

컴퓨팅 사고력의 과정중심 평가 방안

이정훈¹, 조정원^{2*}

¹제주대학교 대학원 컴퓨터교육전공 학생, ²제주대학교 컴퓨터교육과 교수

Process-oriented Evaluation Method for Computational Thinking

Jeonghun Lee¹, Jungwon Cho^{2*}

¹Student, Major in Computer Education, Graduate School, Jeju National University

²Professor, Dept. of Computer Education, Jeju National University

요 약 소프트웨어 교육은 4차 산업혁명을 이끌 미래인재 양성을 위한 교육으로 주목받고 있다. 유아부터 성인까지 모두를 위한 소프트웨어 교육의 목적은 단순히 프로그래밍 능력을 기르는 것이 아닌, 컴퓨팅을 기반으로 현실 세계의 문제를 효과적으로 해결해 나가는 문제 해결 능력인 ‘컴퓨팅 사고력’을 기르는 것이다. 따라서 어떻게 컴퓨팅 사고력을 함양하고 평가할 것인가는 매우 중요한 사안이다. 본 논문은 학습자가 문제를 해결하는 과정에서 과정중심 수행평가 방식을 적용하여 컴퓨팅 사고력을 평가하는 방식을 제안하였다. 개발된 내용은 컴퓨터과학·컴퓨터교육 전공의 대학 교수 5인, 컴퓨터과학·컴퓨터교육 전공 현직 정보교사 5인으로 구성된 전문가 집단의 2차에 걸친 델파이 조사를 통해 수정·보완하여 최종 모델의 타당성을 검증하였다. 본 논문이 문제 해결 관점에서 컴퓨팅 사고력을 평가하기 위한 연구에 기여할 수 있기를 기대한다.

주제어 : 컴퓨팅 사고력, 평가, 수행평가, 과정중심평가, 문제 해결 과정

Abstract Software education is drawing attention as an education for fostering future talents who will lead the 4th industrial revolution. The purpose of software education for everyone from kinder to adults is not simply to develop programming skills, but to develop "Computational Thinking," a problem-solving ability that effectively solves real-world problems based on computing. Therefore, how to cultivate and evaluate computational thinking is a very important issue. This paper proposed a method of applying a process-based performance evaluation method to evaluate computational thinking ability in the process of solving learners' problems. The developed contents were revised and supplemented through two Delphi surveys by a group of experts consisting of five university professors and five incumbent information teachers majoring in computer science and computer education to verify the effectiveness of the final model. I hope This paper can contribute to the study of evaluating computer thinking ability from the perspective of problem solving.

Key Words : Computational thinking, Evaluation, Performance evaluation, Process-driven evaluation, Problem solving process

*This work was supported by the Ministry of Education of the Republic of Korea and the National Research Foundation of Korea. (NRF-2017S1A5A2A03067937)

*This article is excerpted from the master thesis(Graduation school of education, Jeju National Univ.) of first author.

*Corresponding Author : Jungwon Cho(jwcho@jejunu.ac.kr)

Received September 27, 2021

Revised October 11, 2021

Accepted October 20, 2021

Published October 28, 2021

1. 서론

세계경제포럼(WEF, 2016)은 4차 산업혁명으로 인한 미래사회의 변화를 예고하였다[1]. 산업혁명은 일자리의 변화를 야기하며, 미래 일자리의 변화는 미래 사회를 주도해 나갈 미래인재상의 변화를 뜻한다. 이에 세계경제포럼은 미래인재상이 가져야 할 핵심 역량으로 4C(Creativity, Communication, Critical thinking, Collaboration)를 제시하였다[1]. 이러한 핵심 역량을 기르기 위한 교육으로 소프트웨어 교육이 주목 받고 있다. 우리나라는 2015 개정 교육과정에서 소프트웨어 교육을 강조하며 정보 교과를 필수 과목으로 지정하였다[2]. 유아부터 성인까지 모두를 위한 소프트웨어 교육의 목적은 컴퓨터 과학의 원리를 바탕으로 실생활의 문제를 논리적, 절차적으로 해결해 나가는 문제 해결 능력인 ‘컴퓨팅 사고력’을 함양하는데 있다. 이에 따라 어떻게 컴퓨팅 사고력을 함양하고 평가할 것인가에 대한 고민이 필요한 실정이다. 하지만, 컴퓨팅 사고력의 평가는 ‘사고력’이라는 추상적인 의미를 포함하고 있어 이를 평가하는 것은 쉽지 않다[3].

컴퓨팅 사고력 평가는 매우 중요한 문제이다. 학습자는 평가를 통해 자신의 부족한 부분을 파악하고, 결과에 대한 피드백과 자기성찰을 통해 성장할 수 있다[4]. 즉, 평가를 통해 컴퓨팅 사고력을 향상하고 개선할 수 있다. 최근 평가의 방향은 기존의 평가 방법과 다르게 결과 중심의 평가가 아닌 학습자의 수행을 통해 이루어지고, 학습을 위한 평가인 과정중심 평가로 변화하고 있다[5]. 과정중심 평가는 학습의 과정에 중점을 두고 평가하여 학습자의 수행 과정에서 적절한 피드백을 주며 학습을 강화하는 평가이다. 이를 가장 잘 반영할 수 있는 평가는 수행평가 방식이다[6].

본 논문은 학습자가 문제를 해결하는 과정에서 발현되는 컴퓨팅 사고력을 평가하기 위해 과정중심 수행평가 방식을 적용한 컴퓨팅 사고력 평가 모델을 제안하고자 한다. 이를 위해 선행 연구와 2015개정 정보교과 교육과정을 분석하여 ‘문제 해결 과정’을 제시한다. 또한 성취기준을 바탕으로 평가를 구성하는 과정중심 수행평가를 적용하기 위해 앞서 제시한 문제 해결 과정별로 ‘성취기준’을 제안한다. 마지막으로 학습자의 수준을 파악하고, 학습자의 성취기준 도달 여부를 알려주는 기준을 제공하기 위하여 ‘평가요소’와 ‘평가기준’을 제시하고자 한다. ‘문제 해결 과정’, ‘성취기준’, ‘평가요소’, ‘평가기준’에 대해 2차에 걸친 델파이 조사를 실시하며, 이를 활용한 문제 해결 과정중심의 컴퓨팅 사고력 평가 모델을 제안하고자 한다.

