컴퓨터교육학회 논문지 2026년 제29권 제6호
https://doi.org/10.32431/kace.2026.29.6.008

문항반응이론 기반 컴퓨팅 사고력 평가의 타당성 및 동등화 가능성 검토*

Examining Validity and Score Comparability of Computational Thinking Assessment

김의겸† · 신효정††
Euigyum Kim† · Hyo Jeong Shin††

요약

21세기 핵심 역량으로 부각되는 컴퓨팅 사고력(CT)에 대한 관심이 높아지고 있는 가운데, 실제 교육 및 산업 현장에서 널리 활용되고 있는 CT 평가를 위한 텍스트 프로그래밍 언어(TPL) 기반의 알고리즘 문제 해결력 검사에 관한 경험적 타당화 연구는 부족한 실정이다. 이에 본 연구는 국내 주요 대기업에서 최근 4년간 실시한 대규모 알고리즘 문제 해결력 검사 데이터에 대해 문항반응이론(IRT) 기반 Rasch 모형과 가중우도추정법을 적용하여 CT의 측정 및 평가를 위한 해당 검사의 타당성과 신뢰성을 검증하였다. 연구 결과, 문항 난이도 모수 및 피험자 잠재 능력치 분포의 안정성을 확인하였으며, 이를 기반으로 마련한 검사의 등급화 체계에 대한 타당성을 실증하였다. 본 연구는 그동안 경험적 타당화와 측정학적 기반이 미흡했던 TPL 기반 알고리즘 문제 해결력 검사에 대해 IRT 모형을 적용함으로써 증거 기반 평가의 신뢰성과 타당성을 제고하였으며, CT 중심 SW 교육평가의 고도화, 산업계의 공정한 채용 도구, 자격인증 체계 및 국가수준 검사 도구 개발에 실질적으로 기여할 수 있는 방법론적 근거와 시사점을 제시하였다.

주제어 알고리즘 문제 해결력, 컴퓨팅 사고력, 소프트웨어 교육, 문항반응이론, 검사 동등화

ABSTRACT

Despite the widespread use of algorithmic problem-solving tests based on text-based programming languages (TPL) for assessing computational thinking (CT) in both educational and industrial sectors, empirical validation of such an instrument remains limited. To address this gap, this study analyzes large-scale algorithmic problem-solving assessment data using item response theory to estimate item parameters and examinee abilities, while also establishing a framework for score comparability through equating procedures. The findings confirm that the Rasch model provided reliable estimates of item difficulty and latent abilities, and a grading system developed from these estimates demonstrated significant potential for score comparability. Overall, this study advances evidence-based assessment practices for TPL-based algorithmic problem-solving tests and provides a methodological foundation for enhancing CT-oriented SW education assessments. Furthermore, the present study offers practical implications for developing fair industrial hiring tools and national-level assessment systems.

Keywords Algorithmic Problem Solving, Computational Thinking, Software Education, Item Response Theory, Test Equating

†정회원 서강대학교 교육대학원 AI융합 교육설계·경영 전공 석사과정
††정회원 서강대학교 인문대학 교육문화 연계전공 조교수 (교신저자)

논문투고 2026년 02월 26일
심사완료 2026년 04월 02일
게재확정 2026년 04월 18일
발행일자 2026년 06월 30일

* 본 논문은 2024년 서강대학교 산학협력단을 통한 산업체 연구용역의 재원으로 수행된 연구이며, 2025년 한국교육학회 연차학술대회에서 "문항반응이론 기반 알고리즘 문제 해결력 검사의 동등화 연구"의 제목으로 발표된 논문을 확장한 것임.

1. 서론

빅데이터를 비롯한 첨단 지능형 기술의 급속한 발전에 따라, 방대한 정보와 지식이 실시간으로 축적 및 확산되고 있다. 이에 단순히 정보를 습득하고 이해하는 것을 넘어, 주어진 정보를 바탕으로 복잡한 문제를 효과적으로 해결할 수 있는 역량이 부각되고 있다. 지식의 누적과 갱신이 그 어느 때보다도 가속화되고 있는 가운데, 미래 사회에 효과적으로 대응할 수 있는 핵심 역량에 대한 학술 논의가 활발히 전개되었으며, 그중 컴퓨팅 기술을 활용한 문제 해결 역량, 곧 컴퓨팅 사고력(Computational Thinking, CT)은 21세기 핵심 역량으로 주목받고 있다[1].

우리나라도 CT에 대한 높은 관심으로 인공지능 시대의 핵심적인 6가지 역량, 이른바 6C 중 하나로 CT를 상정하였으며, 학교 교육을 통해 이를 체계적으로 배양할 수 있도록 하는 교육 정책을 추진 중에 있다[2, 3]. 교육부는 2015 소프트웨어(Software, SW) 교육 운영지침[4]을 통해 CT 역량을 가진 창의 · 융합 인재 양성을 주창하며 구체적인 SW 교육 방안을 설계하였고, 2022년 개정 교육과정[5]을 통해 미래 변화에 대응할 수 있도록 디지털 소양과 정보교육의 중요성을 강조하였다. 이에 따라, 정보교과에서는 CT 역량을 핵심적인 교과 역량으로 선정하고 있는데, 컴퓨터를 활용한 문제의 발견과 분석, 체계적인 해법 설계, 그리고 문제를 해결할 수 있는 프로그램의 구현과 자동화를 강조하였고, 이에 필요한 지식과 태도를 배양할 수 있도록 성취목표 및 평가 방안을 섬세히 구조화하기도 했다[6].

CT에 대한 높은 관심과 함께 CT 교육의 필요성과 효과적인 교수법에 관한 학술적 논의가 활발히 이루어지고 있으나, CT 역량의 평가에 대한 연구는 상대적으로 부족한 상황이다[7, 8]. 구체적으로 초 · 중등교육에서는 SW 프로그램의 직접적인 개발보다 CT 기반 문제 해결력과 디지털 소양 함양에 초점을 두고 있기에[4], 블록형 코딩 등 교육용 프로그래밍 언어(Educational Programming Language, EPL)를 활용한 SW 교육평가가 활성화되었다. 반면, Java, Python, C 등의 텍스트 프로그래밍 언어(Text-based Programming Language, TPL)를 직접적으로 사용하여 자동화 SW 프로그램을 구현함으로써 문제 해결력을 평가하는 사례는 드물다. 그러나 TPL 기반 평가는 프로그래밍 언어를 습득한 이후, CT의 자동화에 해당하는 영역인 프로그램 개발을 수행하는 SW 교육이나 SW 인재를 육성 및 확보하기 위한 교육 맥락에서 중요하다.

이와 관련하여 SW 프로그래밍 역량의 평가를 위해 산업계와 고등교육 현장에서는 알고리즘 문제 해결력 검사를 적극적으로 활용하고 있다. 대표적인 사례가 바로 '알고리즘 코딩 테스트' 혹은 '코딩 테스트'로 일컬어지는 검사도구이다. 실제로 거의 모든 주요 IT 기업에서 알고리즘 문제 해결력 검사를 인재 채용 및 평가에 적극적으로 채택하고 있다[9]. 이와 유사한 검사 형식으로, 국내 최고 권위의 청소년 SW 경진대회인 한국정보올림피아드는 TPL을 활용한 알고리즘 문제 해결력을 평가하고 있으며[10], 100여 개 이상의 기업 및 대학기관에서 활용하고 있는 소프트웨어 역량 검정(TOPCIT) 또한 TPL 기반의 문제 해결력을 평가하고 있다[11]. 그러나 초 · 중등 단계에서 CT 중심의 SW 교육을 이수한 학생들이 심화 교육에 참여하거나 상급 학교에 진학한 이후 실제로 직면하게 될 이러한 검사도구, 곧 CT 역량 기반의 TPL 활용 알고리즘 문제 해결력 검사에 대한 측정학적 타당성을 체계적으로 검증한 연구는 거의 이루어지지 않은 상태이다.

