

# 초·중학교 SW교육을 위한 프로그래밍 평가지표 개발

안성훈<sup>†</sup>

## 요 약

본 논문에서는 2016년부터 초·중학교에서 확대·강화되는 SW 교육에서 학생들이 개발한 SW 프로그램을 창의력, 논리력, 문제해결력 등과 같은 교육적 가치 측면과 효율성, 신뢰성, 완전성 등과 같은 SW 품질적 측면이 조화롭게 평가될 수 있는 평가지표를 개발하고자 하였다. 이를 위해서 먼저 SW교육의 단계별로 교육적 가치 측면과 SW 품질적 측면에서의 15개의 SW 프로그램 평가요소를 설정하고 각 평가요소별 총 36개의 평가기준을 개발하였다. 그리고 15개 평가요소와 36개의 평가기준에 대하여 SW교육 및 프로그래밍 전문가 39명에게 평가요소와 평가기준의 타당성을 검토 받았으며, 그 결과, 15개 평가요소와 36개 평가기준 모두 타당성이 양호한 것으로 나타났다. 본 논문에서 개발된 SW 프로그램 평가지표는 SW교육의 신뢰성을 높여주고 학생들에게 자신의 학습에 대한 자아성찰의 기회를 제공해 주며, 정보영재교육과 SW 프로그램 공모전 및 경시대회, SW 능력 인증 및 자격제도 등의 다양한 교육활동 활성화에 기여할 것으로 기대된다.

주제어 : SW 프로그램, 평가지표, 교육적 가치 측면, 품질적 측면

## Development of SW Program Assessment Indicator for SW Education in Elementary and Middle School

Seonghun Ahn<sup>†</sup>

### ABSTRACT

In this paper, I developed a assesment indicator to evaluate SW program developed by students in SW education. The SW education carry out in elementary school, middle school and high school from 2016. But the assesment indicator to evaluate SW program had not been developed until now. Therefore, I tried to develop the assesment indicator that have harmoniously educational characteristic and SW qualitative characteristic. First of all, I had set the 15 attributes for assesment and developed the 36 criteria for 15 attributes. And the validity of 15 attributes and 36 criteria for assesment had been reviewed by 39 SW educational experts. As a result, the 15 attributes and 36 criteria for assesment were determined to which that have validity. I expect that the assesment indicator developed in this paper will raise the reliability of evaluation in SW education. And the assesment indicator may be used at SW competition conference for teenagers.

**Keywords** : SW Program, Assesmen Indicator, Educational Characteristic, Qualitative Characteristic

---

<sup>†</sup> 정 회 원: 경인교육대학교 컴퓨터교육과 교수(교신저자)

논문접수: 2016년 5월 20일, 심사완료: 2016년 7월 4일, 게재확정: 2016년 7월 17일

\* 이 논문은 2015년 대한민국 교육부와 한국연구재단의 지원을 받아 수행된 연구임 (NRF-2015S1A5A8018317)

## 1. 서 론

2014년 7월 정부는 「소프트웨어 중심 사회 실현」을 위한 전략 보고대회를 개최하고, 각 부처별 SW 중심사회 실현을 위한 정책 방향을 발표하였는데 여기에서 교육부는 창조경제 시대에 필요한 창의적 사고력과 문제해결력을 갖춘 인재 양성 기반 조성을 위해 어릴 때부터 학교에서 SW교육을 체계적으로 배울 수 있는 교육 기회를 강조하고 이를 추진하기 위한 「초·중등 SW교육 활성화 방안」을 발표하였다[1].

교육부의 「초·중등 SW교육 활성화 방안」에 따르면 SW교육을 위해 초등학교에서는 정보관련 교과 내용을 SW 기초 소양 교육 내용으로 개편하고, 중학교에서는 정보관련 교과 내용 개편 및 ‘정보’교과를 ‘SW’교과로 전환하며, 고등학교에서 ‘정보’교과를 심화선택에서 ‘SW’교과 일반선택으로 전환하도록 하였다.

그리고 교육부는 2014년 9월 「2015 교육과정 총론」을 통해 SW교육 필수 시간을 초등학교는 17시간 이상, 중학교는 34시간 이상을 각각 확보해 운영하는 방안을 발표하였으며[2], 2015년 2월에는 이 방안을 추진하기 위한 「SW교육 운영 지침」을 마련하여 17개 시·도교육청을 통해 일선 초·중등학교에 제시하였다[3].

교육부가 제시한 「SW교육 운영 지침」에 따르면 초·중등학교에서 SW교육은 일상생활에서 다루어지는 SW의 종류와 특징을 살펴보는 생활과 소프트웨어 영역, 정보를 디지털화하여 논리적으로 구조화하고 자동화시키는 알고리즘과 프로그래밍 영역, 실생활에서 직면하는 문제들을 해결하기 위한 알고리즘을 설계하고, 이를 프로그램으로 구현하는 컴퓨팅과 문제해결 영역으로 구성되어 있다[3]. 그리고 SW교육에서의 평가는 교육 목표의 달성 여부를 정확하게 판단할 수 있는 체계적인 평가 계획을 수립하여 운영하되 타당한 평가 도구를 활용하여 소프트웨어 교육을 통한 컴퓨팅 사고력의 향상 정도를 지속적으로 측정하며, 지식 위주의 평가를 지양하고, 수행 평가의 특징을 최대한 반영하여 평가할 수 있도록 필요한 평가 자료를 개발·활용하도록 하고 있다.