2. 관련 연구

2.1 컴퓨팅 사고력

컴퓨팅 사고력(Computational Thinking, CT)은 Wing교수에 의해 세계적으로 주목을 받기 시작하였다.

2006년 Wing교수는 ‘컴퓨팅 사고력은 컴퓨터 과학의 기본 개념을 이용하여 문제를 해결하는 사고 과정’이라 하였다. 또한 컴퓨팅 사고력을 ‘읽기, 쓰기, 셈하기와 더불어 모든 사람이 갖추어야 할 기본적인 학습 능력이며, 컴퓨팅 시스템에 의해 효과적으로 수행될 수 있는 방법으로 문제와 해결책을 표현하는 것에 관련된 사고 과정’으로 정의하였다[7].

교육부에서는 컴퓨팅 사고력을 ‘컴퓨터 과학의 기본 개념과 원리 및 컴퓨팅 시스템을 활용하여 실생활 및 다양한 학문 분야의 문제를 이해하고 창의적 해법을 구현하여 적용할 수 있는 능력’으로 정의하였다[8].

미국의 CSTA&ISTE(2011)는 ‘현대 사회에서 컴퓨팅이 가진 힘과 제약사항을 이해하고, 새로운 지식을 만들어 내거나, 시스템을 설계하여 문제를 해결하는 능력’으로 정의하였다[9].

이를 통해 살펴본 컴퓨팅 사고력의 정의는 ‘실생활에서 접할 수 있는 문제 상황에서 컴퓨터 과학의 개념과 원리를 기반으로 문제를 해결해 나가는 사고 과정 및 절차적 능력’이라 할 수 있다.

2.2 문제 해결 과정

우리는 살아가면서 수많은 문제를 접하고 해결해 나간다. 문제가 있다는 것은 ‘Fig. 1’과 같이 내가 목표로 하는 상태와 현재의 상태가 같지 않은 상태이다. 여기서 현재 상태는 현재 어떻게 되어있는가에 대한 상황, 목표 상태는 어떻게 되었으면 좋겠는가에 대한 전망 등으로 가정할 수 있으며, 현재 상태와 목표 상태간의 차이가 있다는 것은 문제 상황에 놓여있다는 것이다[10].



Fig. 1. Relationship between the current state and the target state[10]

문제를 인식하기 위해서는 현재 상태와 목표 상태 간의 차이를 인식할 수 있어야 한다. 즉, 현재 상태와 목표

상태를 인식하여 차이(GAP)를 발견할 수 있으면 남들이 발견하지 못하는 문제를 발견할 수 있다. 우리는 이를 문제 인식이라 한다. 문제를 해결한다는 현재 상태와 목표 상태간의 차이가 발생한 원인을 도출해 내고 그 차이를 줄일 수 있는 솔루션을 마련하는 것이다[10].

Polya(1957)는 문제 해결을 위해 자신이 배운 지식과 원리를 응용하여 문제의 해결 방안을 발견해 나가는 것이라 말하며 문제 해결 단계를 ‘문제 이해’, ‘계획 작성’, ‘계획 실행’, ‘반성’으로 나누어 제시하였다[11].

Osborn(1953)은 문제 해결을 위해서는 먼저 해결해야 할 문제가 무엇인지 파악하는 것이 중요하며, 문제가 파악이 되면 문제와 관련한 자료를 수집, 분류, 분석해야 한다고 하였다. Osborn은 문제 해결 단계를 ‘오리엔테이션’, ‘준비’, ‘분석’, ‘가설’, ‘부화’, ‘종합’, ‘확인’ 7단계로 나누어 제시하였다[12].

각각의 문제 해결 단계를 분석해보면 일반적으로 문제 해결 단계는 문제 인식 및 이해에서 출발하여 문제 해결을 위한 자료를 수집, 분석하여 해결안을 도출해 내고 해결안을 실행하여 평가로 끝나는 구조를 가지고 있다.

2.3 과정중심 수행평가

미래사회에 요구되는 핵심 역량을 함양하기 위해 평가 패러다임도 변화하고 있다. 우리나라는 2015개정 교육과정에서 창의·융합형 인재를 양성하기 위해 과정중심 평가를 강조하고 있다. 과정중심 평가는 ‘교육과정을 기반으로 평가 계획에 따라 교수·학습 과정에서 학생의 변화와 성장에 대한 자료를 다각도로 수집하여 적절한 피드백을 제공하는 평가’이다[13]. 즉, 평가를 통해 학습하게 되는 형태인 ‘학습을 위한 평가’를 의미하는 것이다. 이러한 평가를 잘 반영할 수 있는 평가 방법은 수행평가 방식이다. 수행평가는 실생활에서 일어날 수 있는 문제 상황에서 개인이 습득한 지식을 어떻게 적용해 문제를 해결하는지 종합적으로 판단하는 평가방식으로, 학습의 결과와 함께 과정을 중시하는 평가 방식이다[13].