이에 본 연구는 TPL 기반의 알고리즘을 설계하고 프로그램을 구현하는 SW 프로그래밍 교육, 직무 특성화 교육과 고등교육, 그리고 산업 현장에서 활발히 채택되고 있는 알고리즘 문제 해결력 검사를 보다 정교하고 타당한 CT 검사도구로 고도화하기 위한 기반을 마련하고자 한다. 이를 위해 국내 주요 대기업 A사가 실시한 알고리즘 문제 해결력 검사의 실제 응시 데이터에 대해 문항반응이론(Item Response Theory, IRT)을 적용하여 문항 모수치(item parameter) 및 피험자의 잠재 능력치(latent proficiency)를 추정하며 검사도구의 신뢰성과 타당성에 대해 검증하였다. 특히, 해당 결과를 토대로 4등급 수준의 등급 체계를 마련하고 이에 대한 타당도를 점검함으로써 알고리즘 문제 해결력 검사에서 IRT 모형의 적용과 결과 해석 및 활용의 가능성을 실증하였다. 이러한 접근은 평가가 일회성으로 종료되는 데 그치는 것이 아니라, 평가 결과를 기반으로 검사 간 연계성과 연속성을 높여, 지속 가능한 학습자 성장 지원과 안정적인 컴퓨터 기반 평가 체계를 구축하는 것에 기여할 수 있다. 따라서 본 연구는 두 가지 질문을 바탕으로 CT 평가를 위한 알고리즘 문제 해결력 검사의 타당성과 동등화 가능성 연구를 수행하였다.

첫째, 알고리즘 문제 해결력 검사 결과를 통해 어떻게 피험자의 능력을 추정할 수 있을 것인가?

둘째, 알고리즘 문제 해결력 검사의 회차 간 결과 및 점수의 비교 가능성을 고려한 등급 체계를 어떻게 마련할 수 있을 것인가?

2. 선행연구 고찰

2.1 컴퓨팅 사고력의 이론과 실제

2.1.1 컴퓨팅 사고력의 개념과 교육

CT는 복잡다단한 문제 사태를 해결하기 위하여 컴퓨터 과학의 문제 해결 원리를 기반으로 문제 상황을 분석하고, 패턴을 추출하여 효율적인 알고리즘을 설계하며, 이를 프로그래밍 언어로써 자동화하는 사고과정으로 정의된다[12]. 다만, CT가 최초로 제시되었을 당시에는 컴퓨터 과학자를 위한 사고 역량으로 제시되었지만, 최근에는 컴퓨터 과학의 영역이나 코딩 기반 문제 해결에만 한정하지 않고, 문제 분해(decomposition), 추상화(abstraction), 알고리즘 설계

(algorithm design), 디버깅(debugging), 그리고 일반화(generalization)라는 일련의 사고과정을 포괄하는 개념으로 확장되어 일상 곳곳에서 발현될 수 있는 사고력으로 이해되고 있다[13]. CT의 원리와 문제 해결 방식은 실생활에서도 적용될 수 있지만, CT는 결국 특정한 문제를 해결할 수 있도록 하는 크고 작은 SW 프로그램의 구현을 위해 필요한 절차적 사고라고 할 수 있다[14].

미래 사회에 대응하는 중핵적 역량이라는 인식 속에서 국내 SW 교육과정은 상당히 구체적으로 CT의 교육목표와 성취기준을 제시하고 있는데, 문제의 발견과 체계적 분석, 추상화를 통한 해결책의 논리적 구상, 그리고 프로그램의 설계 및 구현 과정을 통한 자동화 수행을 특히 강조하고 있다[15]. 이러한 목표에서 국내 초 · 중등 SW 교육현장에서는 학습자의 인지적 부담을 줄이고자 주로 EPL 기반 교육을 진행하고 있다. 데이터 리터러시 관련 교육 요소를 통합한 SW 교육[16]에서부터 Sage 도구 기반 수학교육[17], 그리고 Scratch 기반 과학교육[18] 등 CT 함양을 위한 여러 가지 융합형 교육이 설계되어 실제적인 문제상황에서도 CT를 폭넓게 적용할 수 있도록 하고 있다. 이러한 교과 융합형 교육설계에 대해 그 특성과 효과성에 대한 긍정적인 검증 결과 또한 메타분석 등을 통해 보고되고 있다[19, 20].

한편, TPL 기반 SW 교육은 고등교육에서 본격적으로 많이 이루어지고 있는데, 최근에는 CT 요소를 반영하여 대학교 비전공 학생들이 실제로 알고리즘을 구현하고 프로그램을 개발할 수 있도록 하거나[21], 생성형 인공지능을 활용한 프로그래밍 교육에서 CT의 세부 요소별로 CT 및 문제 해결력에 대한 학습 효과의 차이를 검증하는 연구[22]도 수행된 바 있다. 이러한 최근 SW 교육 연구에서는 프로그래밍 기술을 단순히 기술 습득의 대상으로 한정하기보다, 실제적 맥락 속에서 학습과 문제 해결의 도구로 접목 및 활용할 수 있도록 하는 통합적 교수설계의 효용성을 제시하고 있다[21, 22].

2.1.2 컴퓨팅 사고력의 평가

국제적으로 가장 공신력 높은 CT 평가 체계는 국제교육성취도평가협회(International Association for the Evaluation of Educational Achievement)에서 주관하는 국제 컴퓨터 및 정보 소양 연구(International Computer and Information Literacy Study, ICILS)라고 할 수 있다. ICILS 검사는 TPL 기반 프로그래밍 역량을 직접적으로 평가하지는 않지만, 비선형 시스템 전이 과제(non-linear systems transfer task), 시뮬레이션 과제(simulation task), EPL 중심의 블록 기반 코딩 과제(block-based coding task)를 통해 CT 기반 문제 해결력을 평가하고 있다[23]. 최근 발표된 우리나라의 ICILS 2023의 CT 검사 결과는 참여국 중 2위로 높은 성적을 보였으나, SW 교육의 강화에 따른 유의한 성취 향상이 나타나지 않았고, 국내 중학생의 정보통신기술에 관한 자아효능감은 오히려 하위권인 것으로 나타나 SW 교육의 내용과 방향성에 대한 숙의가 요청되기도 했다[23]. 이에 국가 수준의 검사도구가 부재한 상황을 고려할 때[24], SW 교육과정과 그 구성 요소의 효과성을 체계적으로 평가하고 교수-학습 상황에서의 개선 방향을 도출할 수 있는 검사도구 개발의 필요성이 제기되고 있다.

국내에서도 SW 교육에서 활용할 수 있는 CT 평가 체계와 방법에 대한 연구가 활발히 이루어지고 있다. 특히, EPL 기반의 닥터 스크레치(Dr. Scratch), CT 역량을 진단하는 베브라스 검사(Bebras Test), 그리고 지필형 검사 등 각각의 특성과 효용성이 실증된 만큼, 단일 도구에만 의존하기보다는 이들의 상호보완적 활용 가능성에 주목하고 있다[8]. 일반적으로 교육 효과성 점검이나 CT 수준 평가를 위한 도구로는 주로 지필형 검사와 자기보고식 평가[25]가 채택되고 있는데, 대부분은 설문지 형식으로 구성된 한국교육학술정보원과 한국과학창의재단에서 개발한 CT 역량 중심 SW 교육 효과성 측정 도구[26]가 그 준거로 활용되고 있다.

