

컴퓨팅 사고기반 융합 수업모델 개발

유정수^{1*}, 장용우²

¹전주교육대학교 컴퓨터교육과 교수, ²전주교육대학교 체육교육과 교수

Developing a Learning Model based on Computational Thinking

Jeong-Su Yu^{1*}, Yong-Woo Jang²

¹Professor, Dept. of Computer Education, Jeonju National University of Education

²Professor, Dept. of Physical Education, Jeonju National University of Education

요약 디지털 사회를 위한 AI와 빅데이터 시대의 컴퓨팅 사고는 컴퓨터가 실행할 수 있는 방식으로 우리가 해결하고자 하는 문제와 솔루션을 표현하는 일련의 문제해결 방법을 의미한다. 컴퓨팅 사고는 컴퓨터 과학의 기본 개념을 도출함으로써 문제를 해결하고 시스템을 설계하고 인간의 행동을 이해하는 것으로, 학생들에게는 어려운 문제와 애매한 퍼즐을 맞추는 접근법이다. 본 논문에서 우리는 컴퓨팅 사고를 댄스 동작과 융합하여 학생들이 문제를 해결할 수 있는 수업 모델을 개발하였다. 개발된 수업 모델을 가지고 ○○대학교 1학년 예비교원 93명을 대상으로 1학기 동안 수업한 결과, 수업 참가자들은 비디오 수준의 만족스러운 알고리즘을 만들어 냈다. 또한, 제안된 모델이 수업 참여 학생들의 컴퓨팅 사고 이해에 크게 기여함을 알 수 있었다.

키워드 : 컴퓨팅 사고, 컴퓨팅 사고기반 융합, 댄스 동작, 인공지능, 컴퓨팅 사고자, 컴퓨팅 창조자

Abstract Computational thinking in the AI and Big Data era for digital society means a series of problem-solving methods that involve expressing problems and their solutions in ways that computers can execute. Computational thinking is an approach to solving problems, designing systems, and understanding human behavior by deriving basic concepts in computer science, and solving difficult problems and elusive puzzles for students. We recently studied 93 pre-service teachers who are currently a freshman at ○○ university. The results of the first semester class, the participants created a satisfactory algorithm of the video level. Also, the proposed model was found to contribute greatly to the understanding of the computational thinking of the students participating in the class.

Key Words : Computational Thinking, Convergence based on Computational Thinking, Dance Moves, Artificial Intelligence, Computational Thinker, Computational Creator

1. 서론

21세기에 디지털 기술을 이해하는 것은 사회적, 경제적, 정치적 측면에서 매우 중요하다. 알다시피 오늘날 경제와 사회는 디지털 매트릭스[1] 안에서 진화하고 있

다. 따라서 혁신적인 기술에 의해 움직이는 세계에서 살아가기 위해서는 전 세계의 모든 사람들이 일하고, 배우고, 살아가고, 참여할 수 있는 일련의 디지털 기술을 알아야 한다.

이러한 디지털 기술의 기준은 끊임없이 성장하는 지

This paper was supported by the research fund of the Jeonju National University of Education National University fostering project in 2021.

*Corresponding Author : Jeong-Su Yu(jsyu@jnue.kr)

Received December 21, 2021

Accepted February 20, 2022

Revised January 28, 2022

Published February 28, 2022

식 경제가 점점 더 인간의 인지 능력과 컴퓨팅 능력의 결합을 요구함에 따라 업무 환경은 근본적으로 변화하고 있다. 컴퓨팅은 다양한 분야에서 활용되고 있으나 이러한 기술 중심 인력에 들어갈 준비가 된 학생이 부족하다[2]. 이러한 문제를 인식하여 컴퓨팅 사고(Computational Thinking, 이하 CT라 함)는 빠르게 변화하는 직업 세계에 대해 학생들을 준비시키는 기본 구성 요소로 확인되었다[3].

최근에 와서는 특히 인공지능(이하 AI라고 함)과 같은 디지털 기술이 국가의 경제 발전 및 사회 발전에 원동력일 뿐만 아니라 우리 사회 모습과 관행 등의 형태를 바꾸고 있고, 앞으로도 계속 우리들에게 영향을 미칠 것이기 때문에 이 기술에 대해 모두가 배워야 할 필요성이 있다[4]. 컴퓨팅을 이용한 기계 지능이 지능을 실현하는 주요 수단이기 때문에 컴퓨팅 사고는 서로 엮여 증척되는 것이 분명하다. 추상화의 힘, 문제해결 기술로서의 휴리스틱의 사용, 통계 학습 및 데이터 과학의 중요성과 같은 컴퓨팅 사고의 많은 구성 요소가 AI 기술을 다루는데 포함된다.

