



# 프로그래밍 학습 과제의 인지적 역량 요구 수준 분석 프레임워크의 개발 및 적용\*

## Development and Application of a Framework for Analyzing Cognitive Competence Demand Levels in Programming Learning Tasks

이다겸<sup>†</sup> · 김윤정<sup>††</sup> · 이영준<sup>†††</sup>  
 Dageom Lee<sup>†</sup> · Yunjeong Kim<sup>††</sup> · Youngjun Lee<sup>†††</sup>

### 요약

정보 교과과는 학생들의 컴퓨팅 사고력(Computational Thinking)으로 대표되는 인지적 역량과 문제 해결 역량을 기르는 것을 목표로 하며, 특히 프로그래밍 교육은 컴퓨팅 사고력을 기르는 데 필수적인 영역이다. 그러나 프로그래밍 학습 과제가 컴퓨팅 사고력을 향상하는 데 적합한 수준과 요소를 지니고 있는지 체계적으로 분석하고 개발하는 연구는 부족하다. 본 연구는 프로그래밍 과제의 인지적 수준 및 요소를 분석하기 위해 Stein 외(2009)의 프레임워크를 수정하여 프로그래밍 과제의 인지적 수준 분석 프레임워크를 개발하였다. 과제 수준은 기억하기 과제(M), 연계성 없는 절차 과제(PNC), 연계성 있는 절차 과제(PWC), 프로그래밍 행하기 과제(DP) 총 4수준으로 분류되며 11명의 정보 교육 전문가를 통해 내용 타당도 검증을 수행하였다. 검증된 프레임워크로 2022 개정 중학교 정보 교과서 10종에서 제공하는 프로그래밍 과제를 분석하였고 10종 교과서 중 5종에 제시된 과제의 수준이 하나의 수준에 치우쳐 제시되었다. 또한, 성취기준별 과제 수는 최대 3.5개로, 프레임워크에서 제시한 네 가지 수준을 달성하기에 부족하였다. 교과서별 과제 수준의 흐름을 분석한 결과 일부 교과서는 높은 수준의 과제에서 인지적 격하 현상이 일어나 과제 간 수준이 크게 변동하거나 낮은 수준에 치중되어 있었다. 학습자의 컴퓨팅 사고력을 효과적으로 향상하기 위해 성취기준 수준별 과제가 1개 이상은 제시되어야 하며, 과제 구성의 일관성과 체계성을 위해 인지적 수준이 점진적으로 증가하도록 구성하고 인지적 격하 현상을 막기 위한 교수법 연구 및 교사전문성 향상 방안이 필요하다.

**주제어** 프로그래밍 학습 과제 분석, 프로그래밍 과제 인지적 요구 수준 분석 프레임워크, 2022 개정 중학교 프로그래밍 과제 분석

### ABSTRACT

The informatics curriculum aims to develop students' computational thinking (CT) and problem-solving skills, with programming education playing a crucial role. However, research on systematically analyzing and developing programming tasks suitable for enhancing CT remains limited. This study modifies Stein et al.'s (2009) framework to develop a Cognitive Level Analysis Framework for Programming Tasks, categorizing tasks into four levels: Memory Tasks (M), Procedures Tasks without Connections (PNC), Procedures Tasks with Connections (PWC), and Doing Programming Tasks (DP). Content validity was verified by 11 experts. Using this framework, programming tasks from ten middle school informatics textbooks (2022 revision) were analyzed. Results show that five textbooks presented tasks skewed toward certain level, and the number of tasks per achievement standard was insufficient to cover all four levels. Additionally, some textbooks exhibited cognitive demand decline, with abrupt variations in task levels or a focus on lower levels. To enhance students' CT effectively, each achievement standard should include at least one task per level, with tasks structured to gradually increase in cognitive complexity. Further research on teaching methods and teacher training is needed to prevent cognitive demand decline in programming tasks.

**Keywords** programming learning tasks analysis, cognitive level of programming learning tasks analysis framework, 2022 revised curriculum, middle school programming tasks analysis

- †정회원 한국교원대학교 대학원 컴퓨터교육과 박사과정
- ††정회원 광주교육정책연구소 교사
- †††중신회원 한국교원대학교 컴퓨터교육과 교수(교신 저자)
- 논문투고 2025년 01월 09일
- 심사완료 2025년 03월 04일
- 게재확정 2025년 03월 05일
- 발행일자 2025년 03월 19일

\* 본 논문은 2024년 광주광역시교육청교육연구정보원 교육정책연구부 광주교육정책연구소에서 수행된 연구 보고서에 기반함(광주교육연구정보원2025-10).

## 1. 서론

4차 산업혁명의 가속화와 인공지능 기술 발전에 따른 급격한 디지털 환경의 변화로 인해 미래를 예측하기 어려운 디지털 대전환의 시대가 도래하였다. 이에 따라 교육부는 미래 사회를 살아갈 학생들이 변화에 발 빠르게 대응하는 주도적인 사람으로 성장하는 데 중점을 두고 2022 개정 교육과정을 추진하였다[1]. 정보 교과는 컴퓨팅 사고력(Computational Thinking, CT)을 기반으로 인공지능을 포함한 컴퓨팅 기술을 활용하여 미래 사회에서 다양한 분야의 문제를 발견하고 해결할 수 있는 기초적인 능력을 함양하는 데 중점을 둔다[2]. 컴퓨팅 사고력은 Wing(2006)에 의해 본격적으로 논의된 역량으로 정보 교과 고유의 인지적 사고 능력을 말한다[3]. 정보 교과 교육과정에서는 이 능력을 '컴퓨팅 시스템을 활용하여 실생활과 다양한 학문 분야의 문제를 이해하고 창의적으로 해법을 구현하여 적용할 수 있는 능력'으로 정의한다[6]. 이러한 역량을 통해 학생들은 복잡한 문제의 본질을 파악하고 자동화된 해결책을 구현하는 능력을 갖추게 된다. 따라서 컴퓨팅 사고력은 디지털 시대를 살아가는 모든 학생에게 필수적인 인지적 역량이다[7].

정보 교과와 프로그래밍 영역은 컴퓨팅 사고력을 증진하는 데 핵심적인 역할을 한다[9-11]. 프로그래밍은 단순히 코드를 작성하는 기술적인 활동에 그치는 것이 아니라, 문제를 분석하고 이를 해결하기 위해 효율적이고 창의적인 알고리즘을 설계하는 과정에서 논리적 사고력과 창의적 문제 해결 능력이 향상된다[9]. 교과서는 교육과정을 가장 잘 반영한 교수-학습 자료로, 교사는 교과서에 기반하여 학생들에게 무엇을 어떻게 가르칠 것인지 전반적인 계획을 세우며, 학습 수행을 확인하고 평가하기 위한 과제 선택도 교과서를 기반으로 이뤄진다[8]. 정보 교과 교과서도 교육과정의 핵심 매개체로서 교과서 내 학습 과제를 통해 학생들은 문제를 분석하고 창의적으로 해결하는 컴퓨팅 사고력을 향상시킬 수 있다.

따라서 정보 교과서에 제시된 프로그래밍 과제는 다양한 인지적 활동, 적절한 난이도, 그리고 단계적인 과제 흐름을 갖추어야 학생들의 컴퓨팅 사고력을 효과적으로 향상할 수 있다. 그러므로 정보 교과서의 프로그래밍 학습 과제가 학생들의 컴퓨팅 사고력을 효과적으로 향상할 수 있도록 설계되어 있는지 체계적으로 분석할 필요가 있다. 그러나 과제의 인지적 요구도와 요소를 체계적으로 분석할 프레임워크가 없다는 한계점이 있었다. 이에 본 연구에서는 Stein 외(2009)가 제시한 인지적 요구 수준에 따라 프로그래밍 과제의 인지적 수준과 요소를 평가할 수 있는 프레임워크를 개발하였다. 그리고 이를 적용하여 2022 개정 교육과정 중학교 정보 교과서 10종에 수록된 프로그래밍 과제를 수준별, 성취기준별, 과제 흐름별로 분석하여 컴퓨팅 사고력 향상에 적합한 과제 분포 및 계열성을 제시하였다.

## 2. 이론적 배경

### 2.1 과제의 인지적 요구 수준

구성주의 관점에 따르면 궁극적으로 학생이 지식을 창조하기 위해서는 교사가 제공한 지식뿐만 아니라 다음과 같은 학습 과정을 거쳐야 한다[23]. Bloom은 그의 디지털 텍소노미(Digital Taxonomy) 연구에서 학습 과정을 다음과 같이 제시했다. 우선 지식을 기억하고 이해하는 기반을 다지는 '지식 획득' 과정을 거친 후, 획득한 지식을 문제 분석 및 해결 적용에 사용하는 '지식 심화' 과정을 거쳐 궁극적으로 사고력을 발휘하고 향상할 수 있는 '지식 창조' 과정으로 연계되어야 한다. 즉, 학습자가 사고력을 개발할 수 있는 학습 과정은 지식을 획득하는 기억하기, 이해하기와 같은 저차원적 사고에서 지식을 심화하는 분석하기, 종합하기, 평가하기 등과 같은 고차원적 사고력으로 나아가는 순차적인 과정임을 확인할 수 있다.