따라서 향후 SW교육에서는 올바른 평가를 위

해서 수행형 평가도구의 개발이 절실히 필요하며, 특히 SW교육의 최종 산출물인 학생들이 개발한 SW 프로그램을 올바르게 평가하기 위해서는 창의력, 논리력, 문제해결력 등과 같은 교육학적 측면의 가치와 효율성, 신뢰성, 완전성 등과 같은 SW 공학적 측면의 가치를 종합적으로 측정할 수 있는 평가 도구의 개발이 필요하다.

지금까지 개발된 초·중등학교 학생들의 SW 프로그램에 대한 평가 도구의 적용 사례를 살펴보면, 학생들이 작성한 SW 프로그램 소스(source)를 시스템을 통해 자동으로 평가하는 방식과 평가지표에 따라 평가자가 수동으로 평가하는 방식이 있다.

그중 자동으로 평가하는 방식의 대표적인 사례로 Online Judge 시스템을 살펴볼 수 있는데 Online Judge 시스템은 C, C++, Java, Pascal, Fortran 등의 언어로 개발된 프로그램을 자동으로 평가할 수 있어 우리나라를 비롯해, 미국, 중국, 러시아, 스페인, 폴란드 등 여러 국가에서 사용하고 있다. 그러나 주로 SW 공학적 측면에서 SW 프로그램의 효율성, 신뢰성, 완전성 등을 중점적으로 평가하고 있어 창의력, 논리력, 문제해결력 등은 교육학적 가치는 평가하기 어렵고, 또한, 최근 초·중등학교에서 SW 프로그램 교육에 많이 사용하고 있는 스크래치, 엘리스, 엔트리 등으로 작성된 프로그램은 평가할 수 없다는 단점이 있다[4][5].

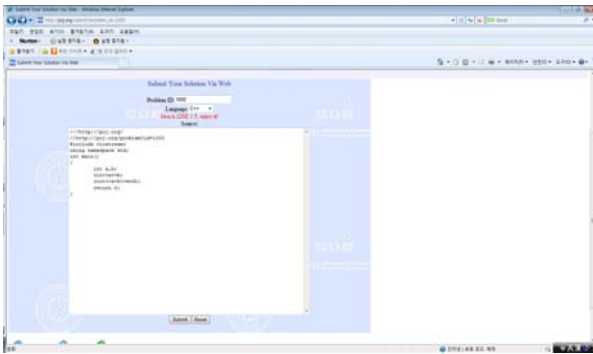
수동으로 평가하는 방식의 대표적인 사례로 미래창조과학부에서 주최하는 한국정보올림피아드 대회와 공개 SW 개발자 대회를 살펴볼 수 있는데 이들 대회에서는 학생들이 개발한 프로그램을 교육학적인 측면에서의 가치로 요구되는 창의성과 논리력 그리고 SW 공학적 측면에서 요구되는 효율성을 종합해 평가하고 있으나 평가지표의 구성이 단순히 초·중등학교의 SW교육 평가 도구로 활용하기에는 어려운 점이 많이 있다[6][7].

이에 따라 본 연구에서는 일선 초·중등학교의 SW교육에서 이루어지는 SW 프로그래밍 교육 평가에 적합하도록 창의력, 논리력, 문제해결력 등과 같은 교육학적 측면의 가치와 효율성, 신뢰성, 완전성 등과 같은 SW 공학적 측면의 가치가 조화롭게 평가될 수 있는 평가지표를 개발하고자 한다.

## 2. SW 프로그램 평가 사례 분석

### 2.1 SW 프로그램 자동 평가 사례 분석

초·중등학교에서 SW교육을 통해 학생들이 개발한 프로그램을 자동으로 평가하는 방식의 대표적인 사례로 Online Judge 시스템을 살펴볼 수 있다.



[그림 1] Online Judge 시스템[8]

Online Judge의 가장 큰 특성은 온라인에서 학생들이 작성한 프로그램 소스 코드를 입력하고 입력된 프로그램을 자동으로 평가할 수 있다는 점이다. 사용자들이 작성하여 프로그램 소스 코드를 온라인상으로 제출하면 Online Judge 시스템에서는 파일을 먼저 저장한 후 컴파일하여 실행 가능한 프로그램 형태로 가공한다. 가공된 실행 프로그램에 입력 데이터와 출력 데이터로 구성된 테스트 데이터 세트를 만들어 적용하여 프로그램의 신뢰성을 평가한다. 즉, 테스트 데이터 세트의 입력 데이터를 실행 프로그램에 입력하여 얻은 출력 값과 테스트 데이터 세트의 출력 데이터가 같고 시간 제한과 메모리 제한을 위반하지 않았다면 해당 데이터에 대해서는 프로그램이 제대로 동작한 것으로 평가되어진다[9].

이러한 Online Judge의 기능은 사용자들은 온라인상으로 언제 어디서나 자신의 소스 코드를 평가할 수 있고, 빠른 시간 내에 결과를 확인할 수 있기 때문에 프로그래밍을 학습하는 과정에서 매우 중요한 의미를 지닌다[9][10]. 또한, 교사들은 소스 코드를 평가하는 업무를 대폭 경감시킬 수 있으며 이 시간을 이용하여 학생들의 질문에 답해주는 등의 활동이 가능해진다[9].

