

# 비전공자 프로그래밍 수업 창의적 산출물의 컴퓨팅 사고력 기반 평가 루브릭 개발

김민자<sup>†</sup> · 유길상<sup>††</sup> · 김현철<sup>†††</sup>

## 요 약

고등교육의 비전공자 대상 정보과학 교육의 수요가 늘고 있지만, 컴퓨팅 사고력 관점에서 적절하게 학생들의 창의적 산출물을 평가하는 도구는 부족하다. 이에 본 연구는 고등교육의 비전공자를 위한 정보과학 수업에서 학생들이 개발한 창의적 산출물을 컴퓨팅 사고력 관점에서 평가하는 루브릭을 개발하기 위한 목적으로 진행되었다. 루브릭 초안의 준거틀(framework)은 CT Practices Design Pattern이며, 개발된 루브릭의 구성 요소는 영역, 역량요소, 평가요소, 평가자료, 평가단계의 다섯 가지이다. 루브릭은 '추상화 모델의 설계', '창의적 산출물의 설계 및 적용', 그리고 '산출물의 자가 평가'의 세 영역으로 구분하여 개발되었다. 전문가 검토를 통해 내용 타당성을 확보하였으며, 신뢰도 측면의 일관성이 검증되었다. 본 연구에서 개발된 루브릭은 해당 목적에 맞게 수정하여 사용할 수 있다.

주제어 : 루브릭, 컴퓨팅 사고력, 비전공자 정보과학 교육, 창의적 산출물

## Development of a scoring rubric based on Computational Thinking for evaluating students' computational artifacts in programming course

Minja Kim<sup>†</sup> · Gilsang Yoo<sup>††</sup> · Hyeoncheol Kim<sup>†††</sup>

### ABSTRACT

The demands of computer science education for non-majors in higher education is increasing but relevant evaluation tools for the students' computational artifacts are lack. This research aims to develop a scoring rubric to assess student's computational artifacts in non-major programming course at Computational Thinking point of view. The rubric was developed based on 'CT Practice Design Pattern' as a framework. The rubric consists of 'domain, skills, evaluation, evaluating resources, and scales'. Domains are 'Design of abstract model', 'Design and application of creative artifacts', and 'Analysis of the artifacts'. Experts reviewed the rubric to ensure contents validity. The rubric is resulted in reliable for consistency. This rubric can be revised and applied to application environment accordingly.

**Keywords** : Scoring rubric, Computational Thinking, Computer Science Education for non-majors, Computational artifacts

---

<sup>†</sup> 정 회 원: 고려대학교 컴퓨터 교육학과 박사수료  
<sup>††</sup> 정 회 원: 고려대학교 정보대학 컴퓨터학과 연구교수  
<sup>†††</sup> 종신회원: 고려대학교 정보대학 컴퓨터학과 교수(교신저자)

논문접수: 2016년 11월 28일, 심사완료: 2017년 1월 25일, 게재확정: 2017년 1월 31일

\* 이 논문 또는 저서는 2015년 대한민국 교육부와 한국연구재단의 지원을 받아 수행된 연구임 (NRF-2015S1A5A2A01011911)

## 1. 서론

정보과학기술은 사회의 모든 분야에 영향을 미치고 있으며, 다양한 영역과 융합하고 있다. 사회 변화에 발맞추어, 여러 국가의 정보과학 교육이 ICT 교육에서 정보과학 내용으로, 접근성이 강화되는 방향으로 변화하고 있다[1][2].

변화를 주도하는 국가로 영국, 일본, 인도 등을 들 수 있다. 영국은 2008년 정보관련 과목인 ICT를 모든 Key Stage에서 배우게 하여 접근성을 확대하였다[3]. 2014년 개정 교육과정에서 과목명을 'ICT'에서 '컴퓨팅'으로 변경하며, 내용적 측면에서 정보과학을 강화하였다[4]. 일본은 초등에서는 '실과'에서, 중학교는 '기술·가정'에서 가르친다. 고등학교는 '사회와 정보', '정보의 과학' 두 과목 중 1과목을 반드시 선택하여 수강해야 한다. 내용적으로도 정보과학 지식을 기반으로 한 문제 해결력에 초점을 두고 있다[5]. 다양한 학제가 존재하는 인도에서는 2013년 정보교육과정(CMC)을 발표하여 초·중등 전 학년에서 정보교육을 받도록 하였다[6].

한국에서도 2007년 개정 교육과정에서 '컴퓨터'에서 '정보'로 과목명을 바꾸며 정보과학 내용을 강화했다. 2015년 개정에서는 내용 강화 뿐 아니라, 초등학교, 중학교에서는 각각 실과와 정보과목에서 필수적으로 정보과학을 배우게 하고, 고등학교에서도 심화선택에서 일반선택으로 바뀌어 접근성이 향상되었다[7].

초·중등 교육에 이어 고등교육도 내용과 접근성 측면에서 정보교육 강화를 위해 노력하고 있다. 강의를 개설하거나 커리큘럼을 변경하는 등 각 학교별로 노력하고 있다[8][9]. 정부는 과제나 사업 형태로 고등교육의 정보과학 교육 강화를 지원하고 있다. 미국의 Transforming Undergraduate Education in Science, Technology, Engineering and Mathematics (TUES)[10]나 한국의 SW 중심대학 사업[11]이 대표적이다.

