers to Effectively Implement Flipped Learning Classes?," *Journal of Korean Teacher Education*, vol. 33, no. 02, Apr. 2016, pp. 181-207.
- [16] S. Lee and J. Seong, "Deriving AI-Based E-Learning and Personalized Education Methods to Improve Efficient Class Satisfaction in the Post-Covid-19 Environment Using Statistical Techniques," *Korea Institute of Electronic Communication Sciences*, vol. 17, no. 06, Apr. 2023, pp. 1213-1220.
- [17] K. Chang, "A Model of Action Learning Program Design in Higher Education," *Educational Technology International*, vol. 27, no. 03, Apr. 2011, pp. 475-505.
- [18] I. Hwang, "The Influence of IS Technology and Communication Uncertainty on IS Voice Behavior: The Role of Susceptibility to Informational Influence of Employee," *Korea Institute of Electronic Communication Sciences*, vol. 18, no. 01, Apr. 2023, pp. 165-176.

저자 소개



장승영(Seung-Young Jang)

2002년 호남대학교 토목공학과 졸업(공학사)

2009년 남부대학교 대학원 디지털경영정보학과 졸업(공학석사)

2013년 남부대학교 대학원 디지털경영정보학과 졸업(공학박사)

※ 관심분야 : 콘텐츠, 코딩, 마케팅, 디자인