최근에는 검사도구 개발에 생성형 인공지능 기술을 접목시켜, 외국어 학습과 프로그래밍 언어 학습 간의 공통점과 유사성을 고려한 TOEFL(Test Of English as a Foreign Language) 형식의 지필평가 문항을 개발한 연구[27]와 프로그래밍 퍼즐 게임 기반의 라이트봇(Lightbot)을 활용한 평가 연구[28]도 진행되면서 자동 문항 생성(Automatic Item Generation)을 비롯한 자동화된 CT 평가 방법에 대한 관심 또한 높아지고 있다. 다만, 이러한 방법들은 프로그래밍 수행 능력을 EPL 기반 수행으로 한정하거나 선다형 혹은 서술형의 지필평가, 그리고 학습자의 주관적 자기보고로 한정한다는 제약이 있다.

이러한 상황에서 CT 관련 측정 및 검사 도구를 종합적으로 분석한 연구에서는 CT 측정 검사도구가 장기적으로는 컴퓨터 기반 검사(Computer-Based Test, CBT)로 발전해야 함을 제안하며, CBT 검사 환경에서 직접적으로 프로그래밍 수행 역량을 측정할 필요가 있음을 강조했다[24, 29]. 이는 TPL 기반 문제 해결력 및 프로그래밍 능력을 CBT 여건에서 측정하고 그 수행 수준을 평가하는 검사도구 연구가 부족하다는 것을 방증한다.

무엇보다 검사도구 기반의 결과를 해석하고 활용하는 실제적 측면을 고려하면, ICILS는 IRT 모형을 적용함으로써 검사도구의 타당성을 제고하고 CT 역량의 척도를 정밀하게 분석하고 있지만, 국내 연구에서는 SW 및 CT 관련 검사도구에 대해 IRT와 같은 평가학적 모형을 적용하여 검사의 동등성이나 연속적인 평가의 가능성을 고찰한 사례를 찾아보기 어렵다. 따라서 SW 프로그래밍 교육의 실제적 맥락, 즉 TPL을 활용한 알고리즘 설계 및 프로그램 구현 능력을 CBT 환경에서 타당하게 측정할 수 있는 검사도구의 개발과 타당화에 대한 체계적인 연구는 CT 평가를 위한 SW 교육 분야의 시급한 연구과제라고 할 수 있다.

2.2 알고리즘 문제 해결력 검사의 이론과 실제

2.2.1 알고리즘적 사고와 문제 해결력

CT의 하위 구인에 대한 충분한 합의가 이루어진 것은 아

니지만, 다양한 문헌을 종합했을 때, CT의 핵심요소는 추상화, 문제 분해, 알고리즘적 사고(algorithmic thinking), 자동화(automation), 그리고 일반화로 정리할 수 있다. 이 중에 '알고리즘적 사고'는 문제 해결에 필요한 구성 요소와 절차를 단계적으로 구조화하는 역량을 의미한다[30]. 알고리즘적 사고는 컴퓨터 과학과 데이터 분석을 비롯한 다양한 영역에서 그 중요성이 더욱 커지는 추세이며, ICILS 검사를 위해 구축된 CT의 평가 체계 역시 알고리즘적 사고 기반 문제 해결이 가능한 문항 개발을 강조하고 있다[31].

컴퓨터 프로그래밍의 관점에서 알고리즘이란, 문제 해결을 위해 정교하게 설계되고 구현된 일련의 절차적 규칙 또는 명령어들의 집합, 즉 프로그램을 의미한다. 이러한 맥락에서 알고리즘적 사고란, 문제 해결을 목표로 효과적인 컴퓨터 프로그램을 개발할 수 있는 사고력을 특정한다고 볼 수 있다. 실제로 TPL 교육이 본격적으로 시작되는 고등학교 정보교과에서도 '알고리즘과 프로그래밍'이 주요한 내용 체계로 제시되고 있다[6]. 세부 내용에는 문제 해결을 위한 알고리즘 설계와 프로그래밍, 알고리즘의 특징 및 효율성 분석, 자료구조에 기반한 프로그램 작성, 그리고 실생활 및 다양한 학문 분야의 문제를 해결하기 위한 협력적 프로그램의 설계, 구현, 그리고 평가가 포함되어 알고리즘적 사고를 통한 문제 해결의 중요성이 강조되고 있다[6].

이때, 선행연구의 논의에 기반하여 본 연구에서 지칭하는 '알고리즘 문제 해결력'은 주어진 문제를 해결하기 위해 컴퓨터 프로그램을 개발하는 전 과정에서 발휘되는 알고리즘적 사고 기반의 논리적 문제 해결력으로 규정할 수 있다. 이러한 개념 규정은 알고리즘과 프로그래밍 교육의 세부 영역을 구분하고, 전국 교육대학교 컴퓨터교육과 교수를 대상으로 개별 교육내용의 중요도를 조사한 연구[32]에서 도출된 결론과도 맥을 같이한다. 해당 연구에서는 논리적으로 설계된 알고리즘을 토대로 한 프로그램 개발이 SW 교육에서 핵심적으로 다루어져야 함을 강조하며, 이를 위해 알고리즘 문제 해결력 신장을 위한 체계적이고 지속적인 교육이 필요함을 강조하였다[32]. 이처럼 CT 중심 SW 교육은 단순한 절차적, 논리적 접근을 넘어, TPL을 활용하여 SW 프로그램을 개발하고, 자동화를 통해 문제를 해결하는 데까지 교수-학습과 그 평가의 활동을 다룰 수 있다.

2.2.2 알고리즘 문제 해결력 검사의 실제

'코딩 테스트'로 널리 알려진 알고리즘 문제 해결력 검사는 프로그래밍 언어 활용 능력과 알고리즘적 사고를 기반으로 주어진 문제를 정확하고 효율적으로 해결할 수 있는 알고리즘 설계 능력 및 프로그램 구현 능력을 중점적으로 평가하는 도구이다. 알고리즘 문제 해결력 검사는 국제정보올림피아드와 같은 경진대회와 국가공인 민간자격 검정 등에서도 활용되지만[10, 11], 눈에 띄게는 거의 모든 국내 기업의 SW 개발 직군의 채용 과정에서도 활용되고 있다[9].

알고리즘 문제 해결력 검사는 적은 수의 문항으로 장시간 동안 진행되는데, 이를 통해 피험자의 문제 분석 및 추상화 역량, 정확하고 효율적인 알고리즘 설계 및 최적화 능력, 그리고 자동화와 일반화 능력을 종합적으로 평가한다. 특정한 문제 상황과 제약 조건 하에서 최적의 해결 방법을 구현하도록 설계된 검사 문항에 대해, 피험자는 문제의 요구 사항을 면밀히 분석하여 최적의 알고리즘과 자료구조를 선정한 후, 이를 바탕으로 소스 코드를 작성하게 된다.

소스 코드는 온라인 판정(Online Judge) 시스템에 의해 채점되는데, 판정은 여러 개의 테스트 케이스(test case)들로 구성된 서브 태스크(sub-task)에 대해 채점하게 된다. 입출력 값의 정확성, 실행 속도와 메모리 활용 측면에서의 효율성이라는 두 가지 주요 서브 태스크를 중심으로 피험자가 구현한 프로그램의 성능을 평가한다[33, 34]. 따라서 학습자는 문제 해결 과정에서 적절한 알고리즘의 설계가 정확성뿐만 아니라 실행 속도 및 메모리 사용 효율에도 직접적인 영향을 준다는 점을 고려하며 알고리즘을 설계해야 한다[33].