정보 교과에서도 컴퓨팅 사고력을 비롯한 인지적 능력의 향상을 위해 이러한 학습 과정의 확보가 필요함을 강조하고 있다. 김자미 외(2020)는 2022 개정 정보과 교육과정의 방향으로 지식의 계속성, 학문적 연계성, 내용의 계열성을 제안하였고, 교육과정의 프레임 또한 학문적 지식체계에 근거해야 함을 강조하였다. 특히, 정보 교과 영역 중 프로그래밍은 학습자가 성취감을 느낄 수 있도록 수준별로 제시할 것을 강조하며 과제의 수준이 점진적으로 증진되도록 구성하는 것도 중요하다[4].

과제의 인지적 요구 수준(Cognitive demand Level) 프레임워크란 학생들이 과제를 성공적으로 해결하는 데 필요한 지식 및 사고 과정의 종류에 따라 과제의 수준을 분류한 프레임워크이다. Stein 외(2009)는 수학 과제 해결 시 학생에게 어떠한 수준의 사고까지 필요로 하는지 나타낼 기준의 필요성을 인식하고, 이를 저수준(low-level)과 고수준(high-level)으로 나누었다. 그리고 저수준, 고수준을 각 2개씩 다시 나누어 총 4개의 수준에 따라 과제를 분류하였다.

가장 낮은 수준의 과제는 기억하기(Memorization, M) 과제로 학습자가 이전에 학습한 사실, 규칙, 공식, 정의를 기억만 해내면 해결할 수 있는 과제이다. 복잡한 사고 과정을 요구하지 않고 단순한 계산으로 문제를 해결할 수 있어 바로 답을 도출할 수 있는 과제가 많다. 이 문제를 해결하는 데 필요한 개념은 문제에 명백히 제시되므로 문제 분석을 크게 요구하지 않는다. 또한, 문제 해결 과정에서 사전 학습한 개념 간의 연결성을 필요로 하지 않고 단독 개념을 활용하도록 한다.

연계성 없는 절차(Procedures Without Connections, PNC) 과제는 문제 해결 과정 또는 알고리즘, 방법을 나타내는 절차적 지식을 요구한다. 그러나 그 절차가 이전의 학습 내용 또는 현재 과제에 명백히 제시되어 있다. 학생이 과제에 사용된 개념과 구조의 의미를 깊이 이해하기보다 정답을 도출하는 데 초점을 두므로 정형화된 해결 방식을

갖고 있다. 이러한 한계로 인해 제한된 인지적 사고를 요구하며, 높은 수준의 인지적 사고력 향상을 기대하기 힘들다.

연계성 있는 절차(Procedures With Connections, PWC) 과제는 절차적 지식을 통해 과제를 해결한다는 점에서 PNC와 공통되지만, 과제에 사용된 개념과 해결 절차의 이해 수준을 향상시키기 위한 과제라는 점에서 차이가 있다. 과제를 해결할 수 있는 일반적인 절차는 제시되어 있지만, 이 과제를 해결하기 위해서는 다양한 개념과 아이디어, 절차적 지식을 종합 및 변형이 필요하다.

마지막으로 행하기(Doing Mathematics, DM) 과제는 가장 높은 수준의 인지적 사고를 요구하는 과제로, 학생들은 과제를 해결하기 위해 과목에 대한 개념뿐만 아니라 과목 외적 지식, 생활할 경험을 활용해야 할 수도 있다. 과제를 해결할 수 있는 절차나 전략이 정해져 있지 않고 학생이 해결 과정 전반을 직접 설계하고 수행해야 한다. 설계한 방법의 결과를 예측할 수 없으므로 해결 과정에 대한 메타인지도 필요하다.

이 프레임워크는 수학 과제를 분석하기 위해 개발되었으나 그외에 다른 교과와 과제 분석에도 사용되었다. Tekkumru-Kisa(2015)는 과학 교육 연구자와 교육자에게 성공적인 과학 학습을 위해 적절한 인지적 수준과 과학적 내용을 포함한 과제 설계를 위해 인지적 과제 요구 수준 프레임워크를 활용한 과학 과제 분석 프레임워크를 제작하였다[26]. Rich 외(2024)는 언플러그드 및 프로그래밍 과제 학습을 통해 길러진 컴퓨팅 사고력이 수학 과제 해결에 도움이 되는지 확인하기 위해 이 프레임워크를 활용하였다[27]. 그는 컴퓨팅 사고력 또한 다양한 맥락을 지닌 문제들을 컴퓨터 과학의 원리와 개념을 활용하여 해결하며, 일련의 알고리즘을 띤 사고가 과제 해결에 필요하다는 점에서 수학 과제의 해결 절차와 일통한다고 주장하였다. Papert, S.(1980)도 로고 프로그래밍 환경에서의 효과적인 수학 학습 연구를 수행함으로써 이를 증명했다[29].

이처럼 Stein 외(2009)에서 제시된 인지적 요구 분석 프레임워크는 수학 외 교과와 과제 분석을 위해 활용되었다. 또한 수학 과제의 수행에서 사용되는 인지적 역량과 프로그래밍 과제의 수행을 통해 길러지는 인지적 역량, 즉 컴퓨팅 사고력이 매우 밀접하게 연관되어 있으며 그 두 과제 간 연결성 및 통합성이 연구를 통해 증명되었다[30]. 그러므로 본 연구에서는 Stein 외(2009)의 인지적 요구 분석 프레임워크를 바탕으로 프로그래밍 과제의 인지적 요구 수준 프레임워크를 개발하였다. 그러나 Rich 외(2024)는 본 프레임워크가 컴퓨팅 사고력을 세부적으로 분석하는 데 어려움이 있었다는 한계점을 밝혔다. 그러므로 본 연구에서는 이러한 한계를 보완하기 위해 프레임워크를 네 개의 차원으로 나뉜 구체적인 과제 분석 프레임워크를 개발한 황해정&박현파(2013)를 함께 참조하여 연구를 수행하였다[5].

## 2.2 정보 교과서 분석 연구 동향

교과서는 교육과정의 목표와 철학을 구현하는 데 있어 핵심적인 역할을 수행하는 중요한 매개체이다. 특히 정보 교과서의 경우, 프로그래밍과 같은 추상적이고 복잡한 개념을 학생들에게 효과적으로 전달하기 위해 체계적이고 단계적으로 구성되어야 한다. 이러한 점에서 정보 교과서의 역할은 단순히 학습 내용을 전달하는 도구를 넘어, 학생들이 컴퓨팅 사고력과 문제 해결 능력을 키우는 데 필수적인 기반으로 자리 잡고 있다.

중학교 정보 교과서의 분석한 연구는 2007년 개정 교육과정부터 교과서의 목표, 핵심 개념, 학습 요소 등 다양한 측면에서 이뤄졌다. 초기에는 중학교 정보 교과서 선택 기준을 개발하거나[13], 2007 개정 중학교의 조기 정착 방안을 마련하기 위해[14] 교과서를 분석하였다. 또한 정보 교과서의 체제와 학습 내용을 비교 분석하여 개선 방안을 제시하거나[15], 교과서의 영역 구성과 탐구적 경향을 분석하였다[16]. 교과서의 학습 목표가 교육과정의 목표 달성에 부합하는지 분석한 연구도 시행되었다. 최현중(2013; 2014)은 컴퓨팅 사고력 교육에 중학교 정보 교육과정과 7종의 인정 교과서가 적합한지 전문가 집단 분석을 실시하였고[17], Anderson의 교육목표분류법을 이용하여 중학교 정보 교과서의 수업 목표 분석을 시행하였다[18]. 최근에는 데이터 마이닝 기법 등 빅데이터 활용 기술을 도입하거나 다양한 교과서 프레임워크를 활용하여 교과서를 분석한 연구가 수행되었다. 윤대웅 외(2019)는 빅데이터 분석 기법인 텍스트 마이닝을 활용하여 중학교 정보 교과서의 핵심 개념을 파악하고 분석하였다.

이와 같은 흐름을 볼 때 정보 교과서의 교과서 연구는 분석의 깊이와 범위가 확대되고 있으며 새로운 분석 기법을 도입해 교육과정의 변화를 반영하고 이를 선도하는 방향으로 발전하고 있다. 그러나 다음과 같은 한계점도 존재한다. 첫째, 많은 연구가 교과서의 학습 목표 분석에 치중하고 있어 컴퓨팅 사고력을 배양할 수 있는 교과서 설계가 이루어졌는지 확인하기 어렵다. 둘째, 교과서에 제시된 학습 과제가 컴퓨팅 사고력을 향상할 수 있는지 분석한 연구가 부족하다[20]. 따라서 본 연구에서는 비록 교과서의 학습 과제 개발 기준과는 상이하나 본 연구를 통해 개발한 프레임워크를 적용하여 교과서 내 학습 과제가 컴퓨팅 사고력 함양이라는 정보 교과서의 목표에 적합한지 분석하였다.