컴퓨팅적이고 창의적인 사고 기술은 컴퓨터 과학과 예술 분야를 넘어 다양한 맥락에서 이해하고 문제해결을 위한 가치 있고, 세계 혁신 경제에서 학생들에게 생각 스킬 역할을 하고 있다.

따라서 본 연구에서는 댄스 프로그래밍을 통해 컴퓨팅 사고를 이해하고 학습자들에게 창의적이며 고차원적인 사고를 길러주는 방법을 탐구한다. 이를 위해 댄스는 전형적으로 컴퓨터에 관심이 없는 학생들에게 호기심과 동기부여를 촉진하고, CT를 가르치는데 트리거 역할을 한다. 또한, 댄스 동작과 프로그래밍을 융합하여 예비교원들에게 CT를 이해시키는데 도움을 주는 새로운 구체화 과정도 만들었다.

2. 인간+AI 시대 컴퓨팅 사고(CT)

2.1 AI시대 CT의 중요성 및 21세기에 필요한 역량 함양을 위한 교육과정 프레임워크

요즘 같은 AI시대에는 일자리가 많은 분야를 전공하는 것이 중요하다. 예를 들어, 많은 사람들이 익힐 수 없는 기술이 컴퓨터 프로그래밍이다. 프로그래머들은 높은 평가를 받기 때문에 고소득 직업을 가질 수 있는 기회가 많다. 따라서 전 세계 많은 국가들이 초·중등 교

육과정에서 컴퓨터 과학(Computer Science)의 일부 내용을 다루고는 있으나 아직까지는 단순히 코딩 정도에 머무르고 있다. 실제로 프로그래밍은 AI와 함께 일하는 미래 사회에는 매우 유용한 기술이고 다양한 직업과 삶의 길에 있는 학생들에게 유익할 수 있는 기술이다. 그러나 프로그래밍은 컴퓨터 과학의 일부분에 불과하다. 프로그래밍은 국어에서의 읽고, 쓰거나 수학에서의 셈하기보다 더 진보된 도구이다.

2006년 제넷 윈은 컴퓨팅 기기의 장점을 알고 컴퓨터 과학자처럼 사고하는 것은 읽고(Reading), 쓰고(wRiting), 셈하기(aRithmetic) 3R 이외에도 21C를 살아가는 모든 사람들이 갖추어야 할 기본 리터러시(literacy)라고 주장하였다[5-7]. 인쇄기가 3R의 확산을 촉진시켰듯이 컴퓨팅과 컴퓨터가 CT의 확산을 촉진시킬 것이다.

오늘날 MZ세대와 α(알파) 세대 학생들은 컴퓨팅에 의해 강력하게 형성된 인력이다. 변화하는 사회패러다임에서 성공하려면 학생들은 다양한 추상화 수준으로 문제를 해결하기 위해 알고리즘적으로, 컴퓨팅적으로 사고하는 법을 배워야 한다. 이러한 CT 기술은 사회적 기능에 통합되어 AI 시대에 기본적인 리터러시가 되고 있다.

인터넷은 전 세계를 연결, 다양한 국제 네트워크를 이용하여 세계 경제를 증가시키면서 세계화를 이루어 냈다. 이로 인해 전 세계는 더 경쟁적이고 도전적으로 변했다. 이렇게 변화되는 사회에서 사람들이 성공적으로 살아가고 잘 작동하는 사회를 만들어내기 위해 개인의 필요한 역량을 길러주기 위한 교육과정 프레임워크를 Fig. 1과 같이 제안하였다[8].