알고리즘 문제 해결력 검사는 주어진 문제를 해결하는 정확하고 효율적인 알고리즘 설계 및 개발 역량을 평가하는 과정이라는 점에서 CT와 긴밀하게 연결된다고 볼 수 있다. 다만, SW 개발자의 CT 역량에 대한 평가 방법과 도구에 관한 연구는 부족한 상황이며[35], 알고리즘 문제 해결력 검사와 관련된 선행연구는 주로 온라인 판정 시스템의 구축과 이에 기반한 자동채점 알고리즘 개발의 효과성에 초점이 맞춰져 있다[34, 36, 37]. 따라서 본 연구는 TPL 활용 능력과 알고리즘적 사고력을 포함한 알고리즘 문제 해결력 검사가 타당한 검사도구로 활용될 수 있는 가능성을 검토하고자 한다.

3. 연구 방법

3.1 검사 자료의 구조와 전처리 방법

3.1.1 연구 자료

본 연구에서는 국내 대기업 A사가 최근 4년간 총 823회에 걸쳐 시행한 알고리즘 문제 해결력 검사의 응시 데이터를 분석하였다. 전체 검사에는 총 14,320명이 응시하였고, 해당 기간 동안 출제된 서로 다른 고유 문항의 수는 223개이며, 평균적으로 11회, 일부 문항은 최대 78회 반복 출제되기도 하였다. 해당 문항들은 전처리 과정을 거쳐 IRT 모형을 통한 문항 난이도 추정 데이터에 사용되었으며, 그 결과를 바탕으로 최초의 등급화 방안을 마련하는 분석자료로 사용되었다.

반복 출제된 문항들은 검사 회차 간 척도를 연결하는 가교 역할을 하게 되는데, 전체 223개 문항 가운데 207개(92.8%)가 그에 해당한다. 해당 문항들의 반복 출제 분포는 다음 Table 1과 같다. 최대 10회에 걸쳐 반복 출제된 문항은 총 152개(68.2%)이며, 11회 이상 출제된 문항은 71개(31.8%)에 해당한다. 전체 분석대상 문항의 약 93%에 해당하는 이러한 반복 출제 문항들은 서로 다른 시행 시점과 응시 집단을 가진 회차 간 검사를 하나의 공통 척도로 통합하

는 준거 문항으로 사용된다.

Table 1. Distribution of anchor items across algorithmic problem-solving assessments

Frequency of repetition (Number of assessments)	Number of items	Percentage (%)
1 – 10	152	68.2
11 – 20	39	17.5
21 – 30	16	7.18
31 – 40	6	2.69
41 – 50	5	2.24
51 – 60	4	1.79
61 – 70	0	0
71 – 80	1	0.45

3.1.2 문항 채점 방식

여타의 알고리즘 문제 해결력 검사의 채점 방식[33, 36]과 동일하게 A사의 검사도 온라인 판정 시스템을 통해 피험자가 제출한 소스 코드의 성능을 평가 및 채점하였다. 이때, 문항 출제자가 사전에 설계한 테스트 케이스로 채점을 진행했는데, 해당 검사에서는 소스 코드의 정확성(correctness)과 효율성(efficiency)을 각각의 서브 태스크로 구성하여 채점했다.

정확성 서브 태스크의 경우, 수십 개의 연산값을 각각의 테스트 케이스로 설정하고 피험자가 구현한 코드를 실행하여 출력되는 결과값들이 테스트 케이스와 일치하는지 개별적으로 채점하는 영역이다[34]. 또한, 효율성 서브 태스크의 경우, 코드의 성능을 시간 복잡도(time complexity)를 통해 평가한다. 이는 시간 복잡도를 고려하여 제한된 시간 내에 결과값이 출력되는지 여러 개의 테스트 케이스로 평가하는 항목이다[34].

본 연구에서는 서브 태스크를 구성하는 각 테스트 케이스를 정오답 여부에 따라 0점(오답)과 1점(정답)의 이진(binary) 형식으로 채점하였다. 이를 종합하여 서브 태스크 내 모든 테스트 케이스가 정답일 때는 1점, 하나라도 오답일 경우에는 0점으로 서브 태스크 단위의 이진 점수를 산출하였으며, 서브 태스크 단위의 점수를 분석자료로 활용하였다. 다시 말해, 본 연구에서는 개별 문항의 서브 태스크를 독립적인 분석 단위로 활용하였다. 이는 장시간 동안 3~4개 수준으로 매우 적은 수의 문항으로 평가를 수행하는 알고리즘 문제 해결력 검사의 시행 조건을 고려하여 정확성과 효율성 각각에 대한 정보를 최대한 활용하면서도 검사도구의 신뢰도를 높이기 위한 것이다.

3.1.3 자료 전처리 방법

본 연구에서는 앞서 제시한 채점 방식에 따라 개별 문항에 대한 서브 태스크의 개수만큼 개별 피험자의 문항 풀이 정보가 확보되도록 피험자 단위의 문항 풀이 정보를 정리하였다. 요컨대 한 개의 문항이 두 개의 서브 태스크(정확성 및 효율성)를 갖고 있다면, 해당 문항을 응시한 한 명의 피험자에 대해서는 두 개의 채점 정보를 갖게 되는 구조이다.

또한, 특정 서브 태스크에 대한 피험자의 응답이 누락(omission)된 상황에 대해 전처리를 수행했다. 국내 알고리즘 문제 해결력 검사는 속도 검사(speed test)로 진행되는 점을 감안할 때, 주어진 시간 내에 문제를 모두 풀지 못하는 것은 피험자의 역량 부족이라고 볼 수 있다. 이에 따라 검사 시간이 부족하여 문항을 확인하지 못한 경우, 문항을 풀이하고자 하였으나 코드를 제출하지 않은 경우, 그리고 모종의 사유로 해당 문항을 확인하지 않고 검사를 종료한 경우 등에 해당하는 서브 태스크들은 모두 오답(0점) 처리하였다.

한편, IRT 모형을 통한 문항 모수 및 피험자 능력치 추정의 안정성을 확보하기 위해, 개별 피험자가 응답한 서브 태스크의 개수가 3개 미만인 경우와 개별 서브 태스크에 응답한 피험자 수가 30명 이하인 경우는 분석에서 제외하였다. 첫 번째 조치에 관한 이유는, 응답 패턴이 3개 미만일 경우, IRT 모형을 통해 추정하는 피험자 능력치 값의 오차가 상당히 크게 나타나기 때문이다. 또한 개별 서브 태스크의 응답 수가 30명 이하인 경우, 전체 검사에 걸친 총 응시자의 현황 대비 해당 서브 태스크의 문항 난이도를 추정하는 데 유의한 결과를 제공하기 어렵기 때문에 분석에서 제외하였다.