## 3. 연구 방법

### 3.1 프로그래밍 과제의 인지적 요구 분석을 위한 프레임워크 개발

#### 3.1.1 프로그래밍 과제의 인지적 요구 분석 프레임워크 개발

프로그래밍 학습 과제의 인지적 수준을 분석하기 위해 수학 학습 과제의 인지적 수준 프레임워크인 Stein 외

(2009)와 이를 세부 기준으로 나눠 기준을 명료히 한 프레임워크[5]를 참조하여 다음과 같이 수정하였다. 우선 분석의 세부 기준을 ‘과제 형태’, ‘해결 절차’, ‘해결 노력 정도’, ‘프로그램적 이해’로 세분화하였다. 또한 각 기준별 수준도 기억하기(M), 연계성 없는 절차(PNC), 연계성 있는 절차(PWC), 프로그래밍 행하기(DP) 과제로 명칭을 수정하였다. 기준별 판별 문항은 내용을 Stein 외(2009)의 프레임워크를 프로그래밍 영역의 특성과 내용이 드러나게 수정하여 Table 1과 같은 초안을 개발하였다.

**Table 1. A Framework for Analyzing Cognitive Demand for Programming Task (draft)**

| Criteria   | Level | Contents   |
|--|-------|--|
| Form of task   | M     | 1. It is in the form of reproducing a fact, law, formula, or definition that has already been learned.   |
|  | PNC   | 2. It is nondescript or presents only a description of the procedure you learned.  |
|  | PWC   | 3. It is presented in a variety of ways (diagrams, manipulatives, problem situations).   |
|  | DP    | 4. It is presented in a form that is close to the student’s relevant knowledge and experience.   |
| Solving procedure  | M     | 5. It is a short-term performance task for which no procedures exist or for which procedures are not available.  |
|  | PNC   | 6. It is solved using an explicit algorithmic procedure learned from previous explanations, experience, and tasks.   |
|  | PWC   | 7. It uses interdisciplinary ideas in problem solving and uses general rather than specific procedures.  |
|  | DP    | 8. It is complex and cannot be solved by an algorithmic process.   |
| Degree of effort to resolve  | M     | 9. It is identical to previously encountered material or the process of reproduction is so obvious that there is no ambiguity.                                       |
|  | PNC   | 10. It requires ‘limited cognitive demands’ to succeed in It. It can be solved without ambiguity by simply recalling what is needed to solve It and how to solve it. |
|  | PWC   | 11. It requires ‘cognitive effort’. You perform It according to the usual procedures but use a number of conceptual ideas required by It.                            |
|  | DP    | 12. It requires you to examine or adjust your own cognitive processes.   |
| 13. It requires significant cognitive effort, and students may experience some level of anxiety due to the unpredictability of the solution. |       |  |
| Understanding the programming concepts   | M     | 14. It does not involve information content concepts (learned or reproduced facts, rules, formulas, definitions, etc.) and can be solved without knowing them.       |
|  | PNC   | 15. It can be solved without knowing the concepts behind the resolution process.   |
| 16. It is focused on getting the answer rather than understanding the concept.   |       |  |

| Criteria                               | Level | Contents  |
|--|-------|---|
| Understanding the programming concepts | PWC   | 17. It requires students to use procedures to solve problems, with an emphasis on understanding the concepts and ideas of the information course. |
|  | DP    | 18. It requires students to explore and understand the nature of concepts, procedures, and relationships in the teaching of information.          |
|  |       | 19. It asks students to analyse itself to investigate the ‘task constraints’ that distinguish between possible and limited solution strategies.   |

**3.1.2 프레임워크 내용 타당성 검증**

초안의 판별 문항들이 프로그래밍 학습 과제의 수준을 평가하기에 적합하고 명료한지 알아보기 위해 전문가의 내용타당도 검사를 실시하였고 I-CVI(Item Content Validity Index) 값으로 내용타당도를 측정하였다[22]. 이는 각 항목에 대해 전문가 합의 비율을 계산하여 타당도를 평가하는 것으로, 전문가의 평가를 체계적으로 정량화함으로써 측정 도구가 연구 목적에 얼마나 적합한지를 객관적으로 판단할 수 있도록 해준다. 내용타당도 검사에 참여한 전문가는 총 11명으로 컴퓨터 교육 교수 1인, 컴퓨터 교육 박사과정 2인, 컴퓨터 교육 석사 3인, 정보 교사 5인이다. 전문가에게 리커트 4점 척도로 응답받았으며, 응답 결과는 Figure 1과 같은 산출식에 따라 계산하였다. 응답한 전문가가 8~12인일 경우, I-CVI 값이 0.78 이상이면 타당도를 확보했다고 볼 수 있다[22].

$$I-CVI = \frac{n}{N}$$

n = number of experts scoring an item with a 3 or 4  
N = total number of expert

**Figure 1. Formula of I-CVI [22]**

검사 결과는 Table 2와 같다. 각 문항에 대한 I-CVI 값을 통해 0.78을 넘는 문항은 내용 타당도를 갖추고 있음을 확인하였다. I-CVI값이 0.78 미만인 2, 8, 14번 문항은 수정 후 다시 내용 타당도 검사를 실시하였다. 또한, 개방형 질문을 통해 전문가의 의견을 수렴하여 문항을 수정하였다.

**Table 2. Statistics Analysis of Content Validity Review**

| Criteria     | item | Mean  | SD    | I-CVI | I-CVI(2nd) |
|--------------|------|-------|-------|-------|------------|
| Form of task | 1    | 3.455 | 1.091 | 0.818 | -          |
|              | 2    | 3.091 | 1.046 | 0.727 | 0.909      |
|              | 3    | 3.455 | 1.091 | 0.818 | -          |
|              | 4    | 3.727 | 1.021 | 0.909 | -          |

| Criteria                               | item | Mean  | SD    | I-CVI | I-CVI(2nd) |
|--|------|-------|-------|-------|------------|
| Solving procedure                      | 5    | 3.636 | 1.016 | 0.909 | -          |
|  | 6    | 3.727 | 0.905 | 1.000 | -          |
|  | 7    | 3.727 | 0.905 | 1.000 | -          |
|  | 8    | 3.000 | 1.092 | 0.636 | 0.818      |
| Degree of effort to resolve            | 9    | 3.727 | 0.905 | 1.000 | -          |
|  | 10   | 3.636 | 0.900 | 1.000 | -          |
|  | 11   | 3.545 | 0.888 | 1.000 | -          |
|  | 12   | 3.545 | 0.888 | 1.000 | -          |
|  | 13   | 3.455 | 1.091 | 0.818 | -          |
| Understanding the programming concepts | 14   | 3.091 | 1.046 | 0.727 | 1.000      |
|  | 15   | 3.455 | 0.985 | 0.909 | -          |
|  | 16   | 3.455 | 0.866 | 1.000 | -          |
|  | 17   | 3.545 | 0.888 | 1.000 | -          |
|  | 18   | 3.545 | 1.004 | 0.909 | -          |
|  | 19   | 3.727 | 1.021 | 0.909 | -          |

두 차례의 내용타당도 검사 결과 모든 문항의 I-CVI 값이 0.78을 넘어 내용 타당도를 확보하였다. 또한 전문가의 추가 응답에서 단계별 과제의 인지적 수준이 모호하게 나타났다고 지적받은 문항은 명료한 용어로 수정하였다. 최종적으로 수정된 프레임워크는 Table 3과 같다.

Table 3. A Framework for Analyzing Cognitive Demand for Programming Task

| Criteria          | Level | Contents  |
|-------------------|-------|---|
| Form of task      | M     | 1. It was to write basic code by implementing previously learned programming concepts, syntax, and algorithms.                                      |
|                   | PNC   | 2. It applied the programming concepts, grammar, and algorithms learned in that lesson.   |
|                   | PWC   | 3. It presented in a variety of ways (tables, symbols, problem situations).   |
|                   | DP    | 4. It is presented in a way that is similar to the student's real-world knowledge and experience.   |
| Solving procedure | M     | 5. It requires no algorithmic steps or is completed by writing simple code.   |
|                   | PNC   | 6. It requires to solve using an explicit algorithmic procedure or programming structure learned from previous explanations, experience, and tasks. |
|                   | PWC   | 7. It requires to solve using multiple algorithmic procedures or programming structures.  |
|                   | DP    | 8. It requires a complex procedure that applies a variety of algorithmic procedures and programming constructs.                                     |

| Criteria  | Level | Contents   |
|---|-------|--|
| Degree of effort to resolve   | M     | 9. It's almost identical to the previous example or practice, and is so obvious that there is no ambiguity.  |
|   | PNC   | 10. It requires "limited cognitive effort" when using the underlying concepts and implementation methods. It can be solved without ambiguity by simply recalling what is needed to solve the task and how to solve it. |
|   | PWC   | 11. It requires "cognitive effort" to connect and apply multiple programming concepts and constructs for solving.  |
| Degree of effort to resolve   | DP    | 12. It requires students to be self-checking and adjusting their cognitive processes.  |
|   |       | 13. It requires "significant cognitive effort" because it is needed to design the problem-solving process from scratch.  |
| Understanding the programming concepts  | M     | 14. It doesn't require a deep understanding of programming concepts and structures.  |
|   | PNC   | 15. It can be solved without knowing the exact background concepts used in the solution procedure to get the answer to the problem.  |
|   |       | 16. It is focused on getting the answer rather than understanding the inherent programming concepts.   |
|   | PWC   | 17. It requires students to use procedures to solve problems, with understanding the inherent programming concepts.  |
|   | DP    | 18. It requires students to explore and understand the nature of programming concepts, procedures, and relationships.  |
| 19. It requires students to "debug" their programming works by analyzing the results and correcting errors. |       |  |