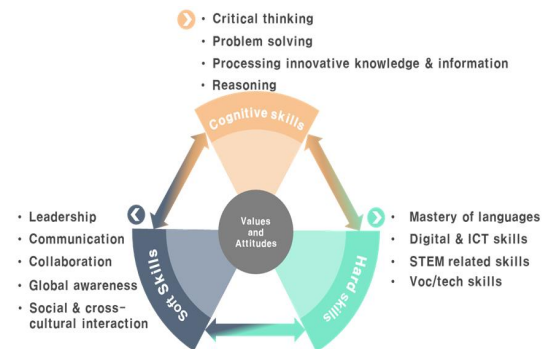


Fig. 1. 21C Curriculum Framework by Dr. Habibah

[9]에서는 21세기에 필요한 스킬을 크게 인지 스킬(recognition skills), 소프트 스킬(soft skills), 하드 스킬(hard skills)로 구분하였다. 인지 스킬은 학생들의 사고 능력, 판단 능력, 표현력을 개발시켜주는 내용을 말한다. 소프트 스킬은 배우고자 하는 의지와 인간성 함양에 대한 교육내용이 소프트 스킬이다. 그리고 지식과 기술이 습득되었는지 확인하는 교육과정이 하드 스킬이다. Fig. 1처럼 많은 사람들은 CT를 핵심 교육과정에 통합하는 것이 모든 학생들에게 CT에 대한 접근을 제공하는 가장 확실한 방법이라고 주장한다. 학생들에게 CT는 기술 이상의 의미를 가지며, 문제를 통해 생각하고 해결책으로 이어질 수 있는 단계를 처리하여 컴퓨팅의 용량과 한계를 개발하는 방법이다. 해리스은 초중등 교육과정에서는 CT를 내용 영역 전반에 걸쳐 통합함으로써 학생들이 이러한 중요한 기술을 습득할 수 있도록 하는 것이 자연스럽다고 주장하고 있다(Fig. 2 참조)[8].

CT는 사고, 처리 및 문제해결 방법으로 모든 분야에 영향을 미치기 때문에 CT를 기반으로 모든 학생들은 융합할 수 있는 능력을 길러야 한다고 교육부는 주장하고 있다[9-11].

초중등 교육 환경에서 학생과 교사에게 CT는 기술이나 컴퓨터 과학 스킬을 사용하는 것 그 이상이다. CT의 본질은 문제를 해결하기 위해 일련의 단계, 즉 알고리즘을 사용하여 복잡한 문제를 보다 친숙하고 관리 가능한 하위 문제로 분해하고(문제 분해), 해결책은 유사한 문제로 이전되고(추상화), 마지막으로 컴퓨터가 이

러한 문제를 보다 효율적으로 해결하는 데 도움이 될 수 있는지 결정(자동화)” 하는 것이다.

기술의 지속적인 발전과 함께 교사는 학생들이 디지털 시민이 되도록 권한을 부여하고 학습에 대한 소유권을 갖도록 지도해야 한다. 따라서 컴퓨터 사고력과 같은 컴퓨팅 아이디어를 조기에 학생들에게 소개함으로써 우리는 아이들이 단순한 기술 소비자 이상의 존재가 되도록 도와야 한다.

APEC InMside I 프로젝트 보고서에 따르면 새로운 교육과정 프레임워크에 대한 주요 논점은 단순 코딩을 넘어 인간성과 코딩에 초점을 맞추도록 권고하고 있다 [2]. AI 시대에 가장 관련이 있는 삶과 일에 대한 가치와 기술을 갖춘 차세대 인재를 길러내는 것이 목적이기 때문에 코딩에 대한 숙련된 기술을 갖추기 위해서는 사람을 통해서도 유연하고 인지적인 기술이 필요하다.

CT는 인식론, 윤리, 지혜와 밀접한 관련이 있다. 이들 모두는 지식, 학습, 가치와 같은 몇 가지 핵심 개념을 공유하고 있다. 예를 들어, 컴퓨팅 사고는 도구를 제공한다. 지능적이고 도덕적인 인공 에이전트를 설계하고 우리에게 감정적이고 지능적인 지원을 한다. 인공 에이전트는 완전히 자율적일 수 있는 에이전트이지만 우리는 그들을 신뢰하는 법을 배워야 한다.

앞으로 다가올 미래는 인간과 AI가 함께 협업하며 일을 해야하기 때문에 인본주의 중심 활동의 가치를 찾아 살아갈 수 있는 역량과 AI와 로봇을 우리 삶의 일부로 인식하고 더욱 더 인간다움을 추구하는 새로운 진보적, 인류적 역할을 만들어 내야 한다.