최근 Online Judge 시스템은 C, C++, Java,

Pascal, Fortran 등의 언어로 개발된 프로그램을 자동으로 평가할 수 있어 우리나라를 비롯해, 미국, 중국, 러시아, 스페인, 폴란드 등 여러 국가에서 사용하고 있다.

### 2.2 SW 프로그램 수동 평가 사례 분석

#### 2.2.1 한국정보올림피아드 SW 프로그램 평가지표

미래창조과학부에서 초·중등 학생을 대상으로 실시하는 한국정보올림피아드에서 학생들이 작성한 SW 프로그램을 평가하는 평가지표는 아래 표 1과 같이 창의성, 과학적 사고/기술적 목적, 완벽성, 기술성, 명확성, 협동성으로 구성되어 있다[6].

<표 1> 한국정보올림피아드 SW 프로그램 평가 기준[6]

평가 항목	평가 내용
창의성	<ul style="list-style-type: none"> <li>- 과학적 착상(작품 아이디어)의 독창성</li> <li>- 문제해결을 위한 접근방법, 데이터 분석 및 해석의 디자인과 구조</li> <li>- 연구주제(연구과제), 조사 및 연구방법 및 결과데이터 등이 과학적이고 창의적인가?</li> <li>- 학문적, 사회적, 과학적으로 가치가 있는가?</li> </ul>
과학적 사고/기술적 목적	<ul style="list-style-type: none"> <li>- 해결과제가 명확하며, 결과도출 후 제언이 과학적으로 적절한가?</li> <li>- 변인설정이 명확하고, 정확한가?</li> <li>- 학생들이 관련 연구·이론에 대해 정확히 이해하고, 각 관련 영역을 탐구하였는가?</li> <li>- 작품이 실현가능하고, 경제적 가치가 있는가?</li> <li>- 작품의 디자인 또는 구조가 유용한가?</li> <li>- 관련된 유사작품에 비해 획기적으로 향상된 점이 있는가?</li> <li>- 다양한 상황에 대한 충분한 테스트가 이루어졌는가?</li> </ul>
완벽성	<ul style="list-style-type: none"> <li>- 해결과제에 대한 결과가 정확하고, 명확한가?</li> <li>- 여러 번의 연구과정을 거쳐 나온 결과인가?</li> <li>- 오류사항에 대한 충분한 인지 및 고려를 하였는가?</li> <li>- 관련된 기존 연구에 대한 이론적 조사가 충분한가?</li> <li>- 결과의 도출된 내용이 충분한 신뢰성을 가지는가?</li> </ul>
기술성	<ul style="list-style-type: none"> <li>- 데이터 수집을 위한 적절한 절차가 있었는가?</li> <li>- 지도교사의 지도 및 전문가 지도(멘토)가 적절히 이루어졌는가?</li> <li>- 학생들이 주제·가설설정, 데이터수집, 결과도출 등 전 과정에 주도적으로 참여하였는가?</li> </ul>
명확성	<ul style="list-style-type: none"> <li>- 학생들이 해결과제(목적), 절차, 결론 및 제언에 대해 정확히 이해하고 있는가?</li> <li>- 수집되거나 도출된 데이터에 조작 등의 문제는 없는가?</li> <li>- 데이터 및 결과에 대한 과학적 설명이 유의미한가?</li> <li>- 학생이 수행할 수 있는 과제인가?</li> </ul>
협동성	<ul style="list-style-type: none"> <li>- 작품 제작과 출품과정에서 팀원들의 협동여부 및 참여정도는 어떠한가?</li> <li>- 출품자들의 역할분담, 최종작품에 각 팀원들의 공헌도 및 이해도 등은 적절한가?</li> </ul>

### 2.2.2 공개SW 개발자대회

미래창조과학부에서 초중등 학생 및 일반 국민들을 대상으로 실시하는 공개SW 개발자대회에서 응모자가 제출한 SW 프로그램을 평가하는 평가지표는 아래 <표 2>와 같이 코딩의 적절성 및 구조의 합리성, 코드의 완성도 및 시연 가능성, 개발 문서의 구체성, 프로젝트 공개 여부로 구성되어 있다[7].

<표 2> 공개SW 개발자 대회 SW 프로그램 평가 기준[7]

평가 기준	점수
코딩의 적절성 및 구조의 합리성	5
코드의 완성도 및 시연 가능성	5
개발 문서의 구체성	5
프로젝트 공개 여부	5
계	20

### 2.2.3 Computational Thinking 평가 루브릭

최형신(2014)은 최근 SW교육에서 주목받고 있는 Computational Thinking(이하 CT) 역량을 중심으로 학생들의 수업활동을 평가하기 위한 루브릭을 <표 3>과 같이 개발하였다. 최형신(2014)은 CT 세부 요소를 절차 및 알고리즘, 병행화 및 동기화, 자료 표현, 추상화, 문제 분해, 시뮬레이션으로 정의하고 각 요소에 대해 기초(Basic), 발달(Developing), 능숙(Proficient)의 3단계로 학생들의 CT 역량을 평가하도록 하였다[11].