정보과학교육 변화의 핵심에는 '컴퓨팅 사고력'(Computational Thinking, 이하 CT)이 있다[12][13]. CT는 사고(thinking)로 컴퓨터 과학 전문가들이 문제를 해결하거나, 시스템을 설계하거나, 인간을 이해하거나 등에 도움을 주는 사고

도구이자 개념을 말한다[14]. CT는 완벽하게 정의되지 않았으며, 학자마다, 교육 프로그램마다 다른 견해를 가지고 있다[15][16][17][18][19][20][21][22]. 그럼에도 불구하고 CT는 개인의 역량을 향상시키고, 다른 분야를 발전시키고, 나아가 국가 경쟁력 향상에 도움이 되기에 전공에 상관없이 모두가 배워야 한다는 것에 대부분의 학자들이 동의한다[14].

고등교육에서 정보과학교육 내용 및 접근성 향상으로 비전공자 대상 교육이 확대되고 있다. 비전공자 대상 교육의 핵심은 CT의 강화이다. 비전공자 교육의 목표는 기술자를 양성하는 것이 아니라 각 전공 영역에 정보과학의 기본 개념과 원리를 적용할 수 있는 인재를 길러내는 것에 있기 때문이다.

교육의 목표와 평가는 연계되어야 한다[23]. 평가가 어긋날 경우, 교육의 목표와 방법이 어긋난 평가에 맞춰질 수 있기 때문이다. 비전공자 대상 교육에서도 CT를 강화하고자 하는 목적 달성을 위해서는 학생들의 CT를 평가해야 한다.

CT를 직접 평가하는 것은 매우 어렵다. CT는 직접 평가할 수 없는 '사고'이기 때문이고, CT의 정의가 다양할 뿐 아니라 측정할 만큼 구체적이지 않기 때문이다. 절대적인 CT를 측정하는 것은 거의 불가능하지만, CT라고 생각되는 특성을 평가하여, 특성의 총합으로 CT를 추정하여 평가할 수 있다. 연구자 혹은 교육과정마다 관점에 따라 CT의 세부 요소를 다르게 구성한다[13][14]. 하지만 크게 추상화와 자동화로 보고[15][21], 다양한 세부 요소를 하위에 넣어 분류하는 방법으로 구성할 수 있다. CT를 직접 측정하지는 못하지만 세부 요소를 평가함으로써 학생들의 CT 정도를 간접적으로 확인할 수 있다.

무엇으로 CT를 평가할 지도 고려해야 한다. 특히, 정보과학 기반 수업에서는 학생들이 직접 창의적 산출물을 제작한다. 교수학습 방법으로 산출물 개발을 사용하며, 결과물은 평가 대상으로 활용한다. 산출물은 학생들의 사고를 간접적으로 확인할 수 있는 대상이지만 평가자의 주관에 따라 평가결과가 치우치기 쉽다는 문제가 있다. 평가 루브릭은(Scoring rubrics)은 이를 해결하기 위해 사용하는 평가방법이다. 루브릭은 학생들의 산출

물을 평가하기 위해 만들어진 평가 가이드라인으로, 평가자는 루브릭을 이용하여 에세이, 프로그램, 발표 등 여러 산출물을 좀 더 객관적으로 평가할 수 있다[24][25]. 학생의 산출물에 CT의 요소가 포함되어있는지를 살펴봄으로서, 루브릭을 학생들의 CT 평가에 활용 가능하다.

본 연구는 컴퓨터 과학 비전공자를 대상으로 실시된 프로그래밍 수업에서 학생들의 산출물을 CT 기반으로 평가할 루브릭을 개발하는 것을 목표로 진행되었다. CT의 다양한 요소 분류 중 CT를 교육 평가적 관점에서 본 ‘CT Practices Design Pattern (컴퓨팅 사고력 수행 설계 패턴)’ [26]을 기반으로 수업에 맞게 설계하였다. 타당하고 신뢰로운 평가 도구 개발을 위해, 타당도와 신뢰도를 검증하였다.

## 2. Computation Thinking (CT) 기반 평가 루브릭

CT 기반으로 개발된 루브릭을 조사하였다. 국내·외에서 개발과정이 명시된 루브릭 연구를 대상으로 하였다.

국내 연구는 다음과 같다. 최형신[27]은 초등학교 교육에서 대인관계, 협력, CT 능력을 평가하기 위한 도구를 개발하였다. 대인관계는 수업 과정 관찰 시 평가할 수 있는 루브릭을, 협력은 Collaboration Web이라는 평가 도구를, CT는 MIT Media lab에서 스크래치를 운영하며 도출한 CT 요소[28]를 준거틀로 설문조사를 개발하여 평가하였다. 이 준거틀은 <표 1>과 같이 CT를 개념(concepts), 수행(practices), 관점(perspectives)의 3가지 차원으로 나누고 하위 구인을 배치하였다. 설문은 각 구인별로 1문항씩, 4점 척도로, 자기 평가 방식으로 실행되었다.

최형신은 다른 연구[29]에서 초등예비교사의 프로그래밍 수업을 CT기반으로 평가하기 위해 루브릭을 개발하였다. 문헌 연구를 통해 CT의 세부 요소를 ‘절차 및 알고리즘, 병행화 및 동기화, 자료 표현, 추상화, 문제 분해, 시뮬레이션’으로 정의하였다. 연구자는 학생 산출물에서 학생이 CT 역량이 있는지를 확인할 수 있는 증거를 다양하

게 수집하는 것을 목적으로 CT 세부 요소를 정의하였다. 정의한 6개의 요소를 바탕으로 수업을 설계하고 평가 루브릭을 만들었다. 평가 루브릭은 정의된 CT 세부 요소를 기초, 발달, 능숙의 3단계 스케일로 구분하였다.