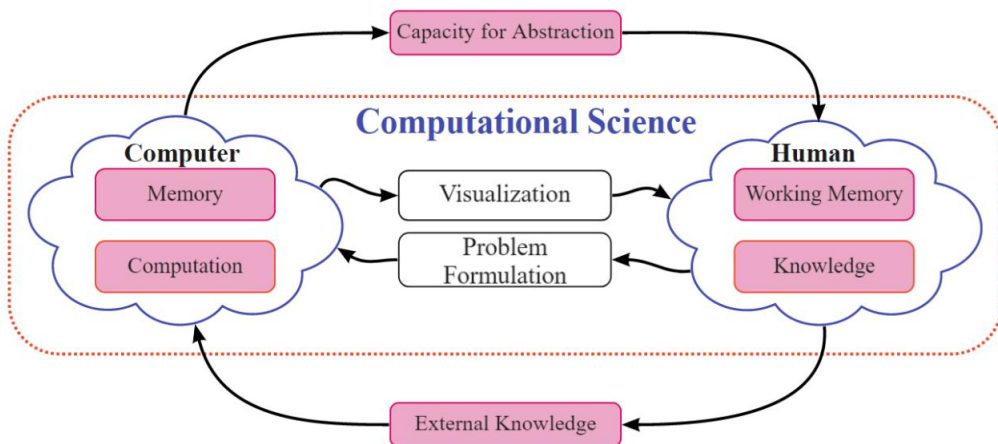


Fig. 2. Connecting computational thinking to life

2.2 댄스와 컴퓨팅 사고(CT)의 융합

사고(Thinking)란 “체화된” 활동이라는 개념, 즉 뇌 뿐만 아니라 활동적인 사람 몸 전체가 상황을 개념화하는 방식에 관여한다는 개념은 철학자, 인지 및 학습 과학자들에 의해 발전해왔다[12]. 그러나 체화된 인지 문헌은 아직 창의적, 문화적, 표현적 매체로 명시적으로 다루고 있지 않다. 신체와 신체의 관념이 대화적이고 표현적인 사상의 부재는 남고, 창조적 표현을 통한 댄스 활동을 통한 융합 학습에 있어서는 특별한 차이가 존재한다[12-14].

본 논문에서는 마린[15] 연구와 같이 신체의 조정된 움직임의 표현성을 구현의 대안적인 프레임으로 제공한다. 이러한 구현의 새로운 프레임은 댄스를 CT 탐구에 참여하기 위한 매개체로 본다. 댄스는 문제 해결을 위한 귀중한 자원으로 사용하며, 융합을 이해하고 창의적이고 구체화된 것으로 상상하는 기회로 확장한다.

3. CT기반 융합 수업 모델

3.1 CT기반 융합 수업 설계

본 연구는 ○○대학교 1학년 학생들이 수강하는 교양필수 과목에서 93명을 대상으로 2021년 1학기 수업 시간(15시간/1학점) 컴퓨팅 개념 학습, 컴퓨팅 사고 개념 이해 및 실습, 컴퓨팅 사고 프로젝트 실습 및 컴퓨팅 관점 개발을 위한 다양한 교수법을 통해 학습하였다. 주요 수업 내용은 Table 1과 같다.

Table 1. Lesson activities of CT-based convergence

Week	Main Content
1	What is the digital age, the AI age?
2	Human value in the age of AI
3	What is the SW and AI education?
4	What is CT and why is it important?
5	CT(decomposition, pattern recognition, abstraction)
6	CT(algorithm)
7	Training to cultivate CT(active learning)
8	Training to cultivate CT(active learning)
9	Training to cultivate CT(digital storytelling)
10	Training to cultivate CT(PBL) Programming with Scratch

본 수업은 유튜브 동영상상을 통해 댄스 동작이 포함된 동요와 가요 동영상상을 분석하고 스크래치를 사용하여 프로그램을 작성하였다. 본 수업 설계는 예비교원들

에게 컴퓨팅 사고에 대한 개념을 이해를 통해 컴퓨팅 사고를 향상시키고 프로그래밍에 참여하도록 장려하는 활동을 개발하는 것을 목표로 한다.

거의 모든 교육자들은 CT를 개발하기 위한 맥락으로서 프로그래밍 활동에 눈을 돌리지만, CT는 다른 분야의 문제 해결과 많은 공통점을 공유한다. 융합은 초등 교사들이 CT와 다른 과목에서 중요한 기술과 관행 사이의 상당한 중복을 이용할 수 있게 해준다. CT를 수학과 과학 통합하는 접근법을 취하면 핵심 과목 교육 과정에서 이를 수행할 수 있다[9-11]. 학생들은 CT 개념 이해를 심화시키면서 CT 연습 및 타 교과 내용과 융합 개발을 촉진한다.

3.2 실험 설계 및 방법

Table 2에서 보듯이 수업 참여자들은 유튜브에서 자신이 원하는 동영상상을 찾아서 동영상상의 춤을 CT 구성요소를 기반으로 분석한 후 스크래치 프로그래밍 언어를 사용하여 멀티미디어 프로젝트를 설계 및 구현하였다. 이 과정을 통해 CT가 댄스와 어떻게 교차하는지, 그리고 댄스와 창의적인 코딩을 혼합하였다. 컴퓨팅 사고의 4가지 구성요소(분해, 패턴 인식, 추상화, 알고리즘)는 댄스를 분석하는데 세심하게 내재되었다. 이를 통해 예비교원들에게 댄스의 순서화된 단계부터 컴퓨터 알고리즘 작성을 통해 코딩과 댄스 동작의 관계를 알 수 있다.

