

대학 소프트웨어 기초교육 효과성 모형 탐색*

홍성연 · 서주영 · 구은희 · 신승훈 · 오하영 · 이택균

아주대학교 다산학부대학

요약

최근 전공과 상관없이 모든 학생들에게 소프트웨어(SW)를 기초교육으로 의무적으로 수강하도록 하는 대학이 늘고 있다. 교육의 제공자와 수요자 간의 원활한 의사소통을 촉진하고, 전교생을 대상으로 하는 교육의 책무성을 확인하기 위하여 SW 교육 효과성 모형을 개발할 필요가 있다. 이를 위해 본 논문은 선행 연구를 기초로 SW 기초교육의 목표와 내용을 검토하고 효과성 모형의 영역으로 컴퓨팅 사고력, SW 문해력, SW 인식, SW 태도를 제시하였다. 컴퓨팅 사고력은 일반적인 문제해결과정과 비교하여 문제의 분석과 설계에 초점을 맞춘 분해, 패턴 인식, 추상화, 알고리즘으로 구성되었다. SW 문해력은 디지털 시대에 정보 읽기와 활용을 넘어서 SW를 바탕으로 한 사회발전을 조망하는 것을 포함한다. SW 인식과 태도는 전공이 정해진 대학생의 상황과 수준을 고려하여 구성하였다. SW 교육 효과성 모형은 추후 진단도구 개발을 위한 기초자료로 활용될 수 있을 것이다.

키워드 : 소프트웨어 교육, 교육 효과성, 컴퓨팅 사고, 소프트웨어 문해

Exploratory study on the model of the software educational effectiveness for non-major undergraduate students

Seongyoun Hong · Jooyoung Seo · Eunhee Goo · Seunghun Shin ·

Hayoung Oh · Taekkyeun Lee

Dasan University College, Ajou University

ABSTRACT

Software courses required for all students regardless of their major in many universities. SW educational effectiveness model needs to be developed to enable effective communication among students, professors, and SW educators, and to identify the responsibilities of SW educators. SW educational effectiveness model based on literature review is composed of computational thinking, SW literacy, SW awareness, and SW attitude. Computational thinking, focused on analysis and design of problem solving processes, consists of decomposition, pattern recognition, abstraction, and algorithms. SW literacy involves viewing social development based on SW beyond information literacy in the digital age. The SW awareness and attitude were organized by considering the collegiate contexts. The SW educational effectiveness model will be used as the basis for diagnosis tools as further studies.

Keywords : Software Education, Educational Effectiveness, Computational Thinking, Software Literacy

* 본 연구는 2018학년도 아주대학교 일반연구비 지원에 의하여 연구되었음

교신저자: 서주영(아주대학교 다산학부대학)

논문투고 : 2019-08-28

논문심사 : 2019-09-07

심사완료 : 2019-10-15

1. 연구의 필요성 및 목적

최근 초등학교에서 대학까지 소프트웨어(이하 SW)를 정규 교육과정으로 편성하는 기관이 늘고 있다. 2015개정교육과정에 따라 초·중등학교에서는 2018년부터 단계적으로 정보교과를 필수적으로 이수하도록 하고 있다[31]. 또한, 2015년부터 SW중심대학사업을 전개하여 SW 융합인재 양성을 목적으로 인문·사회계열을 포함한 전체 학생들에게 SW 교육을 의무적으로 이수하도록 하고 있다. 2018년 말 기준으로 이 사업에 선정된 대학은 전국 35개에 이른다[29][32].

과학 기술을 기반으로 급속한 경제 성장을 이룬 우리나라에서 이처럼 정책적으로 SW 교육을 확대하는 것은 정부가 SW를 미래 산업이 나아갈 방향이자 앞으로 사회를 살아가는데 필요한 핵심 역량으로 설정하고 있음을 시사한다[18]. 4차 산업혁명이라 불리는 기술변혁의 시대를 살아가는 학생들에게 SW와 관련된 지식과 소양을 갖추는 것은 단순히 직업 세계로의 진출을 준비하는 것 이상으로 삶의 질 향상을 위해서도 중요하다.