<표 1> 최형신 연구에서 CT 역량 설문조사에 사용된 준거틀[27]

| 차원                         | 구인                       |
|----------------------------|--------------------------|
| Computational Concepts     | Sequence                 |
|                            | Loops                    |
|                            | Parallelism              |
|                            | Events                   |
|                            | Conditional              |
|                            | Operators                |
| Computational Practices    | Variables/Lists          |
|                            | Incremental/Iterative    |
|                            | Testing/Debugging        |
|                            | Reusing/Remixing         |
| Computational Perspectives | Abstracting/Modularizing |
|                            | Expressing               |
|                            | Connecting               |
|                            | Questioning              |

국의 연구는 다음과 같다. Bort & Brylow[30]는 고등학교 교사 대상 특강에서 교사들이 개발한 교안을 CT 관점으로 평가하기 위해 루브릭을 개발하였다. 이 특강은 CS4HS (Computer Science for High School)라는 고등학교 정보과학교육 교육과정 교사 연수 프로그램이다. 연구자들은 CSTA & ISTE[19]에서 제시한 CT의 9가지 요소인 Data collection, Data Analysis, Data representation, Problem decomposition, Abstraction, Algorithm & procedures, Automation, Simulation, Parallelization에 Connection to other fields를 더하여 CT의 요소를 정의하였다. 10가지 CT요소를 0, 1, 2의 3단계로 나누어 평가하도록 구성하였다.

Sherman & Martin [31]은 프로그래밍 관점으로 CT 평가 요소를 구성하였다. 일반적 CT와 모바일 프로그래밍의 CT 요소는 달라야 한다고 보고, 일반 CT 요소 6가지에 모바일 CT 요소 8가지를 추가하였다(<표 2>). 루브릭은 14가지 요소를 4개 스케일로 분류하여 평가하도록 하였다. 개발된 루브릭을 애플벤터를 활용한 모바일 프로그래밍 수업 산출물 평가에 이용하였다.

<표 2> Shreman & Martin의 모바일 CT 요소[31]

| 일반 CT 요소               | 모바일 CT 요소                           |
|------------------------|-------------------------------------|
| Naming                 | Screen Interface                    |
| Procedural abstraction | Events                              |
| Variables              | Component abstraction               |
| Loops                  | Data persistence                    |
| Conditionals           | Data sharing                        |
| Lists                  | Public web services                 |
|                        | Accelerometer & orientation sensors |
|                        | Location awareness                  |

기존 연구 분석을 통해 얻은 결과는 다음과 같다. 첫째, CT 기반 루브릭 개발 연구의 수가 많지 않다. 연구로 개발 과정이 공개되지 않은 루브릭이 웹상에 많다는 것과 대조적이다. 고등교육 대상 루브릭은 더욱 제한적이다. 비전공자 대상 교육의 확대가 기대되는 바, 향후 CT 기반 루브릭 개발 연구가 확대될 필요가 있다.

둘째, 기 개발된 CT 요소 준거틀을 연구 목적에 맞게 수정하여 적용하였다. 최형신[27] [29]과 Bort & Brylow[30]은 MIT Media Lab[28]과 CSTA & ISTE[19]를 기반으로 CT 기반 평가 루브릭을 개발하였다. Sherman & Martin[31]은 차용 준거틀을 밝히지 않았다. 대부분 연구에서 프레임워크를 도입한 근거에 대해 짧게 설명했거나 명시하지 않았다.

셋째, 대부분의 연구에서 루브릭의 신뢰도 평가가 이루어지지 않았다. 루브릭 또한 평가도구이기 때문에 여러 번 평가하더라도 같은 대상에 대해서는 같은, 혹은 유사한 평가가 내려져야 한다. 때문에 신뢰성 검증은 필수적이다.

따라서 본 연구는 CT 요소 준거틀 도입에 대한 근거를 서술하고 신뢰도 평가를 실시함으로써 기존의 CT 기반 평가 루브릭 연구의 한계점을 보완하고자 하였다.

### 3. 준거틀(framework): CT Practices Design Patterns

루브릭 개발을 위해 준거틀을 설정하였다. 준거틀은 SRI Education의 PACT (Principled Assessment of Computational Development)

프로젝트에서 개발한 ‘CT Practices Design Pattern (컴퓨팅 사고력 수행 설계 패턴, 이하 CTP 설계 패턴)’[26]을 선택하였다. CTP 설계 패턴은 교육 평가 관점에서 개발되었으며, 다음과 같은 특징이 있다.

첫째, CT를 수행(practice) 관점으로 보았다. CT를 개인의 ‘사고(thinking)’에 국한시키지 않고 “존재하고 사고하는 방법(ways of being and doing)”(3p)으로 이해했다. 이 관점은 학생의 생각을 수행 과정에서 평가하도록 하여, 수행을 강조한 정보과학 교육과 맥을 같이 한다. CTP 설계 패턴은 학생의 산출물을 결과 자체로 단순하게 보지 않고, 개발하는 과정이 드러나게 하여 과정을 평가할 수 있도록 하였다.

둘째, CT와 같이 평가하기 어려운 구인을 평가하는데 적절한 Evidence-centered design(ECD)에 근거하여 개발하였다. ECD의 5가지 세부 방법은 Domain analysis, Domain modelling, Conceptual assessment framework, Assessment implementation, Assessment delivery이다.

셋째, 참고 자료의 다양성이다. AP Computer Science Principles와 Exploring Computer Science 등 CS 교육과정, CS 교육 문헌, 과학 교육 문헌, 공학 설계 기준, 커뮤니케이션 및 협업 문헌, 현장 선생님과 교육자 등의 전문가 의견 등을 종합하여 개발하였다.