Table 2. Experimental design stages

Learning Unit	Content
1	<ul style="list-style-type: none"> Choose the dance video you want in youtube Dance analysis without CT concepts Dance analysis in group or pair(active learning, make digital story, pbl)(1 week/1hr)
2	<ul style="list-style-type: none"> Understand four cornerstones of CT with dance moves(4weeks/4hrs) <ul style="list-style-type: none"> - decomposition - pattern recognition(1week/1hr) - Abstraction(2 weeks/2hrs) - algorithm(1 week/1hr)
3	<ul style="list-style-type: none"> Programming with Scratch(1 week/1hr)

이는 사람들이 일련의 단계를 따라가서 순서대로 반복하는 것이 어떤 기분인지 자신의 몸으로 경험할 수 있게 해주었다. 이러한 유형의 신체적 기술은 컴퓨터 프로그램을 코딩할 때 필요한 기술과 유사하다. 예를 들어, 신체에 댄스 패턴을 어떻게 수행해야 하는지를

알려주는 것은 알고리즘을 쓰는 것과 비슷하다.

수업 참여자들은 신체 댄스 연습을 코딩 세션과 번갈아 가며 신체 표현력과 CT 사이를 오갔다. 코딩 세션 동안 각 참여자들은 개인 노트북으로 개별적으로 작업하였다. 이러한 활동의 주요 목표는 최근 획득한 것을 적용할 수 있는 멀티미디어 스크래치 프로젝트를 만드는 것이었다.

본 연구에서는 프로그래밍 경험이 없는 예비교원 1학년 학생들을 대상으로 3단계 학습 유닛 효과를 테스트했다. 우리는 다음과 같은 결과를 예상하고 있다.

- 가설 1 수업 참가자들은 1단계 학습 유닛에서도 비디오 수준의 만족스러운 알고리즘을 만들어 낼 것이다.
- 가설 2 2, 3단계 학습 유닛을 통해 수업 참여 학생들의 컴퓨팅 사고 과정 이해에 크게 기여할 것이다.

본 연구는 다음과 같이 3단계 학습 유닛을 설계하였다.

• 1단계 학습 유닛

수업에 참여한 학습자들에게 Fig. 3과 같은 춤 동작이 포함된 동영상을 보여준다. 예시 이외의 동영상도 학습자가 자율적으로 선택 가능하도록 하였다. 본 학습 유닛에서 학생들은 강의 및 작 활동을 통해 아이디어를 교환하고 지식을 공유하며 강의와 관련된 문제를 짝과 협력하는 액티브 러닝, 디지털 스토리텔링 및 PBL을 통해 이를 수행하였다.



Fig. 3. Examples of kids' songs and pop music videos with dance moves

• 2단계 학습 유닛

본 학습 유닛에서는 선택한 댄스를 컴퓨팅 사고의 구성요소들(분해, 패턴 인식, 추상화, 알고리즘)로 분석하도록 하였다.

① 분해(decomposition)

복잡한 문제에서는 분해하지 않으면 문제를 해결하는데 어려움을 겪을 수 있기 때문에 분해가 중요하다. 여러 가지 단계를 한꺼번에 처리하는 것은 문제를 여러

개의 작은 문제로 세분화하여 한 번에 하나씩 해결하는 것보다 훨씬 더 어렵다. 따라서 문제를 더 작은 부분으로 세분화한다는 것은 각각의 작은 문제를 더 자세히 조사할 수 있다는 것을 의미한다. 마찬가지로, 복잡한 시스템의 작동 방식을 이해하려고 하면 분해를 사용하는 것이 더 용이하다. Fig. 3의 BTS 동영상에서 BTS 멤버들의 댄스를 한꺼번에 분석하기보다는 Fig. 4처럼 각 멤버들 각각의 댄스를 분석하는 것이 춤 동작을 표현하기 쉽다.

No	Face	Upper body	Arms	Legs
1	front	front	bend your elbows and place your hands on your bed;	flex your knees 14 times and then stretch them
2	to the right	front	spread out the right hand and raise it up to the right, the one hand bends the elbow and puts it on the waist.	flex your knees 2 times and then stretch them
3	turn from right to front	front	stretch right hand to front on top right	flex your knees 2 times and then stretch them
4	front	front	shake up and down 5 with right hand extended from top to bottom	flex your knees 2 times and then stretch them
5	turn from front to right	front	Put your right hand up the right	flex your knees 2 times and then stretch them
6	to the right	front	spread out the right hand and raise it up to the right, the one hand bends the elbow and puts it on the waist.	flex your knees 2 times and then stretch them
7	turn from right to front	front	stretch right hand to front on top right	flex your knees 2 times and then stretch them
...				