<표 3> Computational Thinking 평가 루브릭[11]

CT 세부 요소	평가 내용
절차 및 알고리즘	프로그램의 절차 및 알고리즘의 효율성
병행화 및 동기화	동시적 처리나 객체간의 메시지 교환 등의 동기화 처리의 적절성
자료 표현	변수나 리스트 활용의 적절성
추상화	전체 프로그램에서 분해된 모든 논리적 단위의 새 블록 구현 여부
문제 분해	알고리즘의 주요 부분에 대한 논리적 단위 처리 여부
시뮬레이션	새 블록에 대한 파라미터(Parameter) 사용의 적절성

※ 최형신(2014)의 Computational Thinking 평가 루브릭을 일부 재구성함

### 2.2.4 시사점 분석

SW 프로그램 자동 평가 사례로 Online Judge 시스템의 사례를 분석한 결과, Online Judge 시스템은 온라인상으로 언제 어디서나 자신의 소스 코드를 빠른 시간 내에 평가할 수 있고, 교사들은 소스 코드를 평가하는 업무를 대폭 경감시킬 수 있다는 장점을 가지고 있으나 SW 프로그램을 개발 결과 중심으로 평가하고 있어 SW 프로그램을 설계하고 개발하는 과정에서 중요시되는 교육적인 가치의 평가가 소홀한 단점이 있다.

Online Judge 시스템은 테스트 데이터 세트를 통해 입력 값에 대한 출력 값이 정확한지, 정해진 시간 내에 결과가 출력되는지, 프로그램 동작에 필요한 메모리가 제한된 범위를 넘어서지 않는지 등 주로 SW 공학적 측면에서 SW 프로그램의 효율성, 신뢰성, 완전성 등을 중점적으로 평가하고 있다. 따라서 프로그램 개발을 통해 궁극적으로 신장하려는 창의력, 논리력, 문제해결력 등의 교육학적 가치는 평가하기 어렵다. 또한, 최근 초·중등학교에서 SW 프로그램 교육에 많이 사용하고 있는 블록형 프로그래밍 도구(스크래치, 앨리스, 엔트리 등)로 작성된 프로그램은 평가할 수 없다는 단점을 가지고 있다.

SW 프로그램 수동 평가 사례로 한국정보올림피아드와 공개SW 개발자 대회 평가 기준과 최형신(2014)의 CT 평가 루브릭의 사례를 분석한 결과, 한국정보올림피아드의 평가 기준은 SW 프로그램 개발 과정에 대한 창의성, 과학적 사고/기술적 목적, 협동성과 개발 결과에 대한 완벽성, 기술성, 명확성을 평가하고 있어 교육적 가치 측면과 SW로서의 품질 측면이 모두 포괄된 것으로 분석된다. 그러나 각 평가 항목에 대한 평가 내용이 구체적이고 세부적이지 못하여 평가 결과에 대한 객관성을 담보하기는 어려울 것으로 판단된다.

공개SW 개발자 대회의 평가 기준은 SW 프로그램 개발 결과를 중심으로 평가하고 있어 개발 과정에 대한 교육적 가치 측면이 소홀한 단점을 가지고 있다.

최형신(2014)의 CT 평가 루브릭은 CT 역량 개발 수업에 따른 학생들의 CT 역량을 평가하기

위한 것으로 SW 프로그램 개발 과정에 대한 교육적 가치의 측면을 충분히 포괄하고 있으나 SW 프로그램 개발 결과에 대한 품질적 측면에 대해서는 다루지 않고 있다.

이와 같은 내용에 비추어볼 때, 초중등 학생의 SW 프로그램 평가에 대해 현재 활용되고 있는 평가 사례들은 SW 프로그램 개발 과정에 대한 교육적 가치의 측면과 SW 프로그램 개발 결과에 대한 품질적 측면을 모두 충족하는 평가 기준이 부족한 것으로 나타났다.

결국 위와 같은 사례 분석의 내용을 종합해 볼 때, 초중학생의 프로그래밍 평가지표는 SW 프로그램 개발을 통하여 초중등 학생들이 궁극적으로 갖추어야 할 창의력, 논리력, 문제해결력 등의 교육학적 가치 측면과 개발된 프로그램이 SW로서 기본적으로 갖추어야 할 효율성, 신뢰성, 완전성, 기능성 등의 품질적 측면을 모두 균형있게 포괄하여야 한다.

### 3. 평가 도구 개발

#### 3.1 평가 영역 설정

초·중학생이 개발한 SW 프로그램에 대한 평가는 앞에서 살펴 본 SW 프로그램 평가 사례 분석에서 나타난 시사점과 같이 SW 프로그램 개발을 통하여 초·중 학생들이 궁극적으로 갖추어야 교육학적 가치 측면과 개발된 프로그램이 SW로서 기본적으로 갖추어야 할 품질적 측면을 균형있게 측정할 수 있는 평가가 필요하다.

따라서 SW 프로그램 개발 과정과 개발 결과물을 함께 평가할 수 있도록 평가 영역이 구성되어야 한다.

이를 위해서 본 논문에서는 교육부에 2016년부터 전국의 SW교육 연구학교에 배포하여 사용할 초·중학교 SW교육 교재를 분석하여 SW 프로그램 개발 과정을 문제 인식 및 정의(1단계), 문제 해결 방법 탐색(2단계), 프로그램 설계(3단계), 프로그램 구현(4단계) 등의 4단계로 구분하고 1·2·3단계는 교육적 가치 측면에서 4단계는 SW 품질적 측면에서 평가하도록 영역을 설정하였다.