넷째, 해당 영역의 지식 혹은 기술을 학생이 보유하고 있는지 확인 할 수 있도록 측정 가능한 요소를 제공하였다.

다섯째, 일반화 가능하도록 어떤 산출물을 평가하면 되는지, 어떤 평가 환경이 적절한지, 평가 시 어떤 변수가 발생할지 등에 대한 자세한 설명을 제공하였다.

CTP 설계 패턴은 4개의 CTP 설계 패턴과 2개의 보완(supporting) 설계 패턴으로 구성된다. 설계 패턴은 102개의 FKSA(Focal Knowledge, Skills, Attributes)로 구성되어 있다(<표 3>). 본 연구에서는 ‘설계 패턴’은 ‘영역’(domain), ‘FKSA’는 ‘요소’로 명명하였다.

<표 3> CTP 설계 패턴[26]

|                 | 설계 패턴   | FKSA 수 |
|-----------------|---|--------|
| CTP 설계패턴        | Analyze the effects of developments in computing        | 21     |
|                 | Design and implement creative solutions and artifacts   | 32     |
|                 | Design and apply abstractions and models                | 27     |
|                 | Analyze their computational work and the work of others | 5      |
| Supporting 설계패턴 | Communicate thought processes and results               | 5      |
|                 | Collaborate with peers on computing activities          | 12     |

CTP 설계 패턴의 구성을 보면, CT를 수행관점에서 보고 구성하였음을 알 수 있다. 산출물을 개발하는 과정, 즉, 문제해결 과정을 반영하였다. 먼저 Analyze the effect of developments in computing은 문제 발견 단계로 컴퓨팅으로 문제 해결이 가능한 영역을 이해하였는지 여부를 말한다. Design and implement creative solutions and artifacts는 아이디어 수준의 문제 해결 방법을 자동화 할 수 있게 전환하였는지이다. Design and apply abstractions and models는 학생들이 개발 과정에서 일반화, 패턴화, 구조화 등 추상화 전략을 사용하였는지 여부를 말한다. 마지막 Analyze their computational work and the work of others는 학생이 자신의 산출물을 평가하고, 다른 산출물과 비교 하는지를 의미한다.

CTP 설계 패턴의 이상과 같은 특징이 본 연구의 비전공자의 프로그래밍 수업 산출물을 평가하는데 적절하다고 판단하여, 본 연구에서 이 준거들을 기반으로 루브릭을 개발하였다.

## 4. 연구방법

본 연구는 루브릭을 개발하고 직접 수업에 적용하여 검증하는 단계로 진행되었다.

### 4.1 개발

개발은 문헌연구, 1차 개발, 전문가 검토의 3단계로 진행되었다.

첫째, 문헌연구는 정보과학 교육 분야에서 개발

된 루브릭과 CT 평가 준거들을 검토하였다. SRI Education에서 개발한 CTP 설계 패턴을 평가의 준거틀로 결정하였다.

둘째, CTP 설계 패턴에서 Supporting 설계 패턴 2개와 1개의 CTP 패턴을 제외한 3개의 패턴을 기반으로 1차 루브릭을 개발하였다. 3개의 패턴(Design and implement creative solutions and artifacts, Design and apply abstractions and models, Analyze their computational work and the work of others)을 영역(추상화 모델의 설계, 창의적 산출물의 설계 및 적용, 산출물의 분석)으로 본 수업에 맞게 한글화 및 재정의 작업을 하였다.

3개 패턴의 하위 FKSA 64개 중 본 수업에 타당하다고 판단된 11개를 선정하여 하위 역량요소로 재구성 하였다. 예를 들어, Design and implement creative solutions and artifacts 내의 ‘기존의 프로그램을 새로운 문제나 목표의 관점으로 평가하는 능력(Ability to evaluate existing programs and solutions in light of a new problem and purpose)’의 경우, 산출물 개발 시, 기존 프로그램을 반드시 사용해야 한다는 의무가 없었기 때문에 평가 요소에 반영하지 않았다.

각 역량요소는 평가요소, 평가자료, 평가단계(점수표)로 구성하였다.

셋째, 1차 개발 루브릭을 3인의 전문가가 검토하였다. 평가 루브릭의 적절성을 평가하는 설문지를 개발하였다. 설문지는 1) 영역 구성과 정의의 적절성, 2) 역량요소의 구성의 적절성과 각 요소의 중요도, 3) 평가요소의 역량요소 평가의 적절성 등으로 구성하였다. 전문가는 5점 척도로 각 문항에 답하였고 각 항목에 대해 서술형태로 제안하였다.

넷째, 전문가 응답 점수와 서술식 제언을 종합하여 최종 루브릭을 완성하였다. 적절성 점수가 가장 낮았던 ‘산출물을 심미적 기준으로 평가할 수 있는 능력’ 역량기준은 삭제하였다. 평가자가 이해하기 쉽게 내용 및 단어를 수정(예를 들어, 3번째 요소인 ‘산출물의 분석’이 ‘산출물의 자가 평가’로 변경)하였다.

### 4.2 적용

비전공자를 대상으로 하는 프로그래밍 수업의 프로젝트 산출물 평가에 개발된 루브릭을 사용하였다. 2015년 겨울과 2016년 여름에 실시된 두 번의 수업에 적용하였다.

이 수업은 비전공자 대상 프로그래밍 방학 특강으로 아이디어를 컴퓨팅하는 프로토타입 과정과 Swift 언어로 아이디어를 구현하는 2단계로 구성되었다. 학생들은 최종 산출물로 자유 주제의 애플리케이션을 개발하여 시연하였다. 2015년에는 개인별로 프로젝트를 진행하여 평가대상이 24명인 반면, 2016년에는 팀으로 프로젝트를 진행하여 평가대상이 7팀이었다.