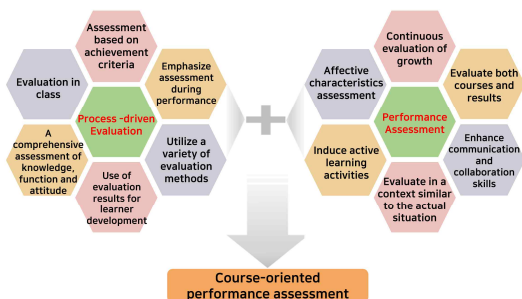


Fig. 2. Process-oriented performance evaluation

기존의 평가 체계는 측정, 분류, 배치를 목적으로 이루어지며 결과를 중시하지만 과정중심 수행평가는 지도, 조언, 개선을 목적으로 이루어지며 과정을 중시하는 평가 체계이다. 최근의 평가 패러다임은 ‘Fig. 2’와 같이 과정을 중시하는 수행평가를 강조한다. 과정중심 수행평가가 가지는 장점은 실생활 문제와 유사한 상황에서 평가를 진행하여 문제 해결력을 기를 수 있으며 학습자가 스스로 문제를 해결함으로써 능동적인 학습활동을 유도할 수 있다. 또한, 문제 해결 과정과 결과물을 동시에 평가하여 과정과 결과를 동시에 평가할 수 있다. 이러한 평가 과정에서 적절한 피드백을 통해 학습자의 행동을 개선하여 성장을 기대할 수 있을 것이다[6].

3. 컴퓨팅 사고력의 과정중심 평가 방안

3.1 연구 방법 및 절차

컴퓨팅 사고력의 의미와 과정중심 평가의 중요성을 생각하면 컴퓨팅 사고력의 평가는 문제 해결 과정에서 이루어지는 것이 바람직하다. 따라서 본 논문에서는 과정중심 수행평가 방식을 적용하여 컴퓨팅 사고력을 평가하는 방안을 제안하고자 한다. 이를 위해 ‘문제 해결 과정’, ‘문제 해결 과정별 성취기준’, ‘평가요소’, ‘평가기준’을 개발하였다. 개발된 내용은 컴퓨터과학·컴퓨터교육 전공의 대학 교수 5인, 컴퓨터과학·컴퓨터교육 전공 현직 정보교사 5인으로 구성된 전문가 집단의 피드백을 통해 수정·보완하여 최종 모델을 개발하였다.

3.2 전문가 델파이 조사

3.2.1 전문가 집단 주요 피드백

전문가 집단의 주요 피드백은 다음과 같다.

첫째, 학습자의 문제 해결 과정에서 문제 인식 단계와 문제 분석단계를 구분해야 한다. 문제를 해결함에 있어 현재 상태와 목표 상태의 차이점을 인식하고 무엇을 해결해야 하는지 정의하는 과정은 중요하다. 이를 통해 문제의 원인을 파악하고 문제 해결을 위한 설계가 이루어져야 한다.

둘째, 문제 해결을 위한 알고리즘을 설계하기 전에 문제 해결 모델링 단계가 필요하다. 모델링은 현실의 문제를 컴퓨터 사이언스로 해결하기 위해 컴퓨터 세계로 가져가는 핵심 절차로, 문제 해결을 위한 알고리즘은 모델링에 따라 완전히 달라진다.

셋째, 문제 해결 단계별로 컴퓨팅 사고력의 핵심 요소를 세분화 하여 표기할 필요가 있다. 기존에 컴퓨팅 사고력 핵심 요소는 자기성찰을 위한 ‘평가’나 문제 해결을 위한 ‘공유’의 개념을 담기 어려운 부분이 있다.

본 논문은 주요 피드백을 반영하여 ‘문제 해결 과정’, ‘문제 해결 과정별 성취기준’, ‘평가요소’, ‘평가기준’을 개발하였다.

3.2.2 타당도 검증

전문가 델파이 조사를 통한 통계 분석은 평균, 중앙값, 표준편차, 1사분위수, 3사분위수를 산출하고 내용 타당도 비율(Content Validity Ratio, CVR)와 수렴도(Convergence), 합의도(Consensus), 안정도(Stability)를 계산하여 산출하였다. 여기서 CVR 기준치는 델파이 전문가 수가 10명이므로 0.62를 기준으로 설정하였다. 2차에 걸친 델파이 조사는 ‘Table 1’과 같다.

Table 1. Expert Delphi Research Results

Category	Q	Cv	Cr	CVR	S
Problem solving process and definition of each step	1.1	0.38	0.85	0.80	0.14
	1.2	0.38	0.85	1.00	0.10
	1.3	0.38	0.85	0.80	0.14
	1.4	0.00	1.00	1.00	0.08
	1.5	0.00	1.00	1.00	0.00
	1.6	0.00	1.00	1.00	0.00
	1.7	0.00	1.00	1.00	0.00
CT Key Element	2.1	0.50	0.75	0.80	0.15
	2.2	0.00	0.96	0.95	0.05
	2.3	0.38	0.85	1.00	0.10
	2.4	0.00	1.00	1.00	0.06
	2.5	0.00	1.00	0.9	0.06
Achievement Criteria	2.6	0.00	1.00	1.00	0.00
	3.1	0.38	0.85	1.00	0.10
	3.2	0.19	0.92	0.9	0.11
	3.3	0.50	0.80	1.00	0.11
	3.4	0.00	1.00	0.80	0.13
	3.5	0.12	0.95	0.93	0.10
Evaluation Factor	3.6	0.00	1.00	1.00	0.07
	4.1	0.19	0.92	0.9	0.07
	4.2	0.00	1.00	1.00	0.02
	4.3	0.38	0.85	0.80	0.14
	4.4	0.00	1.00	1.00	0.06
	4.5	0.00	1.00	1.00	0.00
	4.6	0.00	1.00	1.00	0.00
	4.7	0.00	1.00	1.00	0.00
4.8	0.00	1.00	0.80	0.13	