<표 4> SW 프로그램 평가 영역 설정

측정 방향	평가 영역
교육적 가치 측면	문제 인식 및 정의
	문제 해결 방법 탐색
	프로그램 설계
SW 품질적 측면	프로그램 구현

#### 3.2 평가요소 및 기준 설정

##### 3.2.1 교육적 가치 측면

초·중학생들이 SW 프로그램을 개발하는 과정에서 궁극적으로 갖추어야 할 창의력, 논리력, 문제해결력 등이 어느 정도 형성되고 있는지 이들에 대한 하위 요소 측정이 필요하다.

최근 Wing(2006)은 컴퓨팅 사고력(Computational Thinking)이란 역량 정의를 통해 SW 프로그램 개발 형태와 같은 절차적이고 복잡화된 문제해결과정을 제안하였다. 그리고 미국의 CSTA(Computer Science Teachers Association)와 ISTE(International Society for Technology in Education), CB(Collage Board)와 NSF(National Science Foundation), Selby와 Wollard 등은 컴퓨팅 사고력의 세부 구성요소를 표와 같이 제시하였다[12].

<표 5> 컴퓨팅 사고력 세부 구성요소[12]

CSTA & ISTE	Selby & Wollard	CB & NSF
<ul style="list-style-type: none"> <li>자료 수집</li> <li>자료 분석</li> <li>자료 표현</li> <li>문제 분해</li> <li>추상화</li> <li>알고리즘과 절차</li> <li>자동화</li> <li>시뮬레이션</li> <li>병렬화</li> </ul>	<ul style="list-style-type: none"> <li>알고리즘적 사고</li> <li>분해</li> <li>일반화(패턴)</li> <li>추상화</li> <li>평가</li> </ul>	<ul style="list-style-type: none"> <li>컴퓨팅으로 연결</li> <li>컴퓨팅 산출물 개발</li> <li>추상화</li> <li>문제와 산출물 분석</li> <li>의사소통</li> <li>협동</li> </ul>

본 논문에서는 위와 같은 컴퓨팅 사고력 세부 구성요소를 앞에서 설정한 SW 프로그램 평가영역 중 문제 인식 및 정의, 문제 해결 방법 탐색, 프로그램 설계에 맞추어 평가요소를 설정하고 각 평가요소에 해당하는 평가기준을 <표6>과 같이 정의하였다.

<표 6> SW 프로그램 평가기준(교육적 가치 측면)

단계	평가 요소	평가 기준	
문제 인식 및 정의	문제 정의	문제 정의의 정확성	해결해야할 문제가 정확하게 정의되었는가?
	문제 분해	문제 조건의 명확성	문제 해결에 필요한 조건이 정확히 기술되었는가?
		문제 세분화의 적절성	문제 해결에 필요한 조건별로 문제를 세분화하였는가?
목표 정의	목표 진술의 명확성	문제해결을 위한 목표 정의가 명확한가?	
문제 해결 방법 탐색	자료 수집 및 분석	자료 수집의 적합성	문제 해결에 적합한 자료를 수집하였는가?
		자료 분석의 다각성	다양한 관점에서 자료를 분석하였는가?
		자료 분석 결과의 정확성	자료의 분석 결과가 정확한가?
	자료 표현	자료 정의의 명확성	문제 해결에 필요한 자료가 명확히 정의되었는가?
		자료 구조의 효율성	문제 해결에 필요한 자료의 구성 방법이 효율적인가?
	추상화	객체 정의의 적절성	문제 해결의 핵심 요소가 객체로 정의되었는가?
	객체 간 관계의 명확성	문제 해결 과정에서의 객체 간 관계가 명확하게 정의되었는가?	
독창성	문제 해결 방법의 독창성	문제 해결 방법이 독창적인가?	
프로그램 설계	알고리즘과 절차	논리성	프로그램 구성이 논리적인가?
		수행 효율성	프로그램의 수행이 효율적으로 이루어질 수 있는가?
		상호작용성	사용자 입력이 적절하게 이루어질 수 있는가?
		회귀성	프로그램의 초기화가 적절히 이루어질 수 있는가?
	병렬화	동시성	복수의 프로세스가 동시에 수행될 수 있는가?
		독립성	동시에 수행되는 프로세스가 서로 독립적인가?
	동기화	연계성	프로세스 간 연계가 가능한가?
		일치성	프로세스 간 연계 시점이 일치하는가?

3.2.2 SW 품질적 측면

초·중등학생이 개발한 SW 프로그램도 SW로써 동작하여 기본 목적을 수행하기 위해서는 최소한의 품질은 담보되어야 한다.

SW가 본래의 목적을 수행할 수 있도록 올바르게 동작할 수 있는지를 평가하기 위한 SW 품질요소를 국제표준 ISO/IEC 25010로 정하고 있다. ISO/IEC 25010에서는 표와 같이 SW 품질요소를 기능적합성(functional suitability), 성능효율성(performance efficiency), 호환성(compatibility),

사용성(usability), 신뢰성(reliability), 보안성(security), 유지보수성(maintainability) 및 이식성(portability) 등의 8가지 특성과 부특성을 제시하고 있다[13].