<표 4> 수업 구성 및 산출물

|       | 2015년  | 2016년           |
|-------|--|-----------------|
| 수업 구성 | <ul style="list-style-type: none"> <li>• <b>프로토타입 과정</b>: 학생들의 아이디어를 앱으로 디자인</li> <li>• <b>Swift 과정</b>: 디자인한 아이디어를 Swift 언어를 사용하여 앱으로 구현</li> </ul> |                 |
| 산출물   | 애플리케이션 (개인별)(24명)  | 애플리케이션 (조별)(7팀) |

학생들의 산출물은 3인이 평가하였다. 학생들이 약 15분 동안 개발한 애플리케이션에 대해 설명하고 시연하는 동안 평가자가 루브릭에 따라 평가를 진행하였다. 15분 동안 특정 역량을 학생들이 보여주지 않는 경우 질문을 통해 해당 역량을 평가하였다.

### 4.3 검증

일반 평가도구와 마찬가지로 루브릭도 해당 영역을 얼마나 적절히 평가하는지(타당도)와 얼마나 일관성 있게 평가하는지(신뢰도)를 검증할 수 있다[32][33].

첫째, 타당도는 개발 과정에서 확보하였다. 내용 타당도는 측정 요소가 해당 영역을 측정하는 지에 대한 것이다. 적절한 준거틀인 CTP의 설계 패턴[26]을 근거로 루브릭을 개발하였고, 루브릭 개발 과정에서 전문가 검토를 진행하여 내용 타당도를 확보하였다.

둘째, 신뢰도는 루브릭을 사용하였을 때 평가자가 얼마나 일관(consistency)되게 평가하는 지에

대한 것으로, Cronbach's  $\alpha$ 로 계산하였다.

## 5. CT 평가 루브릭 개발

CTP 설계 패턴을 기반으로 전문가 검토를 거쳐 비전공자 프로그래밍 수업의 창의적 산출물 평가에 사용할 루브릭을 개발하였다.

루브릭은 <표 5>와 같이 영역, 역량요소, 평가요소, 평가자료, 평가단계로 구성하였다.

<표 5> 루브릭의 구성 요소

| 구성    | 설명                      |
|-------|-------------------------|
| 영역    | 평가하고자 하는 영역             |
| 역량 요소 | 해당 영역에서 평가하고자 하는 학생의 역량 |
| 평가 요소 | 해당 역량의 평가 기준            |
| 평가 자료 | 평가 근거로 사용될 자료           |
| 평가 단계 | 평가 단계를 3개로 나뉘 점수 스케일 제공 |

영역은 '추상화 모델의 설계', '창의적 산출물의 설계 및 적용', '산출물의 자가 평가'의 3가지로 구성하였다. 각 영역의 정의는 <표 6>과 같다.

<표 6> 루브릭의 영역과 정의

| 영역               | 정의  |
|------------------|---|
| 추상화 모델의 설계       | 학생들이 실체(entity)나 과정(process)에서 상위개념과 하위개념, 그리고 실체, 과정 간의 관계/구조를 나타내는 개념과 표현을 사용하는지 확인 |
| 창의적 산출물의 설계 및 적용 | 학생들이 아이디어나 문제 해결을 computing적 해결방법이나 산출물(컴퓨터 프로그램 등)로 변환할 수 있는지 확인                     |
| 산출물의 자가 평가       | 학생들이 자신의 산출물(computational artifacts)을 기준에 따라 스스로 평가할 수 있는지 확인                        |

개발된 루브릭은 <표 7>과 같다. 예를 들어, 첫 번째 영역인 '추상화 모델의 설계'에 포함된 역량에는 '문제/해결 방안을 구조화하여 생각하는 능력'이 있다. 이 역량은 '세부 구조에 어떤 요소들이 있는지 정의하고, 요소 간, 세부 구조 간 관계를 표현한 정도'로 평가할 수 있다(평가 요소). 이 평가요소는 ADS와 시연을 통해 평가할 수 있다(평가 자료). 평가를 위해 3단계로 구성되어 점수 스케일(1-3, 4-6, 7-10점)을 제공(평가 단계)하였다.