Rating Criteria Hierarchy	4.9	0.16	0.92	1.00	0.08
	4.10	0.00	1.00	1.00	0.03
	4.11	0.00	1.00	1.00	0.03
	4.12	0.00	1.00	1.00	0.03
	5.1	0.00	1.00	1.00	0.06
	5.2	0.38	0.85	0.80	0.14
	5.3	0.00	1.00	1.00	0.08
	5.4	0.00	1.00	0.80	0.13
	5.5	0.00	1.00	1.00	0.00
	5.6	0.00	1.00	1.00	0.06
	5.7	0.00	1.00	1.00	0.06
	5.8	0.00	1.00	1.00	0.06
Assessment Criteria Behavior Statement	5.9	0.00	1.00	1.00	0.06
	5.10	0.00	1.00	1.00	0.08
	5.11	0.00	1.00	1.00	0.06
	5.12	0.00	1.00	1.00	0.06
	6.1	0.00	1.00	1.00	0.07
	6.2	0.00	1.00	0.93	0.08
	6.3	0.00	1.00	1.00	0.04
	6.4	0.00	1.00	1.00	0.04
	6.5	0.00	1.00	1.00	0.05
	6.6	0.00	1.00	1.00	0.04
	6.7	0.00	1.00	1.00	0.06
	6.8	0.00	1.00	1.00	0.04
6.9	0.00	1.00	1.00	0.04	
6.10	0.00	1.00	1.00	0.04	
6.11	0.00	1.00	1.00	0.07	
6.12	0.00	1.00	1.00	0.07	

수렴도는 0.5를 넘지 않았으며, 합의도 또한 최솟값 0.75로 안정권에 든 것을 알 수 있다. CVR을 살펴보면, 기준치인 0.62를 넘는 결과가 산출되었고, 안정도 또한 0.2이하로 더 이상 설문을 진행하지 않아도 되는 수준으로 타당도를 확보하였다.

3.3 컴퓨팅 사고력 기반 문제 해결 과정

과정중심 수행평가는 실생활과 비슷한 인위적인 문제 상황에서 학습자가 습득한 지식을 어떻게 적용하여 문제를 해결하는지 과정에 초점을 둔 평가 방법이다. 컴퓨팅 사고력 또한 컴퓨팅 과학을 기반으로 문제를 해결해 나가는 사고과정, 절차이기 때문에 과정중심 수행평가로 평가하기 적절하다. 이를 토대로 본 논문은 문제 해결 과정을 정의하기 위해 컴퓨팅 사고력을 기반의 문제 해결 과정을 정의한 선행 연구들을 분석하였다.

최숙영(2016)은 컴퓨팅 사고력과 관련하여 문제 해결 과정을 살펴보는 것은 컴퓨팅 사고력을 이해하는데 도움이 될 수 있다 하였고, 컴퓨팅 사고력의 핵심 요소를 뽑

아내고 문제 해결 과정을 ‘문제인식’, ‘문제분석’, ‘해결책 고안’, ‘문제 해결의 실천’, ‘문제 해결의 평가’ 5단계로 나누어 제시하였다[14].

이철현(2016)은 CT-PS 모델을 제시하며 문제 해결 과정에서 컴퓨팅 사고력 핵심 요소를 경험하고 습득함으로써 컴퓨팅 사고력을 향상시킬 수 있다고 하였다. 이철현의 문제 해결 과정은 크게 ‘문제분석’, ‘추상화’, ‘구현’ 3단계로 나뉜다[15].

전용주(2017)는 창의적 문제 해결 과정의 각 단계에서 컴퓨팅 사고력 핵심 요소를 적용하여 컴퓨팅 사고력 기반의 창의적 문제 해결 과정을 제시하였다. 제시한 단계는 ‘문제인식 및 분석’, ‘아이디어 구상’, ‘설계’, ‘구현 및 평가’로 나누어 제시하였다[16].

각각의 연구 결과를 분석하면 문제 인식 및 분석 과정에서 시작하여 해결책을 탐색하고 알고리즘을 설계하여 프로그래밍을 통해 자동화하여 구현하고 평가하는 단계를 거치는 것을 알 수 있다.

본 논문에서 제시하는 컴퓨팅 사고력 기반 문제 해결 과정은 기존 문헌 연구 및 2015개정 교육과정을 분석하여 개발하였으며 전문가 델파이조사를 통해 수정 및 보완하였다. 개발된 컴퓨팅 사고력 기반 문제 해결 과정은 ‘Table 2’과 같다.

Table 2. Proposed Computational Thinking-Based Problem Solving Process

Problem Solving Process	CT Key Element
Problem Recognition	Abstraction (key element extraction)
Problem Analysis	Collecting Data
	Analyze Data
	structuring
	Abstraction (Decomposition)
Solution Design	Abstraction (key element extraction)
	Abstraction (modelling)
Algorithm Design	Abstraction (Algorithm)
Program Implementation	automation(coding)
	automation(simulation)
Share and Evaluation	Generalization
	Evaluation

문제 해결 과정은 6단계로 정의하였다.

문제 인식 단계는 문제를 발견하고 문제 상황의 현재 상태와 목표 상태를 파악하는 단계이다.

문제 분석 단계는 문제와 관련된 자료를 수집, 공유, 관리하며, 문제의 원인을 파악하고 문제의 규모를 줄여 분해하는 단계이다.

해결책 구상 단계는 문제 해결을 위해 중요한 핵심 요소를 추출하고 패턴을 파악하거나 다양한 해결 방법을 적용하기 위해 문제를 탐색하여 문제 해결 모델을 설계해 나가는 단계이다.

알고리즘 설계 단계는 문제 해결 모델의 세부 절차를 의사코드나 순서도로 표현하는 단계이다.

프로그램 구현 단계는 설계된 알고리즘을 토대로 문제 해결 모델을 자동화하는 단계이다.

마지막으로 공유 및 평가는 최종 결과물을 공유하고 피드백을 받으며, 문제 해결 방법과 유사한 다른 문제에 어떻게 적용할지 검토하는 단계이다.

문제 해결의 각 단계에는 컴퓨팅 사고력 핵심 요소가 발현되며, 핵심 요소는 교육부가 제시한 컴퓨팅 사고력 구성 요소를 기반으로 개발하여 제시하였다.