<표 7> SW 품질 요소(ISO/IEC 25010)[13]

품질특성	부특성
기능적합성	기능완전성, 기능정확성, 기능적절성
성능효율성	시간효율성, 자원사용성, 수용력
호환성	공존성, 상호운영성
사용성	적합인지성, 학습성, 운영성, 사용자 오류보호, 사용자 인터페이스, 호감성, 접근성
신뢰성	성숙성, 가용성, 결함허용성, 복구성
보안성	기밀성, 무결성, 부인방지, 추적성, 인증성
유지보수성	모듈화, 재사용, 분석성, 변경성, 시험성
이식성	적응성, 설치성, 대체성

본 논문에서는 위와 같은 SW 품질요소를 앞에서 설정한 SW 프로그램 평가영역 중 프로그램 구현에 맞추어 평가요소를 설정하고 각 평가요소에 해당하는 평가기준을 표와 같이 정의하였다

<표 8> SW 프로그램 평가기준(SW 품질적 측면)

단계	평가 요소	평가 기준	
프로그램 구현	기능화	기능완전성	계획한대로 프로그램의 기능이 모두 구현되었는가?
		기능정확성	계획한대로 프로그램의 기능이 정확하게 구현되었는가?
		기능적절성	전체적으로 구현된 프로그램의 기능이 적절한가?
	효율화	시간효율성	프로그램의 처리속도와 사용자 입력에 대한 반응속도가 적절한가?
		자원활용성	목적수행에 적합한 자원들이 적절하게 선택되어 사용되었는가?
		용량충족성	목적수행을 위해 활용된 자원의 양이 적절한가?
	일반화	모듈성	프로그램 구성이 모듈 단위로 이루어져 있는가?
		재사용성	프로그램의 각 모듈이 재사용하기 용이한가?
	신뢰성	성숙성	프로그램의 동작 결과가 정확한가?
		가용성	프로그램이 올바르게 동작하는가?
		복구성	오류를 방지하고 대처할 수 있는 기능이 있는가?

사용성	적정성 인식도	프로그램의 기능, 인터페이스, 메시지 등을 쉽게 이해할 수 있는가?
	학습성	프로그램의 기능이 사용하기 쉬운가?
	운영성	프로그램 사용 중 발생하는 요구사항을 반영하기 쉬운가?
	사용자 오류보호	프로그램 사용 중 발생할 수 있는 에러를 방지하기 위한 대책이 적절한가?
	사용자인터페이스	사용자 인터페이스 구성이 심미적인가?

프로그램 설계	알고리즘과 절차	4.9	0.2
	병렬화	4.1	0.9
	동기화	4.1	0.9
프로그램 구현	기능화	4.6	0.5
	효율화	4.7	0.5
	일반화	4.5	0.8
	신뢰성	4.4	0.8
	사용성	4.4	0.8

## 4. 평가 도구 검증

본 논문에서 설정된 초·중등학생의 SW 프로그램 개발 과정 및 결과를 평가하기 위한 평가요소와 평가기준의 내용이 타당한지를 확인하기 위하여 2016년 1월부터 2월까지 SW교육 및 프로그래밍 전문가 39명에게 타당성을 5점 Likert 척도(1점 매우 부적합 ↔ 5점 매우 적합)로 검토하였다. 그리고 그 결과를 이용하여 단계별 평가요소와 평가기준의 구성이 적절한지를 확인하기 위하여 확인적 요인분석을 실시하였다.

### 4.1 내용타당성

#### 4.1.1 단계별 평가 요소의 타당성

먼저 각 단계별 평가 요소의 내용타당성을 검토한 결과, 표와 같이 나타났다. 4개 단계 15개 평가 요소의 내용타당성 평균이 최소 4.1점에서 4.9점으로 모두 적합과 매우 적합 사이에 있는 것으로 나타났고, 표준편차도 최소 0.2점에서 최대 1.2점으로 15개 평가 요소 모두 내용타당성에 대한 의견 편차가 크지 않은 것으로 나타났다. 따라서 단계별 평가 요소의 내용타당성은 적합한 것으로 판정되었다.

<표 9> 단계별 SW 프로그램 평가 요소의 내용타당성

단계	평가 요소	평균	표준편차
문제 인식 및 정의	문제 정의	4.6	0.6
	문제 분해	4.2	1.2
	목표 정의	4.5	0.7
문제 해결 방법 탐색	자료 수집 및 분석	4.6	0.5
	자료 표현	4.6	0.5
	추상화	4.4	0.6
	독창성	4.1	0.9

#### 4.1.2 평가 요소별 평가 기준의 내용타당성

각 평가 요소별 평가 기준의 내용타당성을 검토한 결과, 표와 같이 나타났다. 4개 단계 15개 평가 요소 36개 평가 기준의 내용타당성 평균이 최소 4.1점에서 4.7점으로 모두 적합과 매우 적합 사이에 있는 것으로 나타났고, 표준편차도 최소 0.5점에서 최대 1.0점으로 36개 평가 기준 모두 내용타당성에 대한 의견 편차가 크지 않은 것으로 나타났다. 따라서 평가 요소별 평가 기준의 내용타당성은 적합한 것으로 판정되었다.