Fig. 4. Expression of dance movs decomposition

② 패턴 인식(pattern recognition)

패턴이란 일부 문제가 공유하는 유사성 또는 특성을 의미한다. 패턴 인식은 컴퓨터 과학의 4대 초석 중 하나로 우리가 더 복잡한 문제를 더 효율적으로 해결하도록 도울 수 있는 작고 분해된 문제 사이의 유사성 또는 패턴을 찾는 것을 말한다. Fig. 4에서 빨간색 박스 부분이 춤 동작 중 찾아진 패턴의 일부이다.

③ 추상화

추상화는 컴퓨터 과학의 4가지 초석 중 하나로 추상화가 중요한 이유는 우리가 해결할 문제에 대한 이해와 어떻게 문제를 해결할 것인가에 대한 일반적인 생각을 만들 수 있게 해준다. 이 프로세스에서 문제를 해결하는데 도움이 되지 않는 구체적인 세부사항과 패턴은 모두 삭제한다. 이는 우리가 문제에 대한 우리의 생각을 형성하는데 도움이 된다. Fig. 5와 같이 춤 동작을 좌우 이동과 회전 유무 및 템포(댄스속도) 블록으로 추상화하였으며, Fig. 6은 실제 스크래치 블록으로 추상화하여 구현한 예이다.

Movement	Rotation	Tempo (Dance Speed)	Movement	Rotation	Tempo (Dance Speed)
move left	×	moderate	stop	×	slightly slow
RM does not dance			stop		
move center	×	moderate	forward	×	very very fast
move left	×	moderate	stop		
move right	×	fast	stop	×	×
move forward	×	fast	move left		very fast
move left and right	○	fast	stop	○	×
move left	×	very fast	move forward		moderate
move left and right	×	fast	move right	×	fast
move forward	×	very fast	move right	○	fast

Fig. 5. Abstraction process of Dance moves

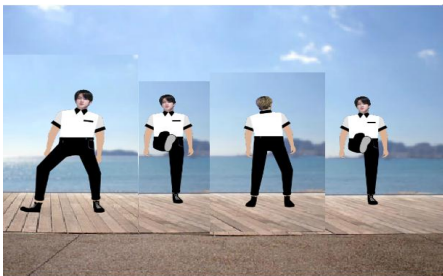


Fig. 6. Examples of Abstraction expression on Scratch stage

④ 알고리즘(Algorithm)

CT와 밀접하게 관련된 개념은 알고리즘적 사고이다. CT는 컴퓨터를 사용하여 데이터를 요약하거나 메시지를 보내거나 보다 쉽게 이해할 수 있는 형태로 변환함으로써 데이터에 대해 생각하는 것이다. 도구를 구축하기 위해서는 알고리즘적 사고가 필요한 반면, 도구 사용에는 컴퓨팅 사고가 필요하다.

알고리즘적 사고는 특정 목적을 달성하는 방법에 대해 생각하는 것이다. 즉, 방법에 대한 세부지향적 사고이다. 컴퓨터 프로그래머에게 알고리즘적 사고는 데이터 구조와 프로그램의 설계에 들어가는 것이다[16]. 데닝(Denning)은 알고리즘이 CT의 핵심이라고 강조한다[3]. 알고리즘적 사고는 식별할 수 있는 패턴이나 순서를 가진 것을 단계별로 세분화하는 것으로, 구글 검색 등과 같이 알고리즘은 일상적인 기술에 깊이 녹아 있다. 알고리즘이 종종 알려진 시작과 끝을 가지고 있다는 것을 고려하면, 창의성은 문제가 아닌 것처럼 보일 수 있다. 그러나 알고리즘의 단계에는 반드시 정해진 형식이 있는 것은 아니다. 이러한 단계의 순서 결정, 즉 알고리즘 자체의 구축은 프로세스의 일부이다(Fig. 7 참조). 알고리즘적 사고는 창의력을 필요로 한다.



Fig. 7. Expression of dance moves algorithm

• 3단계 학습 유닛

이 단계에서는 프로그래밍을 통해 알고리즘 결과를 구현하도록 한다. 2단계 학습 유닛에서는 프로그래밍 기술을 가르치는 것보다 CT의 발달을 중심으로, 즉 학생들이 다양한 분야와 맥락에서 문제를 해결하는데 사용될 수 있는 컴퓨터 과학 개념과 전략을 배울 수 있도록 도와준다.

본 학습 유닛을 통해 우리는 프로그래밍 같은 잠재력을 보았다. 대부분 학생들은 전문 프로그래머나 컴퓨터 과학자로서 직업을 추구하지 않을 것이지만 프로그래밍을 유창하게 할 수 있게 되는 것은 모두에게 가치가 있고, 특히 예비교원들에게는 매우 중요하다. 3단계 학습 유닛을 통해 예비교원들을 코드로 자신만의 이야기를 만들면서 사물을 창조하고 새로운 기술로 자신을 표현하는 능력에 대한 자신감을 키우기 시작했다(Fig. 8 참조).