3.4 문제 해결 과정별 성취기준 개발

과정중심 수행평가는 학습을 위한 평가로, 평가와 교육과정, 교수학습이 연계된 평가방식이다. 교육과정은 성취기준을 준거로 교수 학습을 위한 계획과 교수 학습을 위한 평가 계획을 세워야 한다. 즉 과정중심 수행평가가 제대로 이루어지기 위해서는 성취기준이 제시되어야 한다[5]. 2015개정 정보교과 교육과정에서는 교육 내용의 영역별로 성취기준을 제시하고 있으며 본 논문에서는 정보교과에서 제시하는 성취기준과 기존 문헌 분석을 통해 문제 해결 과정별 성취기준을 ‘Table 3’과 같이 재구성하였다. 개발된 성취기준은 전문가 델파이를 통해 수정 및 보완이 이루어졌다.

3.5 성취기준 기반 평가요소 추출

성취기준은 교육 내용을 통해 배워야 할 지식과 지식을 통해 할 수 있기를 기대하는 수행능력을 서술한 것으로, 내용(지식)과 기능(수행, 태도)로 이루어져있다[2]. 즉, 성취기준을 분석하면 학습자에게 무엇을 가르칠 것인지, 무엇을 평가할 것인지 추출할 수 있다. 본 논문에서는 앞서 개발된 성취기준을 분석하여 내용과 기능으로 분해하고 평가요소를 추출하였다. 학습요소와 평가요소 추출 방법은 ‘Table 4’와 같다.

Table 3. Achievement criteria by problem-solving process

Problem Solving Process	Achievement Criteria
Problem Recognition	01-01 Find problems in real life and understand the current state and goal state.
Problem Analysis	02-01 Collect, store, manage, and share problem-related data, and express it structured in tables and diagrams.
	02-02 To reduce the complexity of the problem, decompose it into smaller solvable problems.
Solution Design	03-01 It extracts patterns or key elements necessary for problem solving, suggests ideas for problem solving, or explores solutions to other problems similar to the current problem.
	03-02 Ideas for problem solving are modeled for computer implementation.
Algorithm Design	04-01 Explore different methods and procedures for solving problems and present them in visually clear algorithms.
	04-02 Use control structures to logically design algorithms for problem-solving models.
Program Implementation	05-01 Select a programming language to implement the designed algorithm. and It is implemented by automation according to the characteristics of the development environment and programming language.
	05-02 Write a program that receives data, processes it, and outputs it.
	05-03 When an error occurs, the program or algorithm is analyzed to find the cause of the error. And fix the errors to make it executable.
Share and Evaluation	06-01 Check whether the program was developed through collaboration with colleagues, such as 'sharing roles', 'execution of missions', and 'communication between team members'.
	06-02 Share and evaluate results. And have an attitude of giving feedback through reflection.

Table 4. Extract learning and evaluation elements

Achievement Criteria	
01-01 Find problems in real life and understand your current state and target state.	
Contents	Skill
Real life, problem, current state, goal state	discover, understand
Extraction of Learning Elements	
- Finding problems in real life - Understanding your current state - Understanding goal states	
Extract Evaluation Factors	
- Can you spot problems in real life? - Can you understand and describe your current state and goal state?	

본 논문에서 제시하는 성취기준 '01-01 실생활에서 문제를 발견하고 현재 상태와 목표 상태를 이해한다.'에서 내용은 '실생활', '문제', '현재 상태', '목표 상태', 기능은 '발견하기', '이해하기'로 나눌 수 있다. 이를 통해 학습요소를 추출하면 '실생활에서 문제 발견하기', '현재

상태 이해하기', '목표 상태 이해하기'로 추출할 수 있다. 평가요소는 '실생활에서 문제를 발견할 수 있는가?', '현재 상태와 목표 상태를 이해하고 설명할 수 있는가?'로 추출할 수 있다. 위와 같은 방법으로 각 성취기준별 평가요소를 추출하면 'Table 5'와 같다.

Table 5. Evaluation Elements by Achievement standard

AC	Evaluation Factor
01-01	01-01-01 Can you spot problems in real life?
	01-01-02 Can you understand and describe the current state and target state of the problem?
02-01	02-01-01 Can you gather data related to the problem?
	02-01-02 Can the collected data be stored, managed, and shared on a computer?
02-02	02-01-03 Can the collected data be stored, managed, and shared on a computer?
	02-02-01 Can you decompose the problem into solvable unit problems?
03-01	03-01-01 Can you extract recurring patterns or key elements from the problem or data?
	03-02-02 Can you provide an idea for solving the problem through a pattern or key element? Or can you explore and apply a solution to a problem similar to your current one?
03-02	03-02-01 Can you formulate ideas for solving problems and design models for solving problems?
04-01	04-01-01 Can you explore, compare, and analyze different problem-solving methods and procedures?
	04-01-02 Can you articulate your problem-solving procedure as an algorithm using text, pictures, etc.?
04-02	04-02-01 Is it possible to logically design a problem-solving algorithm using control structures?
05-01	05-01-01 Can it be implemented by automating the algorithm designed according to the characteristics of the development environment and programming language?
	05-02-01 Is it possible to implement a program without logical errors using control structures, calls, variables, and operators?
05-02	05-02-02 Is it possible to implement a program that receives and processes various types of data to obtain a desired output value?
	05-03-01 Can you systematically analyze and find the cause of the error?
05-03	05-03-02 Can the error be corrected in a program-executable form?
	06-01-01 In the problem-solving process, can each person divide their roles and accomplish the assigned task to achieve the task goal?
06-01	06-01-02 Can you share, share, and stay in touch with ideas and opinions for solving problems?
	06-02-01 Can you share the results with your co-workers and talk about the good things and the bad things?
06-02	06-02-02 Can you reflect on your problem-solving process and find and discuss similar issues?

3.6 성취기준 기반 평가기준 개발

평가기준이란 학습자를 객관적으로 평가하기 위한 기준이 되는 것으로 학습자들이 무엇을 할 수 있는지, 무엇

을 알고 있는지를 판단하기 위한 기준이 되는 것을 의미한다[17]. 이러한 평가기준은 학습자의 행동을 보고 평가자가 판단할 수 있도록 행동 동사를 명확하게 하는 것이 중요하다. 본 논문에서는 평가기준 개발을 위해 ब्ल록의 개정된 디지털 텍사노미를 분석하여 개발하였다[18].