그러나 초·중등교육에서 정보교과의 도입이 교육과정 개발, 연구학교 운영 등 비교적 체계적인 준비과정을 거친 것과 다르게, SW중심대학사업은 여타 대학의 재정 지원사업과 같이 공모의 방식으로 이루어지면서 준비과정이 길지 않았다. 일부 대학은 구성원과 충분한 합의 없이 갑작스럽게 진행되면서 학생은 물론 교수도 개인의 의사와 상관없는 SW 의무교육에 불만을 제기하기도 하였다. 또한 비전공자 대상 SW 수업에 대한 경험 부족으로 SW 전공자와 동일한 내용과 난이도로 가르치거나 이론의 전달에 치중한 수업을 하거나 학생의 수준과 흥미를 제대로 파악하지 못하는 등 효과적인 교육이 이루어지지 못하기도 하였다. 즉 무엇을 가르쳐야 하고, 어떻게 가르쳐야 하는지에 대한 혼란과 시행착오를 겪기도 하였다[4][11][35].

SW중심대학사업이 5년째 접어들면서 SW 교육의 필요성에 대한 대학 구성원의 인식은 상당히 개선되었으며, 대학마다 학생의 요구에 따른 다양한 SW 교육 프로그램이 제공되고 있다[33]. 또한 SW비전공자의 필요와 흥미를 고려하여 교육과정과 수업방법을 체계적으로 개선해 나가면서 이제는 SW 교과목이 대학의 필수 교육으로 어느 정도 자리매김하고 있다[4][14][21].

그러나 비전공자에 대한 SW 교육이 의도했던 바대로 효과적인 SW 융합인재를 양성하고 있는가에 대해서는 여전히 의문의 여지가 있다. 교육방식의 이질감으로 문과생들의 SW 교육 참여는 아직도 충분하지 못하고[18], 학년과 경험에 따라 수준과 요구의 차이가 상당한 학생들에게 적절한 SW 교육 프로그램을 마련하는 것이 쉽지 않다[33]. 한국교양기초교육원에서 SW 문해 교육을 글쓰기나 수학, 과학과 같이 디지털 사회에서 대학생이면 누구나 갖추어야 하는 기초교육의 한 영역으로 분류하였지만[30], 여전히 SW 교육의 효과가 무엇이며 어느 정도 달성되고 있는지에 대해 의심하는 시선도 대학 내에 존재한다.

이러한 상황에서 대학 SW 교육의 목적과 내용을 면밀히 검토하고, 그 효과를 검증할 수 있는 기준을 제시하는 것은 대학 SW 교육의 체계적인 개선을 위해 중요하다. 이는 대학생이 수강해야 하는 교양으로 SW 기초교육에 대한 목적과 범주를 분명히 할 뿐만 아니라, SW 교육을 제공하는 교수와 학생, SW 교육의 제공 학과와 수요 학과의 구성원 간의 소통을 원활히 하는 데도 도움이 될 것이다.

본 논문에서는 대학의 비전공자 SW 교육의 유형과 목적, 내용 등을 살펴보고, SW 기초교육을 통해 얻게 되는 성과인 컴퓨팅 사고력과 SW 문해력에 대해 검토해 보고자 한다. 이를 토대로 SW 기초교육의 효과성 모형을 제안해 보고자 한다.

연구문제를 구체적으로 기술하면 다음과 같다. 첫째, 대학의 SW 교육의 내용과 현황은 어떠한가? 둘째, 비전공자를 대상으로 하는 대학 SW 기초교육으로 컴퓨팅 사고의 목적과 주요 내용은 무엇인가? 셋째, 비전공자를 대상으로 하는 대학 SW 기초교육으로 SW 문해력의 목적과 주요 내용은 무엇인가? 넷째, 대학의 비전공자 SW 기초교육의 효과성 모형은 어떻게 구성될 수 있는가?