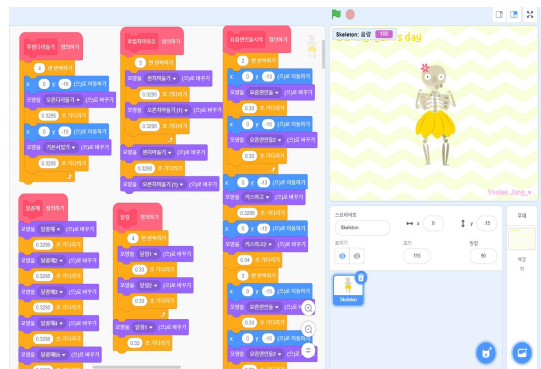


Fig. 8. Dance moves on Scratch Stage

4. 결론 및 제언

본 연구는 개발 중인 융합 수업 모델을 가지고 예비교원들에게 댄스 동작을 통해 컴퓨팅 사고 과정을 쉽게 이해시키는 수업을 진행하였다. 수업은 10시간 중 6시간에 걸쳐 93명을 대상으로 진행되었다. 실제로 개발 중인 모델이 CT 역량을 기르는데 효과가 있는지를 알아보기 위해 체육의 댄스를 선택하였다. 댄스와 CT를 결

합한 이유는 댄스는 전문지식이 없이도 손쉽게 접할 수 있기 때문에 선택하였다. 댄스는 전형적으로 컴퓨터에 관심이 없는 학생들에게 호기심과 동기부여를 촉진하고, 흥미를 유발시켜 CT를 기르는데 트리거 역할을 할 수 있었다. 또한, 액티브 러닝(active learning), 디지털스토리텔링 및 PBL 융합 수업 모델을 통해 CT 개념을 이해시키면서 창의적이며 고차원적인 사고를 길러주고자 하였다.

프로그래밍 경험이 전혀없는 1학년 예비교원들은 3단계 학습 유닛을 통해 댄스 선택부터 댄스 알고리즘 개발 및 각자의 생각을 디지털로 표현하는 프로그래밍까지 진행하였다. 수업 참가자들은 1단계 학습 유닛에서도 비디오 수준의 만족스러운 알고리즘을 만들어 냈다. 또한, 2, 3단계를 거치면서 수업 참여 학생들이 직접 프로그램을 개발해봄으로써 CT에 대한 이해에 크게 도움이 됨을 알 수 있었다. CT를 댄스 동작과 융합함으로써 예비교원들이 CT 개념을 탐구할 수 있는 학습 시간을 찾을 가능성이 더 높음을 알 수 있었다.

우리나라 SW교육 및 AI교육 연구자들 중 일부가 프로그래밍의 가치를 경시하고 있으나 본 연구 결과 프로그래밍은 학생들이 컴퓨팅 개념, 관행 및 관점에 대해 이해의 폭을 넓히는 효과적인 방법이 될 수 있다는 것을 수업 중 발견했다. 학생들이 자신의 프로젝트를 코드화할 때 의미 있는 맥락에서 개념과 문제해결 전략을 접하게 된다.

본 논문에 사용된 융합수업 모델은 10시간 중 6시간에 걸쳐 93명을 대상으로 진행되었기 때문에 개발 중인 모델 효과에 대해 일반화하기는 어렵다. 또한, 초등학교에서 널리 적용될 가능성은 높지 않을 것이다. 그 이유는 시수와 수업 내용이 교육과정 범위를 벗어나 있으며, 현재 교사의 CT에 대한 이해와 융합 역량이 부족하기 때문이다. 개발 중인 모델에 대해 앞으로 연구를 통해 더 많은 예비교원과 초등학교 학생들을 대상으로 통계적인 유의성 검증을 할 예정이다.

대부분 학생들이 문화가로 자라지는 않겠지만, 모든 학생들이 글쓰는 법을 배워야 한다는 강한 공감대가 형성돼 있다. 글쓰기를 통해 학생들은 아이디어를 조직하고 표현하고 공유할 수 있는 능력을 개발하고 자신을 다르게 보기 시작하기 때문이다. 이와 같이 디지털 시대에 프로그래밍은 디지털 글쓰기인 것이다. 따라서 AI 시대에 살아가기 위해서는 컴퓨터 과학을 초등 교육과

정에 필수로 도입하여 컴퓨터 과학에 기초가 되는 CT 스킬을 가르쳐야 할 것이다. CT는 확실히 가치 있는 목표지만, 초중등학교 현장에서는 문법을 가르치거나 학생들에게 하나의 정답을 가진 문제를 제시하는데 너무 좁게 초점을 맞추고 있다. 본 연구에서는 프로그래밍이 학생들에게 자기만의 프로젝트를 만들고 자신의 아이디어를 표현할 기회가 있을 때 얼마나 동기부여가 되고 의미 있게 되는지를 인지시키는데 중점을 두었다. 이러한 경험을 통해 학생들이 컴퓨팅 사고자(computational thinker)뿐만 아니라 컴퓨팅 창조자(computational creator)로 발전하게 될 것이다.