3.2 IRT 적용 및 검사 등급화 방법

3.2.1 IRT 기반의 문항 난이도 추정 방법

본 연구에서는 기존 검사 데이터에 대한 문항 특성 분석과 회차 간 점수 동등화를 위한 기초자료의 구축을 목적으로 IRT 분석을 진행하였다. IRT 기반 분석은 회차 간 시행 조건이 달라지더라도(예: 다른 문제를 푸는 경우), 결과 점수를 비교 가능하게 하는 이론적 기반을 제공한다[38]. 본 연구는 문항의 특성을 분석하기 위해 1-모수 Rasch 모형[39]을 채택하였는데, Rasch 모형은 데이터 요구 조건과 통계적 가정이 최소화된 형태로, 현재까지 축적된 A사의 검사 데이터를 보다 안정적으로 분석할 수 있는 장점이 있기 때문이다. 종전의 검사 설계가 다회차 검사 점수의 비교 가능성을 고려하여 설계되지 않았기 때문에, 통계적 가정과 조건이 비교적 간소한 Rasch 모형을 활용하기로 하였다. 탐색적 단계에서는 2-모수 모형의 적용도 시도하였으나, 기존 검사 설계의 구조적 제한으로 인해 문항 변별도 추정에서 오차가 크게 나타났다. 이에 따라 최종 분석에서는 문항 난이도(item difficulty) 모수만을 추정하는 Rasch 모형을 채택하였다. 분석 모형은 수식 (1)과 같으며, 1-모수 모형은 피험자 i의 잠재 능력치(θ_i)와 문항 j의 문항 난이도(b_j) 및 이에 따른 정답 확률(X_{ij})을 정의한다.

$$P(X_{ij}=1|\theta_i)=\frac{\exp(\theta_i-b_j)}{1+\exp(\theta_i-b_j)} \quad (1)$$

Rasch 모형을 통한 IRT 분석의 경우, 각 피험자의 개별 서브 태스크에 대한 반응(response), 곧 서브 태스크 단위의 정오답 여부를 분석 단위로 하게 된다. 이를 통해 문항의 난이도와 피험자의 잠재 능력치를 추정한다. 문항 난이도 추정의 경우, 한계최대우도추정(Marginal Maximum Likelihood Estimation) 방법, 피험자 능력 추정의 경우, 가중우도추정법(Weighted Likelihood Estimation, WLE)을 활용하였다.

이때, 문항 난이도와 피험자 능력치는 모두 로짓(logit) 척도 상의 값을 갖게 되는데, 대부분의 교육 연구에서 IRT로 피험자의 능력치를 추정할 때는 ±3에서 ±4 로짓 범위에 모수들이 분포하는 경우가 많다. 다만, 이러한 값들은 직관적으로 이해되는 값은 아니기에 실제 결과 점수를 표현하거나 등급화를 구성하는 단계에서는 검사 실무진 및 피험자에게 조금 더 친숙한 형태로 이해될 수 있도록 그 결과를 선형변환 할 수 있다.

3.2.2 검사 등급화 방법

등급화 체계의 마련을 위해 본 연구에서는 여러 회차에 걸쳐 반복적으로 출제된 공통 문항을 가교(anchor)로 두고, 회차별로 상이한 검사 데이터를 동시에 분석함으로써 척도를 연결(linking)하였다. 이는 모든 검사 회차의 데이터를 통합적으로 분석하여 전체 문항 모수치를 한 번에 추정하고 그에 따라 피험자의 능력치를 추정하는 방식으로 동시모수추정(concurrent calibration)이라고 불린다[38]. 동시추정법은 회차 간 동등화 과정에서 발생할 수 있는 오류의 누적을 방지하고, 다양한 회차에 출제된 모든 문항을 동일한 척도상에서 비교할 수 있도록 함으로써 일반적으로 가장 효율적이고 정확한 방법으로 평가된다[38, 40].

동시추정법에 따라 Rasch 모형을 적합하게 되면, 문항 난이도와 WLE 기반으로 추정되는 피험자 능력치는 기본적으로 -∞에서 +∞의 값을 갖는 로짓 척도 상에서 추정된다. 이때, 능력치를 음수로 보고하게 되면 실제 검사 결과를 점수로 설명하는 데 해석상의 어려움이 있어, 본 연구에서는 로짓 척도로 산출된 피험자의 능력치 점수를 표준점수(standard score) 체계의 척도로 선형변환(linear transformation)하였다.

선형변환된 척도 상에서 본 연구는 네 개 수준으로 점수 구간을 분리하여 등급화를 진행하고, 해당 등급화 구조가 타당한지, 후속 회차에도 안정적으로 적용 가능한지 검증하기 위해 검사 전문가와 검사 결과의 해석 타당성 및 일반화 가능성을 논의하였다.

정리하면, 본 연구에서 수행한 방법은 다음 Fig. 1과 같이 요약할 수 있다. Rasch 모형 기반 동시모수추정으로 전체 검사 회차의 문항 난이도를 추정하고, WLE 방법을 기반으로 추정한 피험자의 잠재 능력치를 선형변환하여 다양한 회차의 검사 결과를 비교 가능하고 이해하기 쉬운 하나의 척도 점수로 산출한다. 이에 따라 최초의 등급화 구조를 마련한 뒤, 등급화 체계의 타당성을 검사 전문가와의 논의를 통해 입증한다.

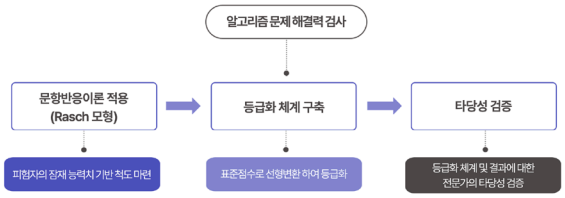

Figure 1. Framework of research procedure and method

4. 연구 결과

4.1 IRT 기반 문항 난이도 및 피험자 능력 추정

4.1.1 최종 연구 자료

최초의 자료에는 총 823회차에 걸친 검사에서 223개 문항에 대해 총 437개의 서브 태스크가 포함되어 있었다. 다만, 모든 회차의 시험이 동등화 과정을 염두하고 설계된 것이 아니기 때문에 문항과 서브 태스크별 동등화 가능성과 피험자의 응답 수 등을 고려하여 전처리 과정을 진행하였고, 최종 266개의 서브 태스크로 분석 단위가 조정되었다. 그 중에서도 12개 서브 태스크는 모든 응시자가 해당 문항에 정답을 맞히지 못한 경우 또는 모든 응시자가 정답을 맞힌 경우에 해당하는데, 이러한 경우는 문항 난이도를 안정적으로 추정할 수 없으므로 분석에서 제외하였다. 최종적으로 등급화 및 동등화 분석에 포함된 서브 태스크는 254개, 피험자는 총 13,931명으로, 본 연구에서 실시한 IRT 기반 분석을 통해 서로 다른 회차에서 시행된 총 254개의 서브 태스크에 대해 동일한 척도 상에서 문항 난이도를 추정하여, 상호 비교가 가능하도록 분석하였다.

4.1.2 IRT 모형의 적합도 및 문항 난이도 추정

Rasch 모형 분석 결과, IRT 기반의 주변 신뢰도(marginal reliability)는 약 0.89로 매우 높은 수준으로 나타났는데, 이는 총 254개의 서브 태스크로 구성된 알고리즘 문제 해결력 검사도구가 신뢰로운 검사도구로 피험자의 잠재능력을 일관되게 측정할 수 있음을 보인다. 다만, 피험자 개인이 총 254개의 서브 태스크에 모두 응답하는 상황은 비현실적이기 때문에, 피험자 개개인에게 점수 결과를 보고할 때에는 능력 추정에 대한 측정오차(measurement error)를 고려할 필요가 있다.

Rasch 모형을 최종 선택하기 이전, 탐색적으로 2-모수 모형과의 비교 분석을 수행하였다. 분석 결과, 2-모수 모형에서 추정된 문항 변별도 모수의 분포는 평균 10.714, 최소값 -0.487에서 최대값 54.627까지로 나타났으며, 문항 난

이도는 평균 0.133, 최소 -22.822에서 최대 33.973으로 매우 넓은 범위를 나타냈다.