<표 7> 비전공자 프로그래밍 수업 산출물 CT 평가 루브릭

| 영역               | 역량 요소   | 평가 요소  | 1-3                        | 4-6                      | 7-10                   |
|------------------|---|--|----------------------------|--------------------------|------------------------|
| 추상화 모델의 설계       | 문제/해결방안을 구조화하여 생각하는 능력                          | 세부 구조에 어떤 요소들이 있는지 정의하고, 요소 간, 세부 구조 간 관계를 표현한 정도      | 요소 정의 및 관계 표현이 거의 되지 않음    | 어느 정도 요소 정의 및 관계 표현      | 적절하게 요소 정의 및 관계 표현     |
|                  | 문제/해결방안을 표현하는 추상화(모델)을 설계하는 능력                  | 요소와 관계가 추상화(모델)로서 적절하게 표현된 정도                          | 추상화(모델)이 잘 표현되지 않음         | 추상화(모델)을 어느 정도 표현        | 추상화(모델)이 적절하게 표현       |
| 창의적 산출물의 설계 및 적용 | 문제 해결방안의 초기상태와 최종상태를 진술하는 능력                    | 초기상태와 최종상태가 정확하게 정의되었는지 정도                             | 거의 정의되지 않음                 | 어느 정도 정의됨                | 정확하게 정의됨               |
|                  | 문제를 해결 가능한 작은 단위로 나누는 능력                        | 문제 해결을 위해 문제를 분해(모듈화, 계층화)했는지 여부와 적절성                  | 문제 분해가 이루어지지 않음            | 문제 분해됨, 부적절              | 적절하게 문제 분해됨            |
|                  | 작은 단위의 문제를 조합하여 최종 해결 방안을 구성하는 능력               | 분해된 문제를 재조합(체계적 구조화)했는지 여부와 재조합된 해결방안의 적절성             | 문제 분해가 되지 않았거나 재조합되지 않음    | 재조합 됨, 부적절               | 재조합된 해결방안이 적절함         |
|                  | 프로그래밍 요소를 사용하는 능력                               | 산출물에서 프로그래밍 요소(구조적/재제지향적, 변수, 조건문, 논리, 반복, 함수)의 적절한 사용 | 프로그래밍 요소가 적절하게 사용되지 않음     | 프로그래밍 요소가 어느 정도 적절하게 사용됨 | 프로그래밍 요소가 매우 적절하게 사용됨  |
|                  | 테스트와 디버깅을 통해 산출물을 검토하고 수정하는 능력                  | 산출물의 정확성(입력값에 대한 정확한 출력값 도출, 에러가 발생 여부)                | 입력값에 대한 출력값이 부적절, 다수 에러 발생 | 입력값에 대한 정확한 출력값, 에러 발생   | 입력값에 대한 정확한 출력값, 에러 없음 |
| 산출물의 자가 평가       | 산출물의 목표 혹은 결과에 대해 설명하는 능력                       | 산출물의 목표 혹은 결과를 명확하게 설명하는 정도                            | 설명을 못하거나 부적절               | 설명이 명확하지 않음              | 설명이 명확함                |
|                  | 산출물이 의도한 사용자의 요구를 만족시켰는지 평가하는 능력                | 의도한 사용자의 요구를 만족시킨 정도                                   | 사용자 요구 분석안됨, 혹은 30%미만 만족   | 30%-70% 미만 만족            | 70%이상 만족               |
|                  | 최적화를 통해 산출물의 효율성(메모리 사용, 속도 등) 향상이 가능함을 인지하는 능력 | 효율성 향상에 대해 설명하는 정도                                     | 효율성 향상 필요 거의 인식 못함         | 1개의 영역에 대한 효율성 향상 적절히 언급 | 2개 이상 영역에 대한 향상 적절히 언급 |

## 6. 적용 결과

<표 8> 점수 결과

### 6.1 채점 결과

|    | 평균   | 표준편차 | 최고점 | 최저점 |
|----|------|------|-----|-----|
| 전체 | 77.6 | 7.6  | 95  | 57  |

비전공자 대상 프로그래밍 수업에 개발된 루브릭을 적용하여 학생들의 창의적 산출물에 드러난 CT 요소를 평가하였다. 2015년 겨울과 2016년 여름 학기 2차례에 걸쳐 총 33개의 산출물을 평가하였다. 프로젝트 운영 방식이 개인에서 팀으로 변경되어 2015년 겨울에는 24개의 산출물을 평가하였지만, 2016년 여름에는 7개의 산출물을 평가하였다.

3명의 평가자는 루브릭을 적용하여 10개 항목(역량 요소)을 각 각 10점으로 하여 점수를 합산하였다. 3명의 평가자 점수를 평균하여 낸 점수를 종합해 보면 <표 8>과 같다.

앞서 언급하였듯 2015년 수업은 개인별 프로젝트로, 2016년에는 팀별 프로젝트로 진행되었다. 산출물 개발 단위(개인별, 팀별)에 따라 루브릭 항목별 점수 평균이 차이가 있는지를 t-test로 검사해 보았다. 그 결과 항목별 평균 점수 차이도 최대 0.9에서 최하 0.1로 크지 않을뿐더러, 유의미하지 않았다(<표 9>). 따라서 본 루브릭을 통한 항목별 평가 점수는 개인별 산출물이든, 팀별 산출물이든 산출물 개발 단위에 따라 유의미한 차이가 있다고 볼 수 없다.

<표 9> 산출물 개발 단위별(개인, 팀) 루브릭 문항 점수 평균 차 t-검증 결과

|      | 1.1* | 1.2  | 2.1  | 2.2  | 2.3  | 2.4  | 2.5  | 3.1  | 3.2  | 3.3  |
|------|------|------|------|------|------|------|------|------|------|------|
| 2015 | 8.2  | 8.2  | 8.5  | 8.2  | 8.0  | 7.5  | 7.6  | 8.3  | 7.3  | 6.1  |
| 2016 | 7.5  | 8.0  | 7.8  | 7.9  | 7.6  | 7.6  | 7.0  | 7.6  | 6.4  | 5.2  |
| 차이   | 0.7  | 0.2  | 0.7  | 0.3  | 0.4  | 0.1  | 0.6  | 0.7  | 0.9  | 0.9  |
| t값   | 2.18 | 0.59 | 1.68 | 1.00 | 1.37 | -0.3 | 1.83 | 1.25 | 1.76 | 2.11 |
| df   | 8.8  | 10.6 | 7.27 | 10.1 | 14.6 | 15.8 | 11.6 | 7.37 | 8.02 | 7.46 |
| p값   | 0.06 | 0.56 | 0.13 | 0.33 | 0.19 | 0.76 | 0.09 | 0.24 | 0.11 | 0.07 |

\*1.1은 1번 영역의 1번째 평가요소를 의미. 동일한 방법으로 평가요소 명시하였음

## 6.2 신뢰도 결과

루브릭의 신뢰도는 일반적으로 평가자간 신뢰도로 검증한다[32][33][34]. 평가자간 신뢰도는 여러 평가자들의 의견의 일관성(consistency)으로 확인한다. 루브릭을 사용하여 평가하였을 때, 루브릭 없이 평가하였을 때보다 한 평가자가 일관되게 평가할 수 있어야 루브릭을 사용한 평가가 의미가 있다고 볼 수 있다.[33].