기억하기(M)는 사전 학습한 프로그래밍 개념, 문법, 알고리즘으로도 해결 가능한 과제로, 이전에 학습한 개념이나 알고리즘을 단순히 반복하는 형태의 과제이다. 그러므로 과제 해결 절차는 간단한 코드 작성 또는 코드 작성 없이도 수행 가능하다. 이전 예제나 실습과 해결 절차가 흡사한 과제이므로 인지적인 노력이 거의 들지 않으며, 프로그래밍의 개념과 구조에 대한 이해가 없이도 해결 가능하다. 연계성 없는 절차(PNC)는 해당 차시에 학습한 프로그래밍 개념, 문법, 알고리즘을 적용하는 형태의 과제이다. 명백한 알고리즘적 절차나 구조를 사용하여 과제를 해결하므로 이를 문제 해결에 적용하기 위한 제한된 수준의 인지적 노력이 필요하다. 과제에 내재한 프로그래밍의 개념과 구조를 정확히 이해하지 못해도 답을 낼 수 있다.

연계성 있는 절차(PWC)는 해결하기 위해 사용해야 할 지식과 기능이 과제에 명백히 제시되지 않고 그림, 상황, 도표 등 다양한 방법으로 제시되어 문제 이해 및 요구 파악이 필요한 과제이다. 이제까지 학습한 프로그래밍 개념

과 구조들을 연결하고 적용하는 과정이 필요하며 이를 위해 인지적인 노력이 필요하다. 문제 상황에 맞게 프로그래밍 개념과 구조를 결합, 응용하기 위해 개념과 구조에 대한 깊은 이해가 필요하다.

프로그래밍 행하기(DP)는 학생의 실생활 소재를 활용해 제작된 과제로 과제 자체에 대한 분석부터 과제 해결 과정 설계, 다양한 프로그래밍 개념 및 구조를 응용까지 복합적인 절차가 요구된다. 수행 결과를 분석하는 메타인지와 오류를 수정하는 디버깅 절차를 요구하는 수준의 과제이다. 이 과정에서 학생들은 해결 과정을 스스로 점검하므로 상당한 인지적 노력이 요구된다.

### 3.2 연구 대상

본 연구는 2022 개정 정보 교육과정에 따라 인정받은 중학교 교과서 10종 중 프로그래밍 과제를 분석 대상으로 삼았다. 분석 대상인 프로그래밍 과제는 2022 개정 교육과정의 프로그래밍 성취기준을 반영한 과제로 정의된다. 그러나, 연구자 간 협의를 거친 결과 성취기준이 반영되었다 하더라도 제시된 코드블록을 재현하는 과제는 학생들의 인지적 활동을 요구한다고 보기 힘들므로 다음과 같은 과제 포함 및 배제 기준을 수립하였다. 포함 기준으로는 첫째, 프로그래밍 성취기준인 [9정03-05], [9정03-06], [9정03-07]을 반영한 과제는 포함하였다. 둘째, 초등학교에서 선행한 학습 요소(예: 데이터의 입출력, 변수 등)를 반영한 과제도 학생의 인지적 활동을 요구하므로 포함하였다. 셋째, 프로그래밍을 선수학습으로 필요로 하는 성취기준 [9정03-08], [9정03-09]를 반영한 과제를 포함하였다. 배제 기준으로는 첫째, 학생의 인지적 활동이 이뤄지지 않는 과제는 제외하였다. 예를 들어, 프로그램 설계를 요구하는 과제에서 모든 코드가 주어져 그대로 재현만 하면 실행되는 단순 따라하기식으로 제시된 과제는 학생의 인지적 활동이 이뤄지기 어렵다는 연구자 간 합의를 통해 분석 대상에서 배제하였다. 둘째, 지필형 평가를 위해 개발된 단원 평가 과제는 배제하였다. 이 과제들은 인지적 활동은 동반하나 성취기준의 지식을 이해했는지에 초점을 맞춘 문제들로 추상화부터 자동화까지 아우르는 컴퓨팅 사고력 향상과는 연관성이 떨어진다고 판단하였다. 이러한 기준에 따라 선정된 프로그래밍 과제는 총 125개이며, 교과서 10종의 분석 대상 과제 개수는 Table 4와 같다.

Table 4. The number of Tasks by Textbook

| Text Book | A  | B  | C | D  | E  | F  | G  | H  | I | J | Total |
|-----------|----|----|---|----|----|----|----|----|---|---|-------|
| N         | 12 | 12 | 7 | 11 | 10 | 17 | 13 | 25 | 9 | 9 | 125   |

(N=The number of the tasks)

### 3.3 과제 분석 방법

과제는 선행 연구의 분석 방법을 참조하여 분석하였다 [5,31]. 우선 연구자별로 과제의 요구사항, 요구 절차 및 해

결 방법, 필요 선행지식을 분석하여 Table 5와 같이 정리하였다. 각자 분석한 이 세 가지 특징이 프레임워크의 어느 측면 및 단계에 해당하는지 분류하였다. 과제의 모든 특징이 한 단계에 해당하는 과제도 있지만 Table 5와 같이 특징들이 여러 단계에 걸쳐 나타나면 가장 많은 특징을 포함하는 수준으로 판단하였다[5]. 과제의 특징이 여러 수준에 산재해 있거나 동수로 분포되어 판단이 어렵거나 연구자 간 의견이 일치하지 않을 때는 연구자 3인의 협의를 통해 인지적 요구 수준을 판단하였다.

Table 5. Example: Analyzing the Level of Task 11

| Task 11: Debugging and Resolving game program             |   |   |      |              |      |       |
|---|---|---|------|--------------|------|-------|
| General features of task                                  | 11-1. Requirements : It requires students to resolve issues arising in a game program where a frog attempts to avoid an eagle. The first issue involves an overlap between the eagle and fly objects, while the second issue concerns to the score being displayed as a negative value.   |   |      |              |      |       |
|   | 11-2. Solution : The code for both the fly and eagle objects is provided, with certain parts left blank for students to complete. The blanks correspond to the y-coordinate value and a conditional statement. Students must adjust the y-value to prevent the two objects from overlapping and insert the condition 'score < 0' to resolve the second issue. |   |      |              |      |       |
| 11-3. Prior Knowledge : Conditional structures, Debugging |   |   |      |              |      |       |
| Task Level Analyzing                                      | criteria  | M | PNC  | PWC          | DP   | Level |
|   | Form of task  |   |      | 11-1         |      | PWC   |
|   | Solving procedure   |   | 11-2 |              |      |       |
|   | Degree of effort to resolve   |   |      | 11-2<br>11-3 |      |       |
|   | Understanding the programming concepts  |   |      |              | 11-3 |       |

## 4. 연구 결과

### 4.1 교과서별 과제 수준 분석 결과

개발된 프레임워크에 따라 2022 개정 정보과 교육과정 기반 교과서 10종에 있는 프로그래밍 과제 125개를 분석하였고 그 결과는 Table 6과 같다.

Table 6. Distribution of cognitive demand levels of programming tasks by textbook

| Text book | Level         |               |               |               | Total |
|-----------|---------------|---------------|---------------|---------------|-------|
|           | M             | MPNC          | PWC           | DP            |       |
| A         | 42%<br>(5/12) | 25%<br>(3/12) | 25%<br>(3/12) | 8%<br>(1/12)  | 12    |
| B         | 42%<br>(5/12) | 8%<br>(1/12)  | 25%<br>(3/12) | 25%<br>(3/12) | 12    |
| C         | 14%<br>(1/7)  | 29%<br>(2/7)  | 14%<br>(1/7)  | 43%<br>(3/7)  | 7     |

| Text book | Level      |            |            |             | Total |
|-----------|------------|------------|------------|-------------|-------|
|           | M          | MPNC       | PWC        | DP          |       |
| D         | 45% (5/11) | 27% (3/11) | 10% (1/11) | 18% (2/11)  | 11    |
| E         | 10% (1/10) | 50% (5/10) | 20% (2/10) | 20% (2/10)  | 10    |
| F         | 41% (7/17) | 12% (2/17) | 47% (8/17) | 0% (0/17)   | 17    |
| G         | 23% (3/13) | 38% (5/13) | 23% (3/13) | 16% (2/13)  | 13    |
| H         | 12% (3/25) | 12% (3/25) | 32% (8/25) | 44% (11/25) | 25    |
| I         | 22% (2/9)  | 11% (1/9)  | 11% (1/9)  | 56% (5/9)   | 9     |
| J         | 33% (3/9)  | 33% (3/9)  | 33% (3/9)  | 0% (0/9)    | 9     |

각 교과서 별 과제의 수준을 분석한 결과 한 수준에 과제가 몰려있거나 하나도 없는 것을 통해 수준별 과제가 고른 분포를 보이지 않았음을 확인하였다. A, B, C, D, F, H 교과서의 경우 전체 과제 중 한 수준의 과제가 절반에 가까운 40%대를 차지하였다. I 교과서는 절반이 넘는 56%의 과제가 DP에 해당했고, F와 J 교과서는 DP 수준에 해당하는 과제가 0%를 나타냈다.