성취기준에 평가기준은 ‘상/중/하’ 세 가지 위계로 나누어 제시하였다. ‘상’수준은 ‘우수하게 문제를 해결할 수 있는 수준’, ‘중’수준은 ‘문제 해결이 가능한 수준’, ‘하’수준은 ‘문제 해결을 하지 못하는 수준’으로 설정하였다. 개발된 평가기준은 전문가 델파이를 통해 수정 보완이 이루어졌으며 ‘Table 6’과 같다.

Table 6. Evaluation Standard by Achievement standard (H:high, M:medium, L:low)

AC.N	Evaluation Standard	
01-01	H	You can find problems in real life, describe current and target states in problem situations, and infer the causes of problems.
	M	Able to discover problems in real life and describe current and target states in problem situations.
	L	I have found a problem in real life, but I am unable to describe my current state and goal state in the problem situation.
02-01	H	You can search for problem-related data, classify it according to the type and nature of the data, and store, manage, and share it on your computer.
	M	You can search for problem-related data, store, manage, and share it on your computer, and visually express the collected data in tables and diagrams.
	L	They lack the ability to retrieve data related to a problem and store it on a computer or present it in a visual form such as a table or diagram.
02-02	H	A problem can be described by dividing it into subproblems of units that can be solved, and can describe the connectivity between subproblems.
	M	Divide the problem into subproblems of solvable units and describe the subproblems.
	L	Failure to divide the problem into solvable subproblems.
03-01	H	By extracting patterns or key elements from problems and data, you can discover and present ideas for how to solve problems, and explore, restructure, and apply existing problem solving methods that are similar to problems.
	M	You can find a way to solve a problem by extracting what you need to solve it from your problem and data, or you can find solutions to previous problems that are similar to your problem.
	L	We've analyzed the problem and data, but can't find a pattern or a solution to a key element or another problem that is similar to the current problem.
03-02	H	Decomposed problems can be structured and expressed, and ideas for solving problems can be realized by computer by designing problem-solving models.
	M	I can tell you how to implement an idea to solve a problem on a computer.
	L	It fails to tell you how to implement an idea for solving a problem on a computer.

04-01	H	By exploring and comparing various problem-solving methods and procedures, you can choose an efficient procedure and express it easily and clearly in a visual form using text and pictures.
	M	Problem solving methods and procedures can be easily and clearly expressed in a visual form.
	L	Problem solving methods and procedures are visually expressed, but not clearly expressed.
04-02	H	Logic algorithms can be designed using control structures.
	M	In designing a logical algorithm using the control structure, unnecessary parts are included or necessary parts are omitted.
	L	It is not possible to design a logical algorithm using a control structure.
05-01	H	The designed algorithm can be implemented by automating the same structurally and logically according to the development environment and characteristics of the programming language.
	M	It is implemented according to the development environment and characteristics of the programming language, but only a part of the designed algorithm can be implemented.
	L	The designed algorithm cannot be implemented by automating it.
05-02	H	In order to receive and process various types of data according to the purpose of the program, it is possible to implement a program that outputs the desired result by logically using sequential, selection, and repeating structures, calls, variables, and operators.
	M	It is implemented according to the purpose of the program using sequential, selection, and repeating structures, calls, variables, and operators, but unnecessary blocks or objects are included.
	L	It was not possible to implement a program that receives, processes, and outputs data according to the purpose of the program.
05-03	H	In the event of an error, it is possible to systematically find the cause of the error and correct it in an actionable form, such as continuously trying the test and checking the code.
	M	Random error detection allows you to find the cause of the error and correct the error in an actionable form.
	L	An error occurred, but the cause of the error could not be found and the error could not be corrected.
06-01	H	Roles were shared in the problem-solving process, and each other's tasks were identified to achieve a common goal, and ideas and opinions were continuously shared and communicated.
	M	In the process of problem-solving, opinions among team members were not shared, so the program was developed with an overloaded task given to one side due to unbalanced role division.
	L	In the process of problem solving, roles were not divided, and opinions among team members were not shared.
06-02	H	Through peer evaluation, they actively present their opinions on the strengths and weaknesses of the results and points for improvement. Based on peer feedback and self-evaluation, they reflect on their own problem-solving process and take an attitude of reflection and feedback.
	M	Through peer evaluation, simple opinions are presented on the strengths and weaknesses of the results and points for improvement, but the attitude of reflection based on peer feedback and self-evaluation is insufficient.
	L	They do not give opinions on the results of peer evaluation and do not reflect on their own problem-solving process.

3.7 과정중심의 컴퓨팅 사고력 평가

본 논문에서 제시하는 문제 해결 과정중심의 컴퓨팅 사고력 평가에 대한 개념도는 'Fig. 3'과 같다. 평가자는 본 논문에서 개발된 문제 해결 과정별 성취기준의 내용을 고려하여 수행 과제를 개발하고 평가 및 피드백 계획을 세운다. 수행 과제는 학습자의 수준을 고려하여 실생활 문제를 다루어야 한다.

평가 계획은 본 논문에서 개발된 평가요소와 평가기준을 반영하여 지식, 기능, 태도 측면에서 평가가 이루어지도록 설계하여야 한다. 평가기준은 학습자에게 미리 안내되어 평가에 참여하도록 유도하여야 한다. 평가는 학습자의 문제 해결 과정에서 관찰평가, 자기평가, 포트폴리오 평가 등의 다양한 방법으로 평가를 진행하고, 수행을 통한 결과물에 대해 동료평가, 결과물 평가가 이루어져 과

정과 결과를 동시에 평가하는 것이 바람직하다.

피드백 계획은 학습자의 수행 과정에 대한 관찰과 기록을 통해 진행된다. 평가요소 및 평가기준을 토대로 학습자에게 부족한 부분을 파악하여 적절한 피드백을 제공하여야 한다.