IRT를 실제 데이터에 적용하는 경우, 통상적으로 로짓 스케일 상에서 -6에서 6사이의 난이도 추정값이 기대된다는 점을 고려할 때, 이러한 결과는 2-모수 모형의 추정치가 신뢰롭지 못하고 매우 불안정함을 시사한다. 따라서 본 연구에서는 충분한 신뢰성을 보이며 비교적 안정적인 추정 적합성을 나타내는 1-모수 Rasch 모형을 최종 모형으로 선택하였다. 이때, Rasch 모형을 통해 추정한 알고리즘 문제 해결력 검사의 문항 난이도는 최소 −6.77에서 최대 7.46, 평균 0.49, 표준편차 2.85로 상대적으로 적정한 범위 내에서 안정적으로 추정되었음을 확인하였다.

4.1.3 IRT 기반 피험자 능력치 추정 및 척도화

WLE를 통해 추정한 피험자의 잠재 능력치 추정값은 로짓 척도 상에서 최소 -9.21에서 최대 9.22의 범위를 보였고, 평균은 0.08, 중앙값은 −0.01, 표준오차는 평균 1.311 수준으로 나타났다.

본 연구에서는 이를 보다 실용적인 해석이 가능한 알고리즘 문제 해결력 검사의 전용 표준점수 기반 척도로 선형변환하였다. 선형변환 모형은 다음 수식 (2)와 같고, WLE 추정치에 대해 분산을 조정하는 척도 계수 A와 위치를 조정하는 이동 계수 B를 연산한다. 즉, A는 WLE 분포의 표준편차를 목표로 하는 표준점수의 표준편차에 맞추기 위한 것이며, B는 선형변환 후 평균이 목표 평균에 일치하도록 조정하는 계수이다.

$$Z = A^{*} WLE + B \qquad (2)$$

WLE 기반의 표준점수 선형변환은 검사의 특성 및 활용 목적에 따라 다양한 형태로 설계될 수 있으므로, 기존 검사 설계와 분석 단계에서 발생할 수 있는 추정 오차를 반영하여 적절한 점수 범위를 탐색하였다. 이 과정에서 검사 실무진과 협의를 통해 최종 변환 계수와 기준을 확정하였다. 이어서 능력치 분포의 평균과 표준편차를 기준으로 정규화를 수행한 뒤, 이를 바탕으로 도출된 평균 15, 표준편차 3의 값으로 목표 표준점수 체계를 설정하였다. 이후 수식 (2)의 변환식에 해당 값들을 적용하여 WLE 능력치를 목표 표준점수 체계로 변환하였다. 선형변환을 위한 WLE 기반 추정치 및 표준점수로의 변환 이후 분포는 다음 Fig. 2와 같다

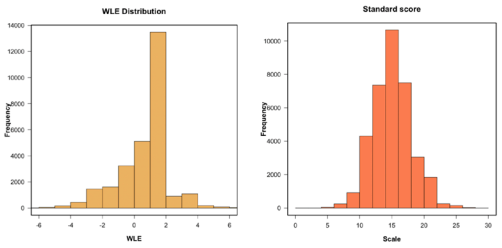

Figure 2. Distributions of WLE scores and standard scores

히스토그램을 통해 확인할 수 있듯이, 응시자의 평균 능력점수는 0점에서 30점까지 정규분포 형태로 고르게 분포하였다. WLE 기반 피험자 능력 추정 및 표준점수 변환 결과에 대해 검사 실무진은 기존 고전검사이론의 단순 정답률 개념으로는 한계가 있었던 분석의 비교 불가능성에 대한 아쉬움을 보완할 수 있었고, 결과가 납득 가능한 수준에서 타당하다고 평가하였다.

4.2 검사 등급화

마지막으로 추정된 잠재 능력치를 선형변환한 표준점수의 체계를 토대로, 전체 피험자의 검사 결과를 보고하기 위한 등급화 작업을 수행하였다. 등급화를 통해 검사의 결과를 해석하고 활용하는 데 있어서 용이한 평가기준을 수립할 수 있게 되며, 이는 후속적으로 시행되는 여러 검사 결과의 타당한 해석 및 검사 동등화의 가능성 여부를 검증하는 데에도 활용된다.

등급화 기준의 설정을 위하여 검사 실무진 및 문항 개발을 담당한 내용 전문가와 협의를 거쳤으며, 최종적으로 Rasch 모형을 적용하여 추정된 문항 난이도의 평균 경계값을 기반으로[41], 다음 Table 2와 같이 표준점수 상의 등급별 구간 점수를 산출하였다. 피험자의 능력 수준은 미인증 구간인 Grade 0에서부터 최고 등급인 Grade 3까지 총 4수준으로 구분하였다.

Table 2. Grading system based on the standard score

Grade	Score range	Population	Percentage (%)
Grade 3	24–30	85	0.61
Grade 2	18–23	2,258	16.21
Grade 1	13–17	8,186	58.76
Grade 0	0–12	3,402	24.42

각 피험자의 능력 점수와 등급화 기준을 고려하여 총 13,931명의 응시 데이터를 분석한 결과, Grade 3에 해당하는 응시자는 전체의 1% 미만이었고, Grade 2는 약 16%, Grade 1은 약 60%, 그리고 Grade 0에 해당하는 미인증 영역의 피험자는 약 24%인 것으로 나타났다. 검사 실무진의 전문적 판단에 근거하여 도출된 표준점수 및 등급화 결과의 타당성을 검토한 결과, 등급화 기준이 실제 응시자 집단에

적합하게 적용된 것으로 판정되었으며 등급 간 변별력이 적절하게 확보된 등급 체계임을 확인하였다.

등급 체계에 관한 구체적인 역량 명세는 선행연구의 내용을 종합적으로 검토하며 검사 실무진과의 논의를 통해 다음 Table 3의 내용과 같이 설정하였다. 이러한 구체적인 등급 체계는 후속적으로 시행되는 알고리즘 문제 해결력 검사 결과의 분석 및 해석, 그리고 피험자의 역량 판정의 준거로 기능하게 된다.

Table 3. Descriptions of competency by grade levels

Grade	Descriptions of competency
Grade 3	Able to utilize algorithms and data structures to develop programs that solve problems by considering both accuracy and efficiency.
Grade 2	Able to develop programs that solve problems with an emphasis on accuracy.
Grade 1	Able to analyze and solve basic problems based on programming language proficiency.
Grade 0	Unclassified (Not certified).

등급별 상세 역량에 관한 내용은 Association for Computing Machinery (ACM), IEEE Computer Society, 그리고 Association for the Advancement of Artificial Intelligence (AAAI)가 공동으로 연구 및 개발한 국제 컴퓨터과학 교육과정(CS2023)의 역량(skill) 체계[42]를 중심으로 구성되었다. CS2023은 개정된 Bloom의 분류체계에 기반하여 Explain – Apply – Evaluate – Develop의 총 4단계로 구성되는데, 여기서 Explain 등급(level)은 다른 모든 수준의 전제 조건으로 활용된다[42].