Table 7은 Table 6의 분석 결과를 M과 PNC 수준은 인지적 요구도가 낮은 수준(Low-level)의 과제로, PWC와 DP 수준은 인지적 요구도가 높은 수준(High-level)의 과제로[25] 재분류하여 제시하였다. 또한 프로그래밍 과제를 인지적 요구도가 낮은 수준과 높은 수준과 인지적 요구 수준 척도값(Cognitive Denad Level Value, CDLV)을 나타낸 인지적 요구 정도 분석표를 같이 제시하였다. 인지적 요구 수준 척도는 과제의 인지적 요구도에서 가장 낮은 수준은 1점으로, 가장 높은 수준은 2점으로 수치화하여 평균 점수로 나타낸 값이다[31,33]. 전체 과제의 인지적 수준 척도의 중앙값은 1.51이었다. 본 연구에서는 중앙값을 기준으로  $\pm 0.25$ 인 1.39 이상 1.62 이하를 ‘중’수준 구간으로 설정하였다. 이보다 낮은 1 이상 1.38 이하는 ‘저’로, 이보다 높은 1.63 이상 2 이하는 ‘고’수준으로 분류하였다.

Table 7. Analysis of cognitive demand levels and scale values of textbook programming tasks

| Text book | Level |      | Total | Cognitive Demand Level Value | Cognitive Demand Level |
|-----------|-------|------|-------|------------------------------|------------------------|
|           | Low   | High |       |                              |                        |
| A         | 8     | 4    | 12    | 1.33                         | Low                    |
| B         | 6     | 6    | 12    | 1.50                         | Medium                 |
| C         | 3     | 4    | 7     | 1.57                         | Medium                 |
| D         | 8     | 3    | 11    | 1.27                         | Low                    |
| E         | 4     | 6    | 10    | 1.60                         | Medium                 |
| F         | 9     | 8    | 17    | 1.47                         | Medium                 |
| G         | 8     | 5    | 13    | 1.38                         | Low                    |
| H         | 6     | 19   | 25    | 1.76                         | High                   |
| I         | 3     | 6    | 9     | 1.67                         | High                   |
| J         | 6     | 3    | 9     | 1.33                         | Low                    |
| Total     | 61    | 64   | 125   | 1.51                         |                        |

교과서 내 과제의 전반적인 인지적 요구 수준이 ‘저’에 해당하는 교과서는 A 교과서(1.33), D 교과서(1.27), G 교과서(1.38), J 교과서(1.33)이다. 이 교과서 과제 중 낮은 수준의 과제가 차지하는 비율이 전체 과제의 2/3 이상이였다. 교과서의 전반적인 인지적 요구 수준이 ‘중’에 해당하는 교과서는 B 교과서(1.5), C 교과서(1.57), E 교과서(1.60), F 교과서(1.47)로, 이 교과서들의 전반적인 프로그래밍 과제의 인지적 요구 수준이 중앙값인 1.51 주위에 분포해 있다. 그러므로 이 교과서들은 낮은 수준과 높은 수준의 과제가 고루 분포되어 있다. 교과서의 전반적인 인지적 요구 수준이 ‘고’에 해당하는 교과서와 척도값은 H 교과서(1.76), I 교과서(1.67)였다. 이들 내 높은 수준의 과제가 차지하는 비율은 전체 과제의 2/3 이상을 보였다.

#### 4.2 성취기준별 과제 수준 분석 결과

2022 개정 정보과 교육과정의 성취기준 중 프로그래밍의 지식·이해 요소를 담고 있는 성취기준은 ‘[9정03-05] 데이터를 순차적으로 저장할 수 있는 구조를 활용하여 문제 해결 프로그램을 작성한다.’, ‘[9정03-06] 논리 연산과 중첩 제어 구조를 활용하여 문제를 해결하는 프로그램을 작성한다.’, ‘[9정03-07] 프로그램 작성에서 함수를 활용하고, 프로그램 수행 결과를 디버거로 분석하여 오류를 수정한다.’, ‘[9정03-08] 실생활의 문제를 탐색하여 발견하고, 프로그래밍을 통해 해결한다.’이다. 여기에 ‘[9정03-09] 다양한 학문 분야의 문제 해결을 위해 협력하여 소프트웨어를 개발한다.’라는 프로그래밍 학습 내용을 실생활 및 다양한 학문에 적용하는 프로그래밍 융합 교육에 해당하므로, 본 연구에서 포함하여 분석하였다. 교과서별 각 성취기준에 해당하는 과제 개수는 Table 8과 같다. ‘그 외(Other)’로 분류된 과제는 초등학교 성취기준의 내용을 복습하는데 초점이 맞추어져 있는 과제이므로 이는 분류 대상에 포함되지 않는다.

Table 8. Number of assignments for each achievement standard in a textbook

| Text-book | Criteria Level |              |              |              |              |       | Total |
|-----------|----------------|--------------|--------------|--------------|--------------|-------|-------|
|           | 9Jeong 03-05   | 9Jeong 03-06 | 9Jeong 03-07 | 9Jeong 03-08 | 9Jeong 03-09 | Other |       |
| A         | 2              | 2            | 2            | 1            | 1            | 4     | 12    |
| B         | 1              | 4            | 2            | 1            | 2            | 2     | 12    |
| C         | 1              | 1            | 2            | 1            | 2            | 0     | 7     |
| D         | 4              | 2            | 2            | 1            | 2            | 0     | 11    |
| E         | 1              | 2            | 5            | 1            | 1            | 0     | 10    |
| F         | 4              | 4            | 4            | 2            | 3            | 0     | 17    |
| G         | 1              | 2            | 6            | 0            | 2            | 2     | 13    |
| H         | 2              | 4            | 9            | 4            | 5            | 1     | 25    |
| I         | 1              | 2            | 0            | 2            | 2            | 2     | 9     |
| J         | 1              | 1            | 2            | 3            | 2            | 0     | 9     |
| Total     | 18             | 24           | 34           | 16           | 22           | 11    | 125   |

[9정03-05]에 해당하는 과제의 수준별 개수는 Table 9와 같다. 총 18개의 과제 중 M은 9개, PNC는 3개, PWC는 5개, DP는 1개가 있었다. 이 성취기준에 제시된 과제 개수는 평균 1.8개이며, 인지적 요구 수준은 1.35로 중간 정도의 인지적 수준을 요구했다. 모든 교과서가 이 성취기준에 적어도 1개의 과제를 제시하였다. 대부분 리스트를 생성하거나 항목을 수정 및 삭제함으로써 실제로 문제를 해결하는 프로그램을 작성하는 형태의 과제를 제시하였으나, 버킷리스트 적어보기와 같이 언플러그드 과제만 제시한 교과서도 있었다.

Table 9. [9Jeong03-05] Distribution of cognitive demands of tasks

| Text-book | Level |     |     |    | Total | Cognitive Demand Level Value |
|-----------|-------|-----|-----|----|-------|------------------------------|
|           | M     | PNC | PWC | DP |       |                              |
| A         | 2     | 0   | 0   | 0  | 2     | 1                            |
| B         | 1     | 0   | 0   | 0  | 1     | 1                            |
| C         | 0     | 1   | 0   | 0  | 1     | 1                            |
| D         | 3     | 0   | 1   | 0  | 4     | 1.25                         |
| E         | 0     | 0   | 1   | 0  | 1     | 2                            |
| F         | 2     | 1   | 1   | 0  | 4     | 1.25                         |
| G         | 0     | 1   | 0   | 0  | 1     | 1                            |
| H         | 0     | 0   | 2   | 0  | 2     | 2                            |
| I         | 0     | 0   | 0   | 1  | 1     | 2                            |
| J         | 1     | 0   | 0   | 0  | 1     | 1                            |
| Total     | 9     | 3   | 5   | 1  | 18    | -                            |
| Average   |       |     |     |    | 1.8   | 1.35                         |

[9정03-06]에 해당하는 과제의 수준별 개수는 Table 10과 같다. 총 24개의 과제 중 M은 5개, PNC는 9개, PWC는 8개, DP는 2개가 있었다. 이 성취기준에 제시된 과제 개수는 평균 2.4개이며, 인지적 요구 수준은 1.2로 낮은 인지적 수준을 요구했다. 모든 교과서가 이 성취기준에 적어도 1개의 과제를 교과서에 제시하였다. A, B, G, H 교과서의 경우, 해당 성취기준을 반영한 과제를 제시하기 전 선수 학습했던 변수와 연산자에 대한 과제를 제시하였다.