평가는 학습자가 문제 해결을 수행하는 과정에서 이루어진다. 학습자의 문제 해결 과정은 문제를 발견하는 것으로부터 시작한다. 주어진 문제 상황에서 문제를 발견하고 인식하여 문제의 원인을 파악하는 능력을 평가한다.

문제 분석 단계에서는 문제와 관련된 자료를 수집, 관리하고 공유한다. 수집한 자료를 토대로 팀원 간의 의사소통을 통해 문제를 분석하고 문제의 규모를 줄이는 능력을 평가한다.

해결책 구상 단계에서는 문제 해결을 위한 핵심요소와

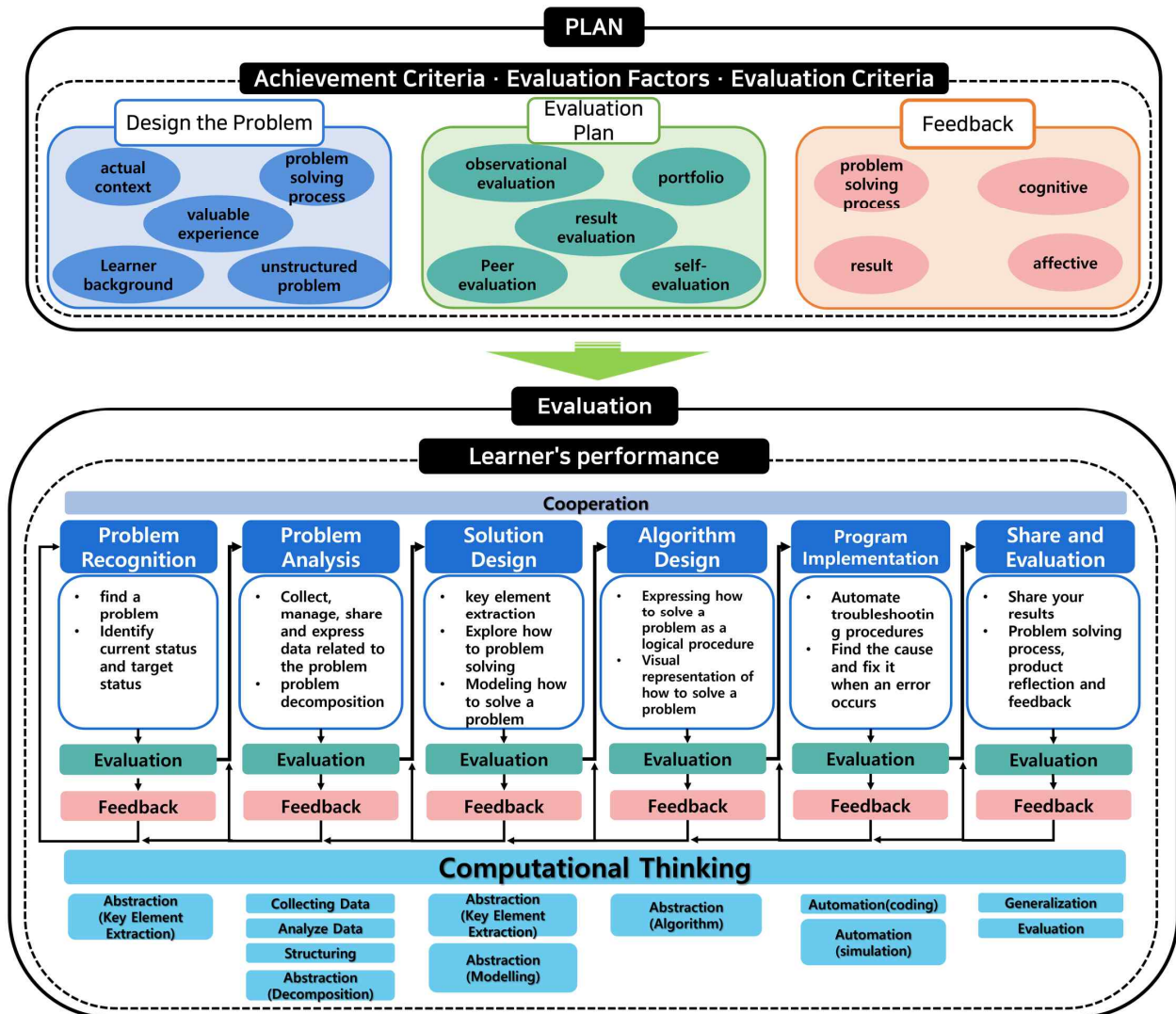


Fig. 3. Problem Solving Central Computational Thinking Model

패턴을 찾고 다양한 아이디어를 통해 문제 해결 모델을 설계하는 능력을 평가한다.

알고리즘 설계 단계에서는 문제 해결 모델을 시각화하고 알고리즘을 설계하는 능력을 평가한다.

프로그램 구현 단계에서는 설계된 알고리즘을 프로그래밍으로 자동화하는 능력을 평가한다. 이때, 팀으로 구성되어 있으면 혼자 프로그래밍을 하지 않고 의사소통을 하며 각자 프로그램을 제작하도록 유도한다.

공유 및 평가 단계에서는 자동화된 프로그램을 공유하고 서로 의견을 나누는 시간을 갖도록 한다. 또한 문제 해결 과정을 전반적으로 돌아보고 자기평가, 토의 등을 갖는 것도 중요하다.

학습자의 문제 해결 과정의 각 단계에서 다양한 평가와 피드백이 이루어지고, 이러한 평가를 통해 학습자의 성장을 도모할 수 있다.

4. 결론

소프트웨어 교육이 미래인재 양성을 위한 교육으로 떠오르며, 컴퓨팅 사고력을 어떻게 함양하고 평가할 것인가는 유아부터 성인까지의 전 연령대에서 매우 중요한 문제이다. 최근 평가의 패러다임은 학습의 결과를 평가하는 것만이 아닌, 학습자의 수행 과정을 중심으로 평가하는 방안이 강조되고 있다. 문제 해결 과정에서 발현되는 컴퓨팅 사고력 또한 학습자가 컴퓨터 과학의 기본 원리를 이용하여 어떻게 문제를 해결해 나가는지 과정을 중심으로 평가해야한다. 따라서 본 논문에서는 컴퓨팅 사고력 평가를 위해 과정중심 수행평가 방법을 적용한 모델을 제안하였다.