2. 연구 방법

본 연구는 관련 자료와 문헌을 비판적으로 검토하고 연구자의 관점에서 재해석하는 문헌고찰연구 방법으로 수행되었다.

문헌고찰(literature review)은 학술 연구나 논문의 한 부분으로 구성될 수도 있지만, 양적연구나 질적연구처럼

주요한 연구 방법으로 활용될 수 있다. 이는 미국심리학회(APA)에서 분류한 연구 방법의 한 유형으로 ‘앞서 출판된 자료를 조직화와 통합화를 하고, 평가함으로써 문제를 명료화하고 연구의 진척을 이루어’내는 연구 방법이다. 단순히 선행 연구를 수집하고 정리하는 것을 넘어서 연구자의 관점에 따라 가치판단적으로 논지를 기술해야 하며, 이러한 과정을 통해 특정 학문 분야의 이론 발전에 기여할 수 있다[7].

문헌고찰의 결과를 토대로 모형을 개발하는 과정에서 소프트웨어, 교육학 등 전공이 다른 여섯 명의 연구진이 최종 합의에 이를 때까지 토론을 거쳤으며, 총 10회 이상의 회의를 진행하였다.

최종 도출된 모형은 수도권과 비수도권의 4년제 3개 대학에서 SW 기초교육을 담당한 경험이 있는 13명의 교수에게 타당성, 이해도, 설명력, 보편성, 유용성 등에 대해 5점 척도로 평정하도록 하고 내용타당도를 검토하였다.

3. 대학의 비전공자 SW 교육

3.1 대학의 비전공자 SW 교육 유형

2015년 SW중심대학사업을 시작한 이후, ‘SW 융합인재양성’을 목적으로 여러 대학에서 전공에 상관없이 모든 학생들에게 SW 교육을 필수교과로 이수하도록 하고 있다.

대학의 비전공자 SW 교육은 유형에 따라 ‘SW 기초교육’과 ‘SW 융합교육’으로 구분할 수 있다. SW 기초교육은 SW를 ‘글쓰기, 말하기, 계산하기’와 같은 기초교양으로 본다. 4차 산업혁명 시대에서 SW는 기본적인 지적, 언어적 능력의 일환이며, 디지털 문명의 자료와 정보를 해독하고 활용할 수 있는 SW 문해력은 전공과 상관없이 모든 학생들이 갖출 필요가 있다[30].

SW 융합교육은 SW를 자신의 전공과 융합할 수 있는 역량을 위한 교육을 의미한다. 이는 전공의 문제해결을 위한 구체적인 도구로 SW를 활용하거나 최소한 SW 개발자와 소통함으로써 필요한 도움을 얻을 수 있는 능력을 확보하는 것을 의미한다. 이를 위해 SW 기초교육을 넘어 전공별 특성을 반영한 프로그래밍 교육이 포함되는 경우가 대부분이다[9][33].

3.2 대학의 비전공자 SW 교육 내용

대학의 SW 교육을 내용에 따라 나누면 일반적으로 도구 활용 교육, 프로그래밍 교육, 컴퓨팅 사고 교육의 세 유형으로 나눌 수 있다.

도구 활용 교육은 엑셀, 파워포인트, 워드 프로세스 등 사무 처리에 주로 사용되는 응용 프로그램의 사용법을 학습하는 것으로, 도구로써 컴퓨터를 활용하는 능력을 배양하는 것을 목적으로 한다. 도구 활용 교육은 대학에서 컴퓨터 교육이 활성화되던 1990년대 초반부터 교양교육으로 운영되었다.