Table 10. [9Jeong03-06] Distribution of cognitive demands of tasks

| Text-book | Level |     |     |    | Total | Cognitive Demand Level Value |
|-----------|-------|-----|-----|----|-------|------------------------------|
|           | M     | PNC | PWC | DP |       |                              |
| A         | 0     | 1   | 1   | 0  | 2     | 1.5                          |
| B         | 2     | 0   | 1   | 1  | 4     | 1.5                          |
| C         | 1     | 0   | 0   | 0  | 1     | 1                            |
| D         | 0     | 2   | 0   | 0  | 2     | 1                            |
| E         | 1     | 1   | 0   | 0  | 2     | 1                            |
| F         | 1     | 0   | 3   | 0  | 4     | 1.75                         |
| G         | 0     | 2   | 0   | 0  | 2     | 1                            |
| H         | 0     | 1   | 2   | 1  | 4     | 1.75                         |
| I         | 0     | 1   | 1   | 0  | 2     | 1.5                          |
| J         | 0     | 1   | 0   | 0  | 1     | 1                            |
| Total     | 5     | 9   | 8   | 2  | 24    | -                            |
| Average   |       |     |     |    | 2.4   | 1.2                          |

교과서별 [9정03-07]에 해당하는 과제의 수준별 개수는 Table 11과 같다. 총 35개의 과제 중 M은 7개, PNC는 8개, PWC는 14개, DP는 5개가 있었다. 이 성취기준에 제시된 과제 개수는 평균 3.4개로 5개의 성취기준 중 가장 많았으며, 인지적 요구 수준은 1.39로 중간 수준의 인지적 수준을 요구했다. 모든 교과서가 이 성취기준에 적어도 1개의 과제를 교과서에 제시하였으나, I 교과서는 이 성취기준을 달성하기 위한 단독 과제를 제시하지 않았고, 실생활 융합 과제에서 본 성취기준의 내용 요소를 포함한 과제를 제시하였다.

Table 11. [9Jeong03-07] Distribution of cognitive demands of tasks

| Text-book | Level |     |     |    | Total | Cognitive Demand Level Value |
|-----------|-------|-----|-----|----|-------|------------------------------|
|           | M     | PNC | PWC | DP |       |                              |
| A         | 1     | 0   | 1   | 0  | 2     | 1.5                          |
| B         | 0     | 0   | 1   | 1  | 2     | 2                            |
| C         | 0     | 1   | 1   | 0  | 2     | 1.5                          |
| D         | 2     | 0   | 0   | 0  | 2     | 1                            |
| E         | 0     | 2   | 3   | 0  | 5     | 1.5                          |
| F         | 0     | 1   | 3   | 0  | 4     | 1.75                         |
| G         | 2     | 2   | 2   | 0  | 6     | 1.34                         |
| H         | 2     | 1   | 2   | 4  | 9     | 1.67                         |
| I         | 0     | 0   | 0   | 0  | 0     | 0                            |
| J         | 0     | 1   | 1   | 0  | 2     | 1.5                          |
| Total     | 7     | 8   | 14  | 5  | 34    | -                            |
| Average   |       |     |     |    | 3.4   | 1.39                         |

교과서별 [9정03-08]에 해당하는 과제의 수준별 개수는 Table 12와 같다. 총 16개의 과제 중 M은 5개, PNC는 3개, PWC는 3개, DP는 5개가 있다. 이 성취기준에 제시된 과제 개수는 평균 1.6개였고 인지적 요구 수준은 1.38로 낮은 인지적 수준을 요구하고 있다. 모든 교과서가 이 성취기준에 적어도 1개의 과제를 교과서에 제시하였으나, G 교과서에는 실생활과 연계한 프로그램 설계 예제만 있었다. H 교과서의 경우, 이 성취기준에 해당하는 과제를 각각 인지적 사고 수준을 M 수준부터 DP 수준까지 차례대로 4개 제시하였다.

Table 12. [9Jeong03-08] Distribution of cognitive demands of tasks

| Text-book | Level |     |     |    | Total | Cognitive Demand Level Value |
|-----------|-------|-----|-----|----|-------|------------------------------|
|           | M     | PNC | PWC | DP |       |                              |
| A         | 0     | 0   | 1   | 0  | 1     | 2                            |
| B         | 0     | 1   | 0   | 0  | 1     | 1                            |
| C         | 0     | 0   | 0   | 1  | 1     | 2                            |
| D         | 0     | 1   | 0   | 0  | 1     | 1                            |
| E         | 0     | 0   | 0   | 1  | 1     | 2                            |
| F         | 2     | 0   | 0   | 0  | 2     | 1                            |
| G         | 0     | 0   | 0   | 0  | 0     | 0                            |
| H         | 1     | 1   | 1   | 1  | 4     | 1.5                          |

| Text-book | Level |     |     |    | Total | Cognitive Demand Level Value |
|-----------|-------|-----|-----|----|-------|------------------------------|
|           | M     | PNC | PWC | DP |       |                              |
| I         | 0     | 0   | 0   | 2  | 2     | 2                            |
| J         | 2     | 0   | 1   | 0  | 3     | 1.34                         |
| Total     | 5     | 3   | 3   | 5  | 16    | -                            |
| Average   |       |     |     |    | 1.6   | 1.38                         |

교과서별 [9정03-09]에 해당하는 과제의 수준별 개수는 Table 13과 같다. 이 성취기준에 제시된 과제 개수는 평균 2.2개이며, 인지적 요구 수준은 1.89로 모든 성취기준 중 가장 높은 인지적 수준을 요구한다. 모든 교과서가 이 성취기준에 적어도 1개의 과제를 교과서에 제시하였으나 교과서별 수준의 편차가 컸다. 소프트웨어 개발의 전 과정을 거치도록 요구하는 과제도 있었지만, 문제 해결 과정 중 한두 개의 코드만 빈칸으로 뚫어놓고 채우도록 하는 형태의 과제도 있었다.

Table 13. [9Jeong03-09] Distribution of cognitive demands of tasks

| Text-book | Level |     |     |    | Total | Cognitive Demand Level Value |
|-----------|-------|-----|-----|----|-------|------------------------------|
|           | M     | PNC | PWC | DP |       |                              |
| A         | 0     | 0   | 0   | 1  | 1     | 2                            |
| B         | 0     | 0   | 1   | 1  | 2     | 2                            |
| C         | 0     | 0   | 0   | 2  | 2     | 2                            |
| D         | 0     | 0   | 0   | 2  | 2     | 2                            |
| E         | 0     | 0   | 0   | 1  | 1     | 2                            |
| F         | 2     | 0   | 1   | 0  | 3     | 1.34                         |
| G         | 0     | 0   | 0   | 2  | 2     | 2                            |
| H         | 0     | 0   | 0   | 5  | 5     | 2                            |
| I         | 0     | 0   | 0   | 2  | 2     | 2                            |
| J         | 0     | 1   | 1   | 0  | 2     | 1.5                          |
| Total     | 2     | 1   | 3   | 16 | 22    | -                            |
| Average   |       |     |     |    | 2.2   | 1.89                         |

### 4.3 교과서 내 과제 수준 흐름 분석

10종의 정보 교과서에 제시된 프로그래밍 과제가 인지적으로 계열성 있게 제시되었는지 확인하기 위해 과제의 수준 흐름을 분석하였다. 교과서 내 과제를 가로축에 표시하고, 과제의 수준을 세로축에 표시하여 수준 변화를 나타내었고 총 3가지의 양상을 발견하였다.

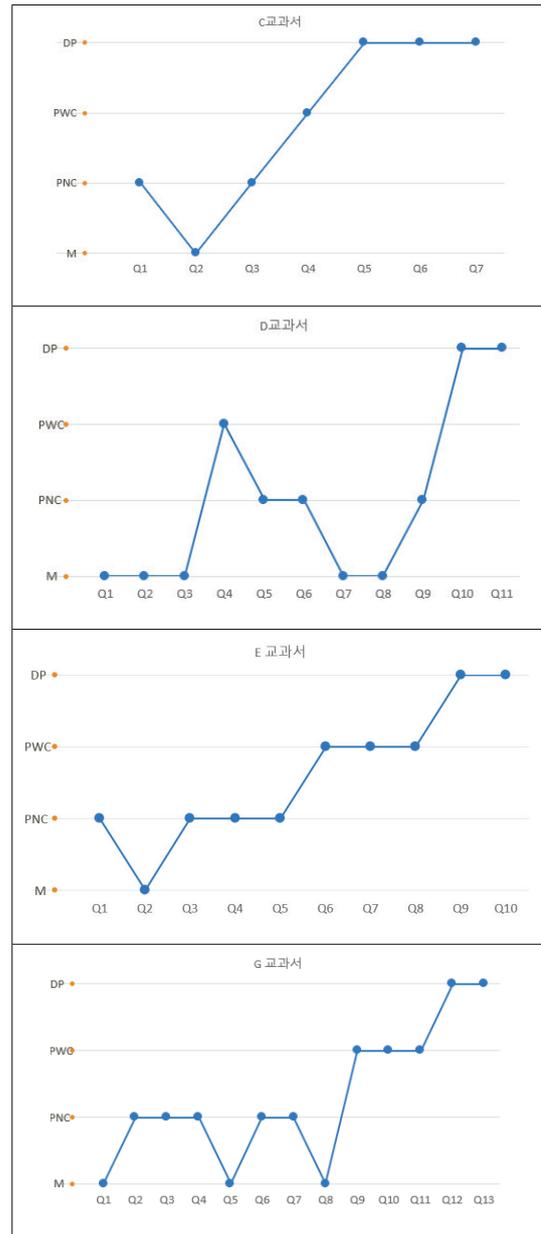
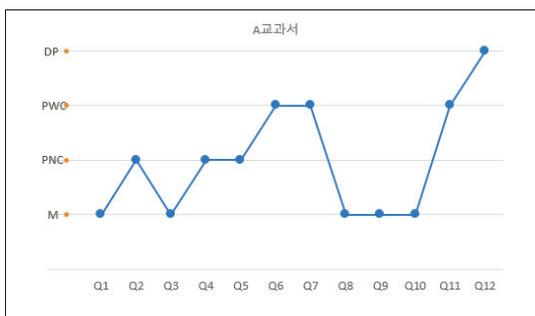