<표 10> 평가 요소별 SW 프로그램 평가 기준의 내용타당성

단계	평가 요소	평가 기준	평균	표준편차
문제 인식 및 정의	문제 정의	문제 정의의 정확성	4.5	0.7
		문제 조건의 명확성	4.6	0.7
		문제 세분화의 적정성	4.2	0.9
	목표 정의	목표 진술의 명확성	4.6	0.6
문제 해결 방법 탐색	자료 수집 및 분석	자료 수집의 적합성	4.6	0.6
		자료 분석의 다각성	4.4	0.7
		자료 분석 결과의 정확성	4.6	0.6
	자료 표현	자료 정의의 명확성	4.4	0.8
		자료 구조의 효율성	4.4	0.8
	추상화	객체 정의의 적절성	4.3	0.8
		객체 간 관계의 명확성	4.4	0.7
독창성	문제 해결 방법의 독창성	4.4	0.9	
프로그램 설계	알고리즘과 절차	논리성	4.7	0.6
		수행 효율성	4.7	0.5
		상호작용성	4.2	0.7
		회귀성	4.1	1.0
	병렬화	동시성	4.1	1.0
		독립성	4.2	1.0
	동기화	연계성	4.3	0.8
		일치성	4.1	0.8

<표 11> 평가 요소별 SW 프로그램 평가 기준의 내용타당성

단계	평가요소	평가 기준	평균	표준편차
프로그램 구현	기능화	기능완전성	4.7	0.7
		기능정확성	4.8	0.5
		기능적정성	4.4	0.7
	효율화	시간효율성	4.3	0.8
		자원활용성	4.4	0.8
		용량충족성	4.2	0.9
	일반화	모듈성	4.7	0.5
		재사용성	4.6	0.5
	신뢰성	성숙성	4.3	0.9
		가용성	4.6	0.6
		복구성	4.4	0.8
	사용성	적합인지성	4.6	0.6
		학습성	4.4	0.9
		운영성	4.3	0.8
		사용자 오류보호	4.4	0.8
사용자 인터페이스		4.3	0.9	

문제 해결 방법 탐색	자료 수집 및 분석	0.87	0.82	0.559
	자료 표현	0.87		
	추상화	0.793		
프로그램 설계	독창성	0.303	0.927	0.808
	알고리즘과 절차	0.899		
	병렬화	0.939		
프로그램 구현	동기화	0.858	0.866	0.564
	기능화	0.781		
	효율화	0.771		
	일반화	0.747		
	신뢰성	0.731		
사용성	0.723			

판별타당성(집중타당성의 제공근 이하)에서는 문제 해결 방법 탐색의 집중타당성 제공근과 프로그램 구현 상관계수의 비교에서만 조건을 만족하지 않았고 교차요인은 존재하지 않는 것으로 나타났다.

4.2 확인적 요인분석

평가 요소에 대한 39명 전문가들의 타당도 검토 결과를 이용하여 단계별 평가요소의 구성이 적절한지를 확인하기 위하여 확인적 요인분석을 실시하였다. 확인적 요인분석은 사례수가 39명이므로 표본수에 조건에 유연한 PLS의 방법을 이용한 SmartPLS 프로그램을 사용하였다.

그 결과, 항목신뢰성(0.7이상)에서는 15개 평가 요소 중 문제 정의와 독창성을 제외한 13개 평가요소가 조건을 모두 만족하였고 내적일관성(0.7이상)과 집중타당성(0.5이상)에서는 15개 평가 요소 모두가 조건을 만족하는 것으로 나타났다.

<표 12> 항목신뢰성

평가 단계	평가요소	항목 신뢰성	내적 일관성	집중 타당성
문제인식 및 정의	문제 정의	0.656	0.887	0.729
	문제 분해	0.95		
	목표 정의	0.924		

<표 13> 판별 타당성

	문제 인식 및 정의	문제 해결 방법 탐색	프로그램 설계	프로그램 구현
문제 인식 및 정의	0.854*	-	-	-
문제 해결 방법 탐색	0.767	0.748*	-	-
프로그램 설계	0.684	0.675	0.9*	-
프로그램 구현	0.83	0.781	0.848	0.751*

\* 집중타당성 제공근

Cross loading analysis(0.5 이상)에서는 문제 해결 방법 탐색의 독창성을 제외한 나머지 14개 평가 요소는 모두 조건을 만족하는 것으로 나타났다.



&lt;표 14&gt; Cross Loading Analysis

	문제 인식 및 정의	문제 해결 방법 탐색	프로그램 설계	프로그램 구현
문제 정의	0.656	0.446	0.235	0.489
문제 분해	0.95	0.772	0.775	0.859
목표 정의	0.924	0.696	0.629	0.724
자료 수집 및 분석	0.733	0.87	0.545	0.729
자료 표현	0.657	0.87	0.58	0.614
추상화	0.569	0.793	0.593	0.613
독창성	0.144	0.303	0.211	0.278
알고리즘 과 절차	0.623	0.693	0.899	0.797
병렬화	0.624	0.552	0.939	0.768
동기화	0.595	0.569	0.858	0.719
기능화	0.611	0.688	0.556	0.781
효율화	0.655	0.601	0.753	0.771
일반화	0.741	0.738	0.608	0.747
신뢰성	0.5	0.459	0.549	0.731
사용성	0.573	0.392	0.706	0.723

이상과 같이 본 논문에서 설정된 초·중등학생의 SW 프로그램 개발 과정 및 결과를 평가하기 위한 평가요소에 대한 확인적 요인분석 결과를 종합해보면, 평가요소 일부 요소를 제외한 대부분의 평가요소들이 조건을 만족하는 것으로 나타났다.