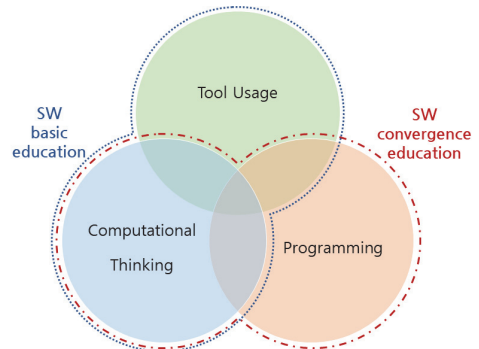
프로그래밍 교육은 프로그래밍 언어에 대한 학습을 목표로 하는 교육을 말한다. 과거 Basic, C와 같은 프로그래밍 언어가 주로 다루어졌다면 최근에는 인공지능이나 빅데이터 분석에 많이 사용되는 Python, R 등의 언어에 대한 프로그래밍 교육이 선호되고 있다. 프로그래밍 교육은 처음에는 SW 전공자가 주 대상이었으나 최근 ICT 시대의 SW 융합에 대한 요구가 높아짐에 따라 다양한 전공에서 꾸준히 수강생이 늘고 있으며, 프로그램이 내재된 스마트 기기나 로봇 개발과 관련된 전자계열, 기계계열과 같은 공과대학에선 전공 기초교과로 자리매김하는 경우도 일반적 현상이다.

컴퓨팅 사고 교육은 2000년대 후반에 Jeannette Wing에 의해 대두되었다. 그는 컴퓨팅 사고 교육을 SW 전공자뿐만 아니라 디지털 사회를 살아가는 사람이라면 누구나 받아야 하는 보편 교육으로 확대함으로써, 컴퓨팅 사고 교육의 중요성을 사회에 인식시키는데 기여하였다. 그에 따르면 컴퓨팅 사고는 ‘컴퓨터 과학의 기본 개념에 기초해 문제를 해결하고, 시스템을 디자인하며, 인간의 행동을 이해하는 접근 방법이며, 디지털 세대에게는 읽기, 쓰기, 계산하기와 같은 동일한 기초적인 능력’으로 정의한다[10]. 즉 컴퓨팅 사고는 프로그래밍 기술에 국한된 것이 아닌 폭넓은 인지사고 단계를 포함하며 다양한 분야의 문제 해결에 적용할 수 있는 4차 산업시대를 맞는 인간이 갖추어야 할 보편적 능력 중 하나이다.

3.3 대학의 비전공자 SW 교육 현황

최소 3년 이상 교양필수로 SW 교육을 운영한 14개 주요 대학의 SW 교육 현황을 분석하면[13], <Table 2>

와 같이 과거 주를 이루던 도구 활용 교육은 모두 사라지고 컴퓨팅 사고 교육과 프로그래밍 교육이 주를 이룬다. 전교생 필수 교육으론 컴퓨팅 사고 교육을, 전공에 따라 추가적으로 필수로 지정되는 교육으론 프로그래밍 교육을 선호하고 있음을 알 수 있다. 즉 SW 기초교육으로 컴퓨팅 사고 교육을, 공과대학이나 자연대학과 같이 SW 융합교육에 대한 필요와 요구가 높은 전공은 프로그래밍 교육을 추가로 실시하고 있다.



(Fig. 1) SW education diagram of non-major undergraduate students

<Table 1> SW basic education status of SW-oriented universities

Universities	Tool Usage	Computational thinking	Programming
Gachon Univ.	-	○	◎
Kyungpook Univ.	-	○	
Korea Univ.	-	○	◎
Sogang Univ.	-	○	◎
Sungkyunkwan Univ.	-	○	◎
Sejong Univ.	-	◎	◎
Ajou Univ.	-	◎	◎
Chungnam Univ.	-	○	◎
Kookmin Univ.	-		◎
Dongguk Univ.	-	○	○
Pusan Univ.	-		◎
Seoul Women's Univ.	-	○	
KAIST	-		◎
Hanyang Univ.	-	○	

(○ All students required ◎ Required by major)

3.4 대학의 비전공자 SW 교육 유형에 따른 교육 내용 구성

현재 대학의 비전공자 SW 교육은 전체 학생을 대상으로 하는 SW 기초교육과 학생의 학습 필요나 요구에 따른 SW 융합교육으로 구분할 수 있다. 유형별로 SW 교육 내용을 구분하면 SW 기초교육에는 도구 활용 교육과 컴퓨팅 사고 교육이, SW 융합교육에는 컴퓨팅 사고 교육과 프로그래밍 교육이 포함될 수 있다(Fig. 1 참조).