Figure 2. First type of assignment level flow

A, C, D, E, G 교과서의 과제 수준 흐름은 Figure 2와 같다. 위 교과서들은 과제 흐름에 따라 수준이 1~2수준의 차이로 제시되었으며, 전체적인 흐름은 단계별로 우상향하는 흐름을 보였다. 모든 수준의 과제가 골고루 제시되었고, 고차원의 인지적 사고를 요구하는 과제로 진입하기 전 저차원적 사고를 수행하는 과제를 제시하여 체계적으로 과제를 제시하였음을 확인하였다.

B, H, I 교과서의 과제 수준 흐름은 Figure 3과 같이 인접한 과제의 수준이 3수준 이상 건너뛰는 큰 폭의 변동 양상을 보였다. 특히 H, I 교과서의 경우 DP 수준의 과제가 다른 교과서보다 많으나 그 전에 학생들이 학습한 지식을 확인하거나 적용하는 수준의 과제가 제시되지 않고 바로 DP 수준의 과제가 제시되었다.

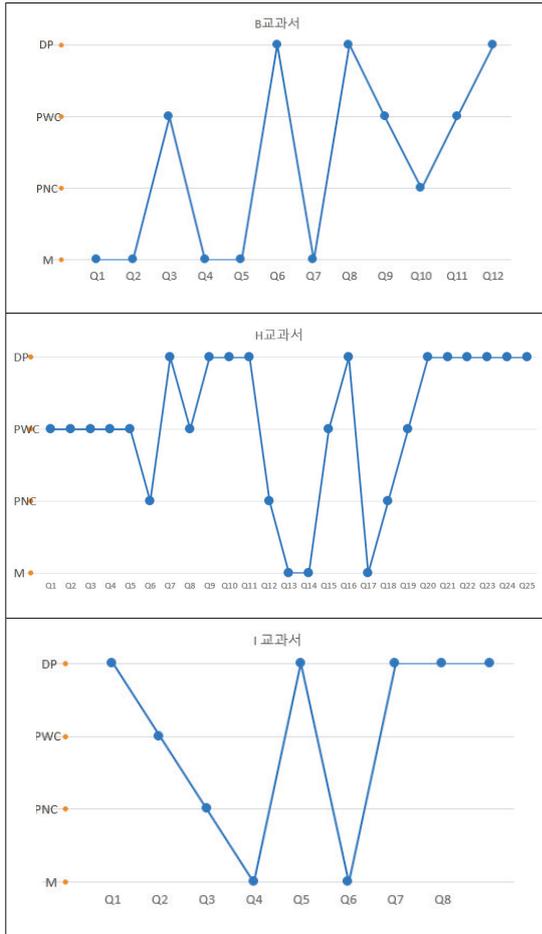


Figure 3. Second type of assignment level flow

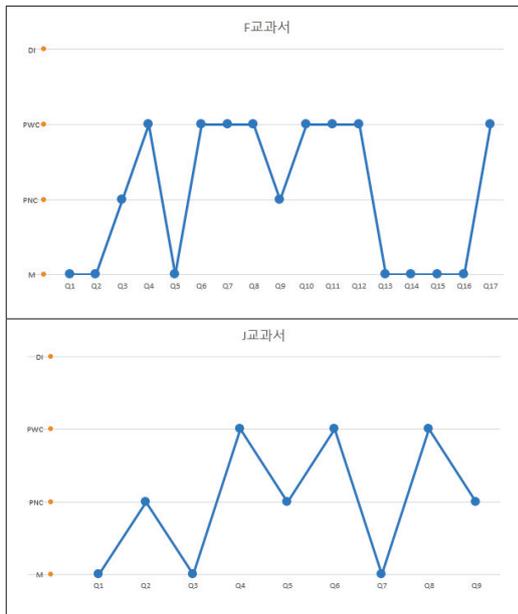


Figure 4. Third type of assignment level flow

F, J 교과서의 경우 과제 수준 흐름은 Figure 4와 같으며 DP 수준의 과제는 제시되지 않아 전체적인 흐름이 수준이 낮은 과제에 몰려있는 양상을 보였다. 이는 두 교과서

에 실린 실생활 과제의 대부분이 고차원적 사고력을 요하는 것처럼 보이나, 프로그램의 코드를 한두 개 채우거나 의사코드를 보고 그대로 블록을 채우기만 하면 해결되는 과제로 제시되어 낮은 수준의 과제로 평가되었기 때문이다. 이는 고차원적 사고력 향상을 위해 설계되었으나 교사의 수업 곤란도와 학생의 과제 해결 기피 등을 고려하여 과제의 인지 요구를 낮추는 인지적 격하(Cognitive demand decline) 현상[26]이 나타난 과제로 판별된다.

### 5. 결론 및 제언

본 연구는 컴퓨팅 사고력 향상을 위해 프로그래밍 과제의 인지적 수준을 분석하는 프레임워크를 개발하였다. 이를 2022 개정 정보과 중학교 교과서 10종의 프로그래밍 과제 분석에 적용하여 인지적 요구 수준을 분석하였다. 분석한 결과 다음과 같은 결론을 얻을 수 있었다.

첫째, 프로그래밍 학습 과제의 인지적 요구 분석 프레임워크는 기억하기(M), 연계성 없는 절차(PNC), 연계성 있는 절차(PWC), 프로그래밍 행하기(DP)라는 총 4가지 수준으로 과제를 분석한다. 분석의 세부 기준은 과제 형태, 해결 절차, 해결 노력 정도, 프로그래밍적 이해로 세분화하였다. 세부 기준별 판별 문항을 구체적이고 명료하게 제시하기 위해 전문가 내용타당도를 거쳐 검증받아 프레임워크의 타당도를 갖추었다.

둘째, 프레임워크를 적용하여 프로그래밍 학습 과제를 분석한 결과는 다음과 같다. 교과서 10종 중 5종에 제시된 프로그래밍 과제는 한 수준의 과제가 전체 과제의 2/3 이상을 차지하는 과제 편중 현상을 보였다. 낮은 인지 수준의 과제만 제공한다면 프로그래밍의 개념과 구조의 깊은 이해와 확장된 연결 기회의 결손이 있고, 높은 수준의 과제만 제시된다면 인지적 부담이 높아 학생의 흥미와 학습 지속 욕구가 감소할 수 있다. 학생의 수준을 고려하여 학생이 프로그래밍 개념과 구조로 사고하고 상호 간 연결성을 발견할 수 있는 다양한 수준의 과제 제시가 필요하다[21]. 이를 위해 교사는 본 프레임워크에 기반하여 학생의 사고 수준을 평가하고 이에 맞는 과제를 제시하고 수업을 설계해야 한다. 또한 프로그래밍 관련 성취기준별 성취기준별 제시된 평균 과제 수는 최대 3.5개로 네 수준을 모두 반영할 수 있는 만큼인 4개도 되지 않았다. 각 수준별 최소 1개 이상의 과제가 제시되어야 성취기준의 깊이 있는 달성이 가능할 것이다.

셋째, 효과적인 컴퓨팅 사고력 향상을 위해 과제의 인지적 요구 수준을 고려한 체계적인 과제 제시가 필요하다. 교과서 내 과제 수준의 흐름은 3수준을 뛰어넘는 큰 폭의 변동을 보이거나 같은 수준의 과제가 이어졌다. 프로그래밍 학습은 학습자의 인지적 부담을 적절히 관리하여[24] 학습을 이어갈 수 있도록 점진적인 인지적 흐름을 고려한 과제 구성이 필요하다. 또한 과제 자체는 고수준이나 해결 절차는 간단하게 한 인지적 격하 현상을 보이는 과제도 있었다. 고차원적 사고력 향상을 위해 높은 수준의 과제 해결에 도전하는 것은

필수이다. 과제의 수준을 격하하기보다 학생의 학습 부담을 낮추고 도전의식을 기를 수 있는 수업 설계와 지원이 필요하다. 이를 위해 과제 분석 및 적절한 교수법을 적용할 수 있는 교사의 전문성이 요구된다[26].