본 논문을 통해 얻은 시사점은 다음과 같다.

첫째, 컴퓨팅 사고력은 컴퓨팅 원리를 기반으로 문제를 해결해 나가는 역량이다. 따라서 학습자의 문제 해결 과정에서 평가하는 것이 바람직하다.

둘째, 과정중심 평가를 통해 학습자에게 적절한 피드백을 제공하여 학습자의 컴퓨팅 사고력을 향상시킬 수 있는 방안에 대한 연구가 활발히 이루어져야 한다.

최근 교육부에서도 과정중심 수행평가를 강조함에 따라 교육 현장에서 과정중심 수행평가를 위한 노력이 이어지고 있다. 그러나 명확한 평가의 기준이 없어 평가의 준거가 교사 개인의 역량에 의해 개발되기 때문에 교사에 평가기준에 따라 평가의 결과가 달라질 수 있다는 문제가 있다. 본 논문에서 제안한 과정중심의 컴퓨팅 사고

력 평가 방안을 통해 보다 객관적인 수행평가가 이루어질 수 있을 것이다. 앞으로 문제 해결 과정에서 어떻게 컴퓨팅 사고력을 평가하고 함양시킬 것인지에 대한 추가 연구가 보다 활발히 이루어지기를 기대한다.

REFERENCES

- [1] The World Economic Forum. (2018). *The Future of Jobs Reprot 2018*.
- [2] Ministry of Education. (2015), *2015 Revised Curriculum General*. Ministry of Education.
- [3] H. C. Kim et al. (2017). Development of a scoring rubric based on Computational Thinking for evaluating students' computational artifacts in programming course. *Journal of Computer Education Society*, 20(2), 1-11.
- [4] J. W. Choi & Y. J. Lee. (2014). The Design of Method for Evaluating Computational Thinking. *Journal of Academic Announcement of the Korean Computer Information Society*, 22(2), 177-178.
- [5] K. H. Jeon. (2016). *The direction and task of process-oriented performance evaluation*. korea institute of education and development institute.
- [6] Ministry of Education. (2017). *How to do performance evaluation that emphasizes the process*. The Korea Institute of Curriculum and Evaluation.
- [7] Wing M. Jeannette. (2016). Computational Thinking. *COMMUNICATIONS OF THE ACM March 2006*, 49(3), 33-35.
- [8] Ministry of Education. (2015). *2015 Revised Computers and Information Curriculum*. Ministry of Education.
- [9] ISTE & CSTA. (2011). *Computational Thinking Leadership Toolkit 1st edition*.
- [10] Kenichi Omae. (2005). *Problem Solving Approach (translated by Kim Young-chul)*. Ilbit publishing.
- [11] Polya, G. (1986). *How will you solve the problem? (Translated by Woo Jung-ho)*. Seoul: Genius Education Co Ltd. (original publication:1956).
- [12] Osborne, A. F. (1953). *Applied Imagination: Principles and Procedures of Creative Thinking*. NEW YORK: Charles Scribner's sons.
- [13] J. H. Im & W. S., Choi. (2018). A Study on Characteristics and Meanings of 'Process-focused Evaluation' in the Free Semester Program. *Korean Education*, 45(3), 31-59.
- [14] S. Y. Choi. (2016). A Study on Teaching-learning for Enhancing Computational Thinking Skill in terms of Problem Solving. *Journal of the Korea Computer Education Association*, 19(1), 53-62.

- [15] C. H. Lee. (2016). Development of Computational Thinking based Problem Solving Model(CT-PS Model) for Software Education. *Korean Society for Practical Education Research*, 22(3), 97-117.
- [16] Y. J. Jeon. (2017). *The development and application of a CT-CPS(computational thinking-based creative problem solving) instructional model for the software education of new curriculum*. Doctoral dissertation. Korean National University of Education, Cheongju.
- [17] K. H. Kim. (2016). *A research on the development of evaluation standards for information subjects according to the 2015 revised curriculum*. Korea Institute of Curriculum and Evaluation.
- [18] B. K. Kye & J. O. Kim. (2013). *Bloom's Digital Taxonomy. 2013 KERIS Issue Report Research*. Korea Educational&Research Information Service.

이 정 훈(Lee, Jeonghun)

[정회원]



- 2018년 2월 : 제주대학교 전자공학과 (공학사)
- 2020년 8월 : 제주대학교 컴퓨터교육 전공(교육학석사)
- 2020년 9월 ~ 현재 : 제주대학교 컴퓨터교육전공 박사과정
- 관심분야 : 정보·컴퓨터(SW, AI)교육,

AI, 컴퓨팅 사고력, 컴퓨팅 사고력 및 AI 교육 평가
· E-Mail : 2ehdrks@jejunu.ac.kr

조 정 원(Cho, Jungwon)

[중신회원]



- 2004년 2월 : 한양대학교 전자통신전과공학과(공학박사)
- 2004년 9월 ~ 현재 : 제주대학교 컴퓨터교육과 교수
- 2020년 3월 ~ 현재 : 한국컴퓨터교육학회 부회장, 논문지편집위원장
- 2012년 12월 ~ 현재 : 한국정보과학

회 전산교육시스템연구회 위원장
· 2018년 7월 ~ 현재 : 제주대학교 지능소프트웨어교육센터 센터장
· 2021년 1월 ~ 현재 : 한국디지털정책학회 인공지능연구회 위원장
· 관심분야 : 정보·컴퓨터(SW, AI)교육, 지능정보윤리, 지능형 시스템, 멀티미디어
· E-Mail : jwcho@jejunu.ac.kr