검사 실무진 및 전문가와의 논의를 통해 도출한 Table 3의 역량 중 Grade 1~3은 각각 CS2023의 역량 2~4단계(Apply, Evaluate, Develop)에 상응한다. 구체적으로, Grade 1은 문제의 진단과 해결을 위해 기초적인 프로그래밍 언어에 대한 이해를 적용(apply)하는 수준이며, Grade 2는 문제 해결을 위해 다양한 알고리즘을 비교하여 프로그램의 정확성을 평가(evaluate)할 수 있는 수준이다. Grade 3에서 알고리즘의 효율성까지 고려할 수 있는 수준은 CS2023의 Develop 단계에서 제시한 실제적인 프로그램의 설계와 통합, 배포, 그리고 개선을 이뤄내는 과정 속에서 시간 및 공간 복잡도, 그리고 자원 소요를 종합적으로 고려하여 최적의 해결 방안을 도출하는 높은 수준으로 구체화된다.

종합하면, 본 연구는 CT에 대한 측정 도구의 타당화 연구를 수행함으로써 새로운 회차의 검사 상황에서도 회차 간의 연계성을 높이는 방향으로 증거 기반의 알고리즘 문제 해결력 검사를 운영 및 활용할 수 있도록 하는 타당화 및 동등화 방법의 토대를 마련하였다.

5. 결론 및 논의

본 연구는 CT 평가를 위한 SW 교육연구에서 그동안 상대적으로 주목받지 못했던 알고리즘 문제 해결력 검사에 대해 IRT 적용 및 검사 동등화 가능성을 경험적으로 실증하였다. 이를 통해 CT의 중요한 하위 요인으로 다뤄지는 알고리즘적 사고에 기반한 자동화 및 프로그램 구현에 특화된 TPL 기반 검사도구의 타당성을 검증하였다. 첫 번째 연구 질문과 관련하여, 본 연구는 서브 태스크 단위로 문항의 분석 단위를 설정한 뒤, IRT 기반 동시추정방법을 활용하여 문항 난이도와 피험자 능력치를 추정하였다. 특히, IRT 분석에서는 능력치가 로짓 척도로 산출되므로, 직관적 해석과 실무적 활용을 고려하여 WLE 추정치를 선형변환함으로써 표준점수 체계를 마련하는 방법의 효과성 또한 확인하였다. 이로써 분석 모형의 타당성을 확보하는 동시에, 피험자의 능력치를 안정적으로 추정하고 검사 결과를 신뢰롭고 용이하게 해석할 수 있는 기반을 마련하였다.

한편, 두 번째 연구 질문과 관련하여, 본 연구는 선형변환된 표준점수 체계에서 피험자의 잠재 능력치를 토대로 검사 실무진과 문항 개발진 등 내용 전문가와의 논의, 그리고 CS2023을 중심으로 한 선행연구 검토를 거쳐 등급화 체계를 수립하였다. 이는 실무적으로 적용 가능한 수준의 타당한 체계임을 확인하였으며, 이를 통해 알고리즘 문제 해결력 검사에서 IRT의 적용 가능성은 물론, 향후 검사의 고도화를 위한 방략으로써 검사 동등화의 가능성 또한 확인하였다.

본 연구는 그간 신뢰성에 대한 실증적 검증이나 경험적 타당화 과정 없이 광범위하게 활용되어 온 알고리즘 문제해결력 검사에 대해 IRT 모형을 적용하고 검사 동등화 가능성을 탐색함으로써, 해당 분야의 연구적 공백을 보완하는 CT 교육평가 체계의 고도화를 위한 연구의 일환으로써 의의를 지닌다. 특히, 비교적 단순한 1-모수 Rasch 모형을 기반으로 한 WLE 추정과 동시추정법만으로도 일정 수준 이상의 신뢰성과 안정성을 확보할 수 있음을 실증적으로 제시하였다. 이는 EPL 기반 평가, 자기보고식 설문, 또는 전통적 지필평가와는 달리, CT를 위한 TPL 기반의 알고리즘 문제 해결력을 요구하는 문항 유형과 CBT 기반 검사 환경에서도 경험적인 타당화가 가능함을 보여주는 연구 성과라고 할 수 있다.

그러나 본 연구에는 몇 가지 한계가 있다. 우선, 역량 명세의 기반이 충분하지 않은 상태에서 문항이 개발되었으므로, 알고리즘 문제 해결력의 측정 구인으로 설정된 정확성과 효율성에 대한 타당성 검토가 필요하다. 이에 후속 연구에서는 구인 타당성을 체계적으로 검토하고, 궁극적으로 CT 역량과의 연계성을 명확히 해야 할 것이다. 또한, 본 연구는 체계적 검사 설계 없이 시행된 기존 데이터를 활용하였기 때문에, 능력 추정과 등급화 결과의 일반화에는 신중한 해석이 요구된다. 체계적 검사 설계는 향후 2-모수 IRT 모형의 활용과 같이 문항의 고유한 속성인 변별도 등의 차이를 보다 현실적으로 반영하기 위한 후속 연구에서도 중요한 과제이다. 한편, 타당성을 검토한 등급화 체계에 대해서는 후속적으로 몇 차례의 파일럿 테스트를 통해 실제 검사 동등화 자체의 타당성을 검증하는 과정이 필요하다. 아울러, 일련

의 과정을 통해 추정된 문항 모수와 피험자의 잠재 능력치에 대해서는 차별기능문항(differential item functioning, DIF) 분석 등 피험자 수준의 보다 정교한 검증이 이루어질 필요가 있으며, 검사 시점의 변화에 따른 문항 불변성과 안정성을 엄밀하게 검증함으로써, 회차 간 연계성과 비교가능성을 확보한 검사 도구로 지속 발전시킬 필요가 있다.

결론적으로 본 연구는 알고리즘 문제 해결력 검사에 측정학적 모형을 적용하여 증거 기반(evidence-based) 평가 방법을 구현함으로써 평가의 신뢰성과 타당성을 제고할 수 있는 경험적 근거를 마련하였다. 이러한 성과는 향후 CT 기반 SW 교육 현장에서 학습자의 TPL 기반 문제 해결 수행력을 진단 및 평가하는 데 기여할 뿐 아니라, 학습 성취를 종합적으로 판단하는 총괄적 평가에까지 확장될 수 있다. 더 나아가 본 연구는 회차 간 결과의 동등성을 확보하는 방법에 대한 시사점을 제공함으로써, 국가 수준의 검사 도구 개발은 물론, 산업계에서 신뢰성 높은 채용 도구와 공정한 자격 인증 체계 구축에도 기여할 수 있을 것으로 사료된다.

참고문헌

[1] OECD (2023), PISA 2022 *Assessment and Analytical Framework*, PISA, OECD Publishing, Paris, https://doi.org/10.1787/dfe0bf9c-en.

[2] Shin, H. J. (2024). The meeting of AI technology and education in the age of AI. *In Statistics Korea* (Ed.), Social trends in Korea 2024 (pp. 132–141). Statistics Korea.

[3] Korea Education and Research Information Service. (2017). *A study on applying digital literacy to the curriculum* (Research Material RM 2017-4). KERIS.

[4] Ministry of Education. (2015). *2015 guidelines for software education*. Ministry of Education.

[5] Ministry of Education. (2022a). *2022 revised national curriculum: General guidelines* (Notice No. 2022-33 [Book 1]). Ministry of Education.

[6] Ministry of Education. (2022b). *High school curriculum* (Notice No. 2022-33 [Book 4]). Ministry of Education.

[7] Kim, M., Yoo, G., & Kim, H. (2017). Development of a scoring rubric based on computational thinking for evaluating students' computational artifacts in programming course. *The Journal of Korean Association of Computer Education, 20*(2), 1-11.

[8] Choi, H., & Kim, M. (2017). A complementary approach of three methods for computational thinking assessment. *Journal of the Korean Association of Information Education, 21*(6), 639-646.