본 연구는 프로그래밍 과제의 인지적 요구 수준을 평가할 수 있는 프레임워크를 개발하고 2022 개정 교과서 내 프로그래밍 과제를 분석하였다. 향후 프레임워크를 통해 학생의 사고력을 길러줄 수 있는 과제 설계와 수준별 적절한 교수법을 적용할 수 있는 교사 전문성 개발 등의 후속 연구로 이어져야 한다.

### 참고문헌

- [1] Yoon, O. H. (2022). Analysis of the meaning of the 2022 revised curriculum. *The Journal of the Convergence on Culture Technology*, 8(5), 59-69. <https://doi.org/10.17703/JCCT.2022.8.5.59>
- [2] Ministry of Education. (2022). *2022 revised practical arts (technology and home economics) and informatics curriculum*. Ministry of Education.
- [3] Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35. <https://doi.org/10.1145/1118178.1118215>
- [4] Kim, J. M., Woo, H. S., Yang, H. J., Kim, M. J., Kim, S. H., Lee, S. Y., Kim, B. S., Kim, Y. A., Kwak, J. H., Choi, H. J., Jung, I. K., Lee, Y. J., & Lee, W. K. (2020). Proposal for a standard curriculum in informatics for the 2022 revision. *The Journal of Korean Association of Computer Education*, 23(1), 1-28. <https://doi.org/10.32431/kace.2020.23.1.001>
- [5] Hwang, H. J., & Jeong, J. hye. (2017). An Exploration of Cognitive Demand Level in MiC Textbook based on the Tasks of "Data Analysis and Probability". *Communications of Mathematical Education*, 31(1), 103-123. <https://doi.org/10.7468/JKSMEE.2017.31.1.103>
- [6] Jung, U., & Lee, Y. (2018). SW education in secondary school and implementation cases. *Communications of the Korean Institute of Information Scientists and Engineers*, 36(11), 16-21.
- [7] Swaid, S., & Suid, T. (2019). Computational thinking education: Who let the dog out?. *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, Las Vegas, NV, USA, 788-792. <https://doi.org/10.1109/CSCI49370.2019.00150>
- [8] Choi, H. S. (2019). The Analysis of Mathematical Tasks for developing the core competencies in High School Mathematics textbook. *Journal of the Korean Society of Mathematics Education*, 22(2), 95-113. <https://doi.org/10.30807/ksms.2019.22.2.001>
- [9] Kim, K. G., & Lee, J. Y. (2016). Analysis of the Effectiveness of Computational Thinking-Based Programming Learning. *The Journal of Korean Association of Computer Education*, 19(1), 27-39. <https://doi.org/10.32431/kace.2016.19.1.004>
- [10] Figueiredo, J., & García-Peñalvo, F. J. (2017). Improving computational thinking using follow and give instructions. *Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM 2017)*, 1-7. <https://doi.org/10.1145/3144826.3145351>
- [11] Li, Y. (2016). Teaching programming based on computational thinking. *2016 IEEE Frontiers in Education Conference (FIE)*, Erie, PA, USA, 1-7. <https://doi.org/10.1109/FIE.2016.7757408>
- [12] Grover, S., & Pea, R. (2013). Computational Thinking in K-12: A Review of the State of the Field. *Educational researcher*, 42(1), 38-43. <https://doi.org/10.3102/0013189X12463051>
- [13] Choi, G. S., Kim, Y. J., & Lee, J. Y. (2010). Development of Selection Criteria for Informatics Textbooks in Middle School. *The Journal of Korean Association of Computer Education*, 13(5), 1-13. <https://doi.org/10.32431/kace.2010.13.5.001>
- [14] Kang, O. H., & Song, H. H. (2010). Analysis of Informatics-1 textbook based on Informatics Curriculum for Middle School Revised in 2007. *The Journal of Korean Association of Computer Education*, 13(3), 35-45. <https://doi.org/10.32431/kace.2010.13.3.004>
- [15] Kim, J. M., Shim, J. K., Kim, J. M., Lee, W. K., & Park, D. S. (2012). Analysis of Inquiry Tendency in the 'Information Representation and Management' Sections in Middle School Informatics Textbooks. *The KIPS Transactions:PartA*, 19(1), 9-16. <https://doi.org/10.3745/KIPSTA.2012.19A.1.009>
- [16] Jin, Y. H., Hur, M., & Kim, Y. S. (2010). Comparative Content Analysis of Middle School Informatics Textbooks and Suggestions for Improvement. *The Journal of Korean Association of Computer Education*, 13(3), 25-34. <https://doi.org/10.32431/kace.2010.13.3.003>
- [17] Choi, H. J. (2014). Study of Analysis about Learning Objectives of Informatics Textbooks in Middle School using Anderson's Taxonomy of Educational Objectives. *The Journal of Korean Association of Computer Education*, 17(1), 51-63. <https://doi.org/10.32431/kace.2014.17.1.005>
- [18] Choi, H. J. (2013). Extraction of Learning Contexts and Analysis of Informatics Textbooks in Order to Teach The Computational Thinking in Middle School. *The Journal of Korean Association of Computer Education*, 16(6), 45-54. <https://doi.org/10.32431/kace.2013.16.6.005>
- [19] Yoon, D. W., & Choi, H. J. (2019). Analysis of the Core Concepts of Middle School Informatics Textbook Using Big Data Analysis Techniques. *Journal of Creative Information Culture*, 5(2), 157-164. <https://doi.org/10.32823/jcic.5.2.201908.157>
- [20] Kim, S. G. (2024). Analysis of Learning Problems in Middle School Informatics Textbooks : Focusing on Problem Solving and Programming Chapter. *The Journal of Korean Association of Computer Education*, 27(1), 113-125. <https://doi.org/10.32431/kace.2024.27.1.009>

- [21] Stein, M. K., Grover, B. W., & Henningsen, M. (1996). Building student capacity for mathematical thinking and reasoning: An analysis of mathematical tasks used in reform classroom. *American Educational Research Journal*, 33(2), 455-488. <https://doi.org/10.2307/1163292>
- [22] Lynn, M. R.(1986). Determination and quantification of content validity. *Nursing research*, 35(6), 382-386. <https://doi.org/10.1097/00006199-198611000-00017>
- [23] Korea Education and Research Information Service. (2013). *Bloom's digital taxonomy*. Korea Education and Research Information Service.
- [24] Choi, J. W., & Lee, Y. J. (2014). The analysis of learners' difficulties in programming learning. *The Journal of Korean Association of Computer Education*, 17(5), 89-98. <https://doi.org/10.32431/kace.2014.17.5.009>
- [25] Stein, M. K., & Kim, G. (2009). The role of mathematics curriculum materials in large scale urban reform: An analysis of demands and opportunities for teacher learning. in J. T. Remillard, B. A.
- [26] Tekkumru-Kisa, M., Stein, M. K., & Schunn, C. (2015). A framework for analyzing cognitive demand and content-practices integration: Task analysis guide in science. *Journal of Research in Science Teaching*, 52(6), 659-685. <https://doi.org/10.1002/tea.21208>
- [27] Rich, K. M., Yadav, A., & Fessler, C. J. (2024). Computational thinking practices as tools for creating high cognitive demand mathematics instruction. *Journal of Mathematics Teacher Education*, 27, 235-255. <https://doi.org/10.1007/s10857-022-09562-3>
- [28] Stein, M. K., Grover, B. W., & Henningsen, M. (1996). Building student capacity for mathematical thinking and reasoning: An analysis of mathematical tasks used in reform classroom. *American Educational Research Journal*, 33, 455-488. <https://doi.org/10.3102/00028312033002455>
- [29] Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas* (pp. 3-18). Basic Books.
- [30] Rich, K. M., & Yadav, A. (2020). Applying levels of abstraction to mathematics word problems. *TechTrends*, 64(3), 395-403. <https://doi.org/10.1007/s11528-020-00479-3>
- [31] Jung, J. H. (2016). *Analysis of cognitive demand levels in the "Data Analysis and Probability" section of MiC textbooks and exploration of task types* [Doctoral dissertation], Chosun University Graduate School of Education.



#### 이다겸

- 2013년 광주교육대학교 초등교육과(학사)
- 2023년 한국교원대학교 컴퓨터교육과(석사)
- 2023년 ~ 현재 한국교원대학교 컴퓨터교육과 박사과정
- 2013년~ 광주광역시 초등학교 교사

✚ 관심분야 : 정보교육, 인공지능교육, 융합교육, 학습과학

✉ gyeomdalee@gmail.com



#### 김윤정

- 2002년 중앙대학교 영어교육과(문학사)
- 2020년 한국교원대학교 컴퓨터교육과(복수전공)
- 2024년 광주교육정책연구소 특별연수 교사
- 2021년~현재 광주광역시 중등 교사(정보)

✚ 관심분야 : 컴퓨팅 사고력, 인공지능, 정보교과교육

✉ eracie@korea.kr



#### 이영준

- 1988년 고려대학교 전산학과(이학사)
- 1994년 미국 미네소타 대학교 전산학과(Ph.D.)
- 2003년 ~ 현재 한국교원대학교 컴퓨터교육학과 교수

✚ 관심분야 : 지능형시스템, 학습과학, 정보교육, 인공지능교육

✉ yjlee@knue.ac.kr