조건을 만족하지 못하는 평가 요소에 대해 구체적으로 살펴보면 항목신뢰성에서는 문제 정의와 독창성 요소가, 판별타당성에서는 문제 해결 방법 탐색과 프로그램 구현 간, Cross loading analysis에서는 독창성 요소가 각각 조건을 만족하지 못하는 것으로 나타났다. 따라서 향후 세부적인 평가 기준을 개발 시에 이에 대한 고려가 필요한 것으로 나타났다.

## 5. 결론

본 논문에서는 일선 초·중등학교의 SW교육에서 이루어지는 SW 프로그래밍 교육 평가에서 활용하기 적합한 평가지표를 개발하였다. 이를 위해서 먼

저 기존의 SW 프로그램 평가지표들을 분석하였다. 그 결과, 기존의 SW 프로그램 평가지표들은 창의력, 논리력, 문제해결력 등과 같은 교육학적 측면에 치중하거나 효율성, 신뢰성, 완전성 등과 같은 SW 공학적 측면에 치중하고 있어 두 가지 측면이 모두 균형있게 평가되어야 하는 초·중등학교의 SW교육에는 다소 적합하지 않은 것으로 나타났다.

따라서 본 논문에서는 일선 초·중등학교의 SW교육을 위해 교육학적 측면의 가치와 SW 공학적 측면의 가치가 조화롭게 평가될 수 있도록 구성된 평가지표를 개발하였다. 이를 위해서 먼저 컴퓨팅 사고력의 세부 요소 분석을 통해 교육학적 측면의 평가 요소를, SW 품질 요소를 정의한 국제표준 ISO/IEC 25010에서 SW 품질적 측면의 평가요소를 각각 추출하였다. 그리고 이를 교육부에서 개발한 초·중학교 SW교육 교재의 운영 단계(문제 인식 및 정의, 문제 해결 방법 탐색, 프로그램 설계, 프로그램 구현)에 맞춰 평가지표를 개발하였다.

개발된 평가지표의 타당성 확보를 위하여 SW 교육 및 SW 공학 전문가 39명에게 타당성 검토를 의뢰하였고 그 결과를 바탕으로 평가지표의 내용타당성과 확인적 요인분석을 실시하였다.

그 결과 본 논문에서 개발한 초·중학교 학생들이 개발한 SW 프로그램에 대한 평가지표는 내용타당성과 요인구성은 모두 적합한 수준인 것으로 나타났다.

이와 같이 본 논문에서 개발된 SW 프로그램 평가지표는 SW교육의 신뢰성을 높여주고 학생들에게 자신의 학습에 대한 자아성찰의 기회를 제공해 주며, 정보영재교육과 SW 프로그램 공모전 및 경시대회, SW 능력 인증 및 자격제도 등의 다양한 교육활동 활성화에 기여할 것으로 기대된다.

## 참 고 문 헌

- [1] 교육부(2014). 「초·중등 SW교육 활성화 방안」. 교육부 보도자료.
- [2] 교육부(2014). 「2015 문·이과 통합형 교육과정」. 공청회 자료집.
- [3] 교육부(2015). 「소프트웨어 교육 운영 지침」. 교육부.

- [4] 위키백과. en.wikipedia.org.
- [5] 정종광 (2010). **과학고 학생을 위한 OnlineJudge 기반 프로그래밍 평가 시스템의 설계 및 구현**. 석사학위 논문, 한국교원대학교 교육대학원.
- [6] 미래창조과학부 (2015). **2015년 한국정보올림피아드 공모대회 운영 계획**.
- [7] 미래창조과학부 (2015). 제9회 공개SW 개발자대회 홈페이지(project.oss.kr).
- [8] UVa Online Judge. uva.onlinejudge.org.
- [9] Andy Kurnia.& Andrew Lim.& Brenda Cheang (2001). Online Judge. *Computers& Education 36*, 299-315.
- [10] Yingwei L., Xiaolin W., & Zhengyi Z (2008). Programming grid: a computer-aided education system for programming course based on onlnie judge. *ACM International Conference Proceeding Series 368*.
- [11] 최형신(2014). Computational Thinking역량 개발을 위한 수업 설계 및 평가 루브릭 개발. **정보교육학회논문지**, 18(1), 57-64.
- [12] 최숙영 (2011). 21st Century Skills와 Computational Thinking 관점에서의 "정보" 교육과정 분석. **컴퓨터교육학회논문지**, 14(6), 19-30.
- [13] ISO/IEC 25010 Software engineering - Software product Quality Requirements and Evaluation (SQuaRE) - Quality model and guide.
- [14] Wing, J. M (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.
- [15] Wing, J. M (2008). Computational thinking and thinking about computing. *Royal Society of London Philosophical Transactions*, 366(1881), 3717-3712.



## 안 성 훈

2001 한국교원대학교  
컴퓨터교육과(교육학 박사)  
2004~2011 한국교육개발원  
연구위원

2011~2013 한국교육학술정보원 선임연구원  
2013~현재 경인교육대학교 조교수  
관심분야: 교육용 콘텐츠, e-러닝, 컴퓨터교육  
E-Mail: shahn@ginue.ac.kr