REFERENCES

- [1] N. V. Venkatraman. (2017). *The Digital Matrix : New Rules for Business Transformation Through Technology*, LifeTree.
- [2] APEC Human Resources Development Working Group. (2022). *Developing Computational Thinking on AI and Big Data Era for Digital Society - Recommendations from APEC InMside I Project*, APEC.
- [3] P. J. Denning. (2009). The profession of IT Beyond computational thinking. *Communications of the ACM*, 52(6), 28-30.
- [4] P. J. Denning. (2017). Remaining trouble spots with computational thinking. *Communications of the ACM*, 60(6), 33-39.
- [5] J. M. Wing. (2016). "Computational thinking," in *Communications of the AMC*, 29(3), 33-35.
- [6] J. M. Wing. (2006). Computational thinking. *Communications Of The ACM*, 49(3), 33-35.
- [7] J. M. Wing. (2014). Computational thinking benefits society. *Social Issues In Computing 40th Anniversary Blog*.
- [8] D. Barr, J. Harrison & L. Conery. (2011). Computational thinking: A digital age skill for everyone. *Learning & Leading With Technology*, 38(6), 20-23.
- [9] Ministry of Education. (2020). Science Education Comprehensive Plan(draft) in 2020~2024.
- [10] Ministry of Education. (2020). Mathematics Education Comprehensive Plan(draft) in 2020~2024.
- [11] Ministry of Education. (2020). Information Education Comprehensive Plan(draft) in

2020~2024.

- [12] R. M. Montoya, R. S. Horton, J. L. Vevea, M. Citkowicz & E. A. Laube. (2017). A re-examination of the mere exposure effect: The influence of repeated exposure on recognition, familiarity, and liking. *Psychological Bulletin*, 143(5), 459-498.
- [13] S. B. Daily, A. E. Leonard, S. Jorg, S. Babu & K. Gundersen. (2014). Dancing alice: Exploring embodied pedagogical strategies for learning computational thinking, *Proceedings of the 45th ACM technical symposium on Computer science education*, 91-96.
- [14] S. F. Vogelstein, L. Brady, C. Steinberg, R. Thomas, C. Champion, D. Lindberg, L. Enyedy, N. DesPortes, K. Payne, W. Bergner, Y. Taylor, E. & R. B. Shapiro. (2021). Embodying STEM: Learning at the intersection of Dance and STEM. *International Society of the Learning Sciences*.
- [15] M. A. Taylor, K. H. Shapiro, B. R. & Hall, R. (2020). Why Learning on the Move: Intersecting Research Pathways for Mobility, Learning and Teaching. *Cognition and Instruction*, 1-16. DOI : 10.1080/07370008.2020.1769100
- [16] Y. S. Jeong, J. S. Yu, J. Y. Hong, J. S. Lim. (2019). *Theory of Software Education*, SIMAS.

유 정 수(Jeong-Su Yu)

[정회원]

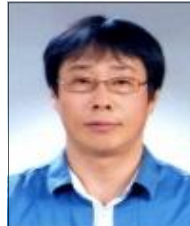


- 1984년 2월 : 전북대학교 전산통계학과(이학사)
- 1986년 2월 : 전북대학교 전산통계학과(이학석사)
- 1999년 2월 : 충남대학교 계산통계학과(이학박사)

- 1992년 3월~현재 : 전주교육대학교 컴퓨터교육과 교수
- 관심분야 : 인공지능 교육, 신경망 모델, 디지털 역량, 미래 교육 공간혁신, 미래교육 플랫폼
- E-Mail : jsyu@jnue.kr

장 용 우(Yong-Woo Jang)

[정회원]



- 1989년 2월 : 우석대학교 체육학과(학사)
- 1995년 2월 : 전북대학교 체육학과(석사)
- 2000년 8월 : 전남대학교 체육학과(박사)

- 2002년 9월~현재 : 전주교육대학교 체육교육과 교수
- 관심분야 : 인공지능융합교육, 운동과학, 트레이닝방법, 근재활
- E-Mail : piapong@jnue.kr