일관성은 Cronbach's  $\alpha$ 로 계산하였다. Cronbach's  $\alpha$ 는 문항의 내적일관성을 알아보는 방법으로 0-1까지 범위를 가지며 1에 가까울수록 일관성 높다[33]. Cronbach's  $\alpha$ 는 루브릭 개발 논문에서 일관성 검증에 주로 사용된다[32]. 본 루브릭의 Cronbach's  $\alpha$ 값은 0.74로 신뢰할만하다는 결과가 나왔다(<표 10>).

<표 10> 일관성 결과 (Cronbach's  $\alpha$ )

|      | raw $\alpha$ | std. $\alpha$ | 평균상관 계수 | 평균 | 표준편차 |
|------|--------------|---------------|---------|----|------|
| 전체   | 0.74         | 0.79          | 0.55    | 78 | 7.6  |
| 2015 | 0.71         | 0.83          | 0.61    | 78 | 7.2  |
| 2016 | 0.84         | 0.84          | 0.64    | 77 | 9.4  |

## 7. 결론 및 제언

본 연구는 비전공자 대상 프로그래밍 수업에서 학생들의 프로젝트 결과물(창의적 산출물)을 CT 기반으로 평가하는 루브릭 개발을 목적으로 진행하였다. 그 결과로 3개의 영역('추상화 모델의 설

계', '창의적 산출물의 설계 및 적용', '산출물의 자가 평가')과 10개의 역량 요소로 구성된 루브릭을 개발하였다. 루브릭은 CT를 수행(practice) 관점에서 볼 때 적절하고 중요하다고 판단된 3가지 영역, 해당 영역에서 학생들이 보유해야 할 10가지 역량 요소, 각 역량 요소를 평가하기 위한 평가 요소, 실제 평가에서 어떤 자료를 근거로 판단할 수 있을지 알려주는 평가 자료, 점수 스케일을 제공하는 평가 단계의 5가지로 구성하였다.

루브릭의 타당도와 신뢰도를 검증하기 위해 개발 과정에서 타당도를 확보하고, 사후 점수를 통계처리하여 신뢰도를 검증하였다. CT를 수행(practice) 관점에서 평가하는 준거틀(CTP 설계 모델)을 기반으로 루브릭을 초안을 개발하였다. 1차 초안에 대한 전문가 검토를 더하여 타당도를 확보하였다. 신뢰도는 평가자간 (inter-rater) 신뢰도의 일관성으로 검증하였다. Cronbach's  $\alpha$ 를 사용하여 계산하였다. 일관성은 0.7이상으로 신뢰할만하다는 결과가 나왔다.

본 루브릭은 비전공자 프로그래밍 교육 현장에 맞게 제작되었다. 다른 교육 과정에서 프로젝트 형식의 결과물을 CT를 기반하여 평가하고자 하는 경우, 각 수업에 맞게 본 루브릭을 수정하여 사용할 필요가 있다. 평가자가 루브릭을 충분히 이해할 때, 신뢰도가 향상된다는 기존 연구 결과[32][33]를 고려하여, 평가 전에 적절한 교육을 제공하는 것도 중요하다.

본 연구 범위에는 포함되지 않지만 루브릭의 평가 도구로서의 가치 뿐 아니라 교수학습 도구로서의 가치 또한 간과해서는 안된다. 이미 여러 문헌에서도 학생들의 동기와 학업성취 부분에서 루브릭이 긍정적 영향을 미치고 있음을 밝히고 있다[32][34][35]. 교수학습도구로서도 본 루브릭이 사용되고, 그 영향에 대해 후속 연구가 진행될 것 기대해본다.

## 참고 문헌

- [1] 우호성, 김자미, 이원규 (2017). 해외 고등정보 표준교육과정 기반의 국내 대학 교육과정 비교 분석, 한국컴퓨터교육학회 논문지, 20(1), 27-38.