[9] Shin, S., Yoon, B., & Jin, H. (2025). *Prospects for changes in software developer recruitment and the impact of generative artificial intelligence*. (Research Report RE 192). Software Policy & Research Institute. https://spri.kr/posts/view/23870

[10] Gil, H. (2017). Introduction and trends of the Korea Olympiad in Informatics (KOI) competition. *Monthly SW-Oriented Society, Software Policy & Research Institute*. https://spri.kr/posts/view/21760

[11] Lee, H., & Seo, Y. (2018). An analysis of the proliferation case of TOPCIT(test of practical competency in ICT) and policy implications. *Journal of Digital Convergence, 16*(5), 1-12.

[12] Wing, J. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33-35.

[13] Shute, V., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review, 22*, 142-158.

[14] Lee, S. (2018). Effect of software education based on computational thinking on the learning interest and career recognition of elementary school students. *The Journal of Elementary Education Studies, 25*(1), 49-72.

[15] Lee, J., & Choe, H. (2025). Analysis of competency reflection in the 2022 revised informatics curriculum based on the OECD education 2030 framework. *The Journal of Korean Association of Computer Education, 28*(1), 1-11.

[16] Moon, H., & Lee, Y. (2024). The effects of data literacy-based sw convergence education on Elementary school student's computational thinking. *The Journal of Korean Association of Computer Education, 27*(4), 247-259.

[17] Park, K. E., & Lee, S. G. (2015). Improving computational thinking abilities through the teaching of mathematics with Sage. *Communications of Mathematical Education, 29*(1), 19-33.

[18] Lee, H., & Lee, T. (2015). Development of science subject program based on programming learning to improve computational thinking ability in middle school. *Journal of The Korea Society of Computer and Information, 20*(12), 181-188.

[19] Jeon, S., & Lee, Y. (2017). Meta-analysis of the programming learning effectiveness depending on the teaching and learning method. *Journal of The Korea Society of Computer and Information, 22*(11), 125-133.

[20] Na, C., Joo, H., Lee, J., & Kim, D. (2018). Inducing computational thinking in Korean SW education: Synthesizings standardized mean changes through meta-analysis. *Journal of Educational Technology, 34*(3), 775-815.

[21] Lee, M. (2022). Development of python education program with computational thinking. *Journal of The Korea Society of Computer and Information, 27*(11), 315-323.

[22] Jang, E., & Oh, K. (2025). Analysis of the impact of generative AI utilization on computational thinking

and problem-solving ability. *The Journal of Korean Association of Computer Education, 28*(7), 83-98.
[23] Seo, M., Shin, H., Lee, D., Han, S., Han, J., Cheon, H., Yang, H., & Kim, S. (2024). *International computer and information literacy study: Analysis of ICILS 2023 results* (Research Report RRE 2024-2). Korea Institute for Curriculum and Evaluation.
[24] Kim, M., Lee, W., & Kim, J. (2017). Presenting the development direction through the analysis of tool used to measure computational thinking. *The Journal of Korean Association of Computer Education, 20*(6), 17-25.
[25] Kim, S., Kim, Y., Jo, A., & Lee, M. (2019). Development of a tool for computational thinking assessment in problem-solving programming education: Paper type inspection and self-report questionnaire. *The Journal of Korean Association of Computer Education, 22*(3), 89-99.
[26] Lee, H., Lee, H., Sung, J., Jeong, S., Kim, S., Kim, S., Kim, J., Park, J., & Lee, B. (2016). *A study on surveying the actual conditions and evaluating the effectiveness of SW education in elementary and secondary schools* (Final report, Report No. TRKO201600014678). Korea Foundation for the Advancement of Science and Creativity.
[27] Park, C., & Hyun, J. (2024). Development of evaluation question types and difficulty level measurement methods based on programming elements for the programming area of high school informatics subject. *The Journal of Korean Association of Computer Education, 27*(7), 25-34.
[28] Lee, Y. (2020). A study on the diagnosis method of knowledge information in computational thinking using LightBot. *Journal of the Korea Convergence Society, 11*(8), 33–38.
[29] Kim, J. (2017). Computational thinking-based problem-solving skills. *Korea information processing society review, 24*(2), 13-21.
[30] Cansu, F., & Cansu, S. (2019). An overview of computational thinking. International Journal of Computer Science Education in Schools, 3(1), 17-31.
[31] Fraillon, J. (2024). National and school contexts for CIL and CT education. In Fraillon, J. (Ed.). *An international perspective on digital literacy: Results from ICILS 2023* (pp. 27-91). International Association for the Evaluation of Educational Achievement (IEA).
[32] Jeong, Y. (2016). Needs analysis of software education curriculum at national universities of education for the 2015 revised national curriculum. *Journal of The Korea Association of Information Education, 20*(1), 83-92.
[33] Mun, H., Kim, S., Kim, J., & Lee, Y. (2021). A source-code similarity-based automatic tutoring method for online coding test service. *Journal of KIISE, 48*(9), 1044-1051.
[34] Kurnia, A., Lim, A., & Cheang, B. (2001). Online judge. *Computers & Education, 36*(4), 299-315.
[35] Park, S., & Ahn, S. (2016). An exploratory study of the competence of computational thinking: For software developer. *The Journal of Korean Association of Computer Education, 19*(5), 41-53.
[36] Hyun, K., Choi, W., & Chung, J. (2022). Learning source code context with feature-wise linear modulation to support online judge system. *The Transactions of the Korea Information Processing Society, 11*(11), 473-478.
[37] Kim, Y., Cheon, J., & Woo, W. (2024). Implementation of a coding style checking system in an online judge system. *The Transactions of the Korea Information Processing Society, 13*(9), 437-443.
[38] von Davier, M., & von Davier, A. A. (2007). A unified approach to IRT scale linking and scale transformations. *Methodology, 3*(3), 115-124.
[39] Rasch, G. (1960). *Probabilistic models for some intelligence and attainment tests*. Copenhagen: Danish Institute for Educational Research.
[40] Kim, S., & Kolen, M. (2007). Effects on scale linking of different definitions of criterion functions for the irt characteristic curve methods. *Journal of Educational and Behavioral Statistics, 32*(4), 371-397.
[41] Wilson, M. (2023). *Constructing measures: An item response modeling approach*. Routledge.
[42] Kumar, A., Raj, R., Aly, S., Anderson, M., Becker, B., Blumenthal, R., Eaton, E., Epstein, S., Goldweber, M., Jalote, P., Lea, D., Oudshoorn, M., Pias, M., Reiser, S., Servin, C., Simha, R., Winters, T., & Xiang, Q. (2024). *Computer Science Curricula 2023*. Association for Computing Machinery. https://doi.org/10.1145/3664191

김의겸

· 2023년 서강대학교 미국문화 전공(문학사), 빅데이터 사이언스 전공(공학사), 교육문화 전공(교육학사)
· 2024년~현재 서강대학교 교육대학원 석사과정

관심분야 : 인공지능 기반 교육측정 및 평가, 교육 데이터 분석, 인공지능 심리측정학

egkim@sogang.ac.kr

신효정

· 2005년 서울대학교 교육학과(문학사)
· 2007년 서울대학교 교육학과(교육학석사)
· 2015년 University of California at Berkeley 교육학과(박사)
· 2022년~현재 서강대학교 인문대학 교육문화 연계전공 조교수

관심분야 : AI 기술을 활용한 교육 측정 평가, 교육 분야 빅데이터, 국제학업성취도평가, 교육과정 평가

hshinedu@sogang.ac.kr