- [2] 김민자, 김현철 (2016). 정보 교과 경험 차이에 따른 대학 전공 선택 및 SW수업 학업 성취 분석, **한국컴퓨터교육학회 논문지**, 19(3), 1-9.
- [3] 김자미, 이원규 (2014). 영국의 교육과정 개정으로 본 정보교과의 지식과 문제해결력에 대한 쟁점, **한국컴퓨터교육학회 논문지**, 17(3), 53-62.
- [4] Department for Education (2014). *The national curriculum in England: Framework document*, Department for Education.
- [5] 김한성, 정혜진, 이원규 (2010). 한국과 일본의 고등학교 정보교과 교육과정 비교 연구, **비교교육연구**, 20(4), 129-151.
- [6] 김자미, 이원규 (2014). 브루너 이론에 근거한 인도의 정보교육과정 고찰, **한국컴퓨터교육학회**, 17(6), 59-69.
- [7] 김경훈, 이은경, 김영애, 양재명, 이영준, 김현철, 김재현, 배정아, 한건우, 박소영, 박종훈 (2015). **정보과 교육과정 시안 개발 연구** (연구보고 CRC 2015-17). 서울: 한국교육과정평가원.
- [8] Malan, D. (2017년 1월 5일). CS50. Retrieved from <http://cs50.harvard.edu>
- [9] 임현석 (2014년 12월 12일). [세종대학교] 인문-과학 통섭교양과목·토론식 수업으로 창의인재 육성, 동아일보, 출처: <http://news.donga.com>
- [10] National Science Foundation (2017, January 5). *Transforming Undergraduate Education in Science, Technology, Engineering and Mathematics*. Retrieved from <http://nsf.gov>
- [11] 미래창조과학부, 정보통신산업진흥원 (2016년 11월 17일). **소프트웨어 중심대학**. 소프트웨어 중심사회. 출처: <https://software.kr>
- [12] Qualls, J. & Sherrell, L. (2010). Why Computational Thinking should be integrated into the curriculum, *Journal of Computing Sciences in Colleges*, 25(5), 66-71.
- [13] Committee for the Workshops on Computational Thinking; National Research Council (2011). *Report of a workshop of pedagogical aspects of Computational Thinking*, Washington, D.C.: The National Academies Press.
- [14] Committee for the Workshops on Computational Thinking; National Research Council (2010). *Report of a workshop on the scope and nature of Computational Thinking*. Washington, D.C.: The National Academies Press.
- [15] Wing, J. (2006). Computational Thinking, *Communications of the ACM*, Viewpoint, 49(3), 33-35.
- [16] Wing, J. (2008). Computational Thinking and thinking about computing, *Philosophical transactions of the Royal Society A*, 366, 3717-3725.
- [17] Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., Malyn-Smith, J., & Werner., L. (2010). Computational thinking for youth in practice, *ACM Inroads*, 2(1), 33-37.
- [18] Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: What is involved and what is the role of the computer science education community?, *ACM Inroads*, 2(1), 48-54.
- [19] CSTA & ISTE (2011). *Computational Thinking in K-12 education - teacher resources*, 2nd edition, CSTA & ISTE.
- [20] Computing at School Working Group (2012). *Computer Science: A curriculum for schools*, Computing at School.
- [21] 이영준, 백성혜, 신재홍, 유헌창, 정인기, 안상진, 최정원, 전성균 (2014). **초중등 단계 Computational Thinking 도입을 위한 기초 연구**, 한국과학창의재단.
- [22] Denning, P. & Martell, C. (2015). *Great principles of computing*, The MIT Press.
- [23] 성태제 (2014). **문항제작 및 분석의 이론과 실제 (개정판)**. 서울: 학지사.
- [24] Josson, A. & Svingby, G. (2007). The use of scoring rubrics: Reliability, validity and educational consequences, *Educational*

- Research Review*, 2(2007), 130-144.
- [25] Reddy, Y. & Andrade, H. (2010). A review of rubric use in higher education, *Assessment & Evaluation in Higher Education*, 35(4), 435-448.
- [26] Bienkowski, M., Snow, E., Rutstein, D., & Grover, S. (2015). *Assessment design patterns for computational thinking practices in secondary computer science: A first look* (SRI technical report). Menlo Park, CA: SRI International.
- [27] 최형신 (2014). 로봇활용교육의 효과성 증진을 위한 평가도구 개발: 사회·문화적 맥락 및 컴퓨팅 사고 연계, **한국정보교육학회 논문지**, 18(4), 541-548.
- [28] Brennan, K. & Resnick, M. (2012). *New frameworks for studying and assessing the development of computational thinking*, Paper presented at annual American Educational Research Association meeting, Vancouver, BC, Canada.
- [29] 최형신 (2014). Computational Thinking 역량 개발을 위한 수업 설계 및 평가 루브릭 개발, **한국정보교육학회 논문지**, 18(1), 57-64.
- [30] Bort, H. & Brylow, D. (2013). CS4Impact: Measuring Computational Thinking concepts present in CS4HS participants lesson plans, *Proceedings in SIGCSE 2013*, March 6-9, Denver, Colorado, USA.
- [31] Sherman, M. & Martin, F. (2015). The assessment of mobile computational thinking, *Journal of Computing Sciences in Colleges*, 30(6), 53-59.
- [32] Josson, A. & Svingby, G. (2007). The use of scoring rubrics: Reliability, validity and educational consequences, *Educational Research Review*, 2(2007), 130-144.
- [33] Moskal, B. & Leydens, K. (2000). Scoring rubric development: Validity and reliability, *Practical Assessment, Research & Education*, 7(10), Retrieved September 19, 2016 from <http://PAREonline.net/getvn.asp?v=7&n=10>.
- [34] Reddy, Y. & Andrade, H. (2010). A review of rubric use in higher education, *Assessment & Evaluation in Higher Education*, 35(4), 435-448.
- [35]한건우, 이은경, 이재희, 이영준 (2006). 학습자 중심 루브릭을 적용한 수행평가가 학습동기와 학업성취도에 미치는 영향, **한국컴퓨터교육학회 논문지**, 9(4), 1-8.



## 김민자

2012~현재  
고려대학교 컴퓨터교육과 박사수료  
관심분야: 컴퓨터교육  
E-Mail: minjaya@korea.ac.kr



## 유길상

2011~현재 고려대학교 연구교수  
2010~현재 (사)한국컴퓨터게임학회  
이사  
2012 (사)중앙스마트학교재단  
이사

관심분야: 컴퓨터교육, 게임기반학습, 디지털콘텐츠,  
디지털영상처리

E-Mail: phd.yoo@gmail.com



## 김현철

1988 고려대학교  
전산과학과(이학사)  
1990 Univ. of Missouri-Rolla  
(전산학 석사)

1998 Univ. of Florida (전산정보학박사)

1999~현재 고려대학교 사범대학 컴퓨터교육과 교수

2014~현재 고려대학교 정보대학 컴퓨터학과 교수

관심분야: 컴퓨터교육, 기계학습

E-Mail: harrykim@korea.ac.kr