SW 기초교육은 과거에는 컴퓨터 도구 활용을 원활하게 하는 것이 주요 목표였다. 그러나 컴퓨터가 익숙한 디지털 세대엔 도구 활용보다 인지적으로 컴퓨팅 사고

를 통한 문제해결 역량을 배양하는 것으로 교육의 중심이 변화하고 있다. 이 과정에서 디지털로 된 자료와 정보를 이해하고, SW 전공자와 의사소통하는 SW 문해력도 함께 얻게 된다.

SW 융합교육도 기존엔 프로그래밍 언어의 문법적 학습에 몰입하였으나 최근에는 컴퓨팅 사고력을 기반으로 문제를 해결하고 이를 구현하기 위한 방법으로 프로그래밍 언어를 익히는 방식으로 변화하고 있다. 즉 다양한 전공의 SW 비전공자가 각자의 분야에서 나타나는 문제를 해결하는 과정에서 컴퓨팅 사고를 하게 되고, SW를 통해 문제를 해결하고자 할 때 SW 융합인재로서 필요한 프로그래밍 언어를 배워서 적용하거나 프로그래밍 지식을 바탕으로 SW 전공자와 원활하게 소통할 수 있다.

대학 비전공자 SW 교육에서 SW 기초교육은 모든 전공의 모든 학생들을 대상으로 하는 교양교육으로 대부분 운영되지만, SW 융합교육은 전공이나 학생의 요구에 따라 선택적으로 수강할 수 있게 전공 또는 교양으로 개설된다. 어떤 유형이든 간에 SW 교육의 핵심은 컴퓨팅 사고 교육이라 할 수 있다. 대학의 비전공자 SW 교육에 대해 다루는 이 연구는 모든 학생들에게 제공되는 SW 기초교육 유형을 집중해서 살펴보고자 한다.

4. 대학 SW 기초교육에서 컴퓨팅 사고

4.1 SW를 활용한 문제해결과

SW를 활용한 문제해결은 일반적인 문제해결과 비교

할 때, 문제를 정의하고 관련 자료를 수집하며 해결 전략을 설계하여 실제 문제를 해결한다는 큰 틀은 유사하다. 그러나 문제해결과정에서 컴퓨터를 활용하는 문제를 다루고 궁극적으로 프로그래밍 기술을 통해 구현하여 자동화한다는 점에서 근본적인 차이가 있다.

SW를 활용한 문제해결과정을 도식화하면 (Fig. 2)와 같이 문제를 정의하고, 관련 자료를 수집하고, 이를 바탕으로 문제를 분석하고, 해결책을 설계하고, 설계된 해결책을 시뮬레이션을 통해 검증하고, 자동화를 위해 프로그램으로 구현하는 과정으로 구성된다[3][8][28].

이 중에서 문제 정의, 자료 수집, 분석, 설계는 문제해결을 위한 인지적 사고의 단계라 할 수 있다. 인지적 사고의 결과물인 알고리즘은 프로그래밍 기술을 이용한 시뮬레이션과 자동화를 통해 프로그램으로 구현될 수 있으며 효율적인 프로그램 구현을 위해선 문제를 분석하고 설계하는 과정부터 프로그래밍적 요소가 고려되는 것이 효과적이다. (Fig. 2)에서처럼 SW 교육을 통한 문제해결과정의 단계들 중 문제를 정의하고 자료를 수집하는 초반 단계는 일반적인 문제해결에서 이루어지는 사고과정과 유사하고, 자동화를 위해 프로그래밍 언어로 시뮬레이션하고 구현하는 후반 단계는 프로그래밍 기술과 밀접하다면, 이들 사이에서 중첩되는 문제의 분석과 설계의 단계는 문제를 컴퓨터가 처리할 수 있는 형태로 변환하기 위해 단순 인지사고가 아닌 프로그래밍 기술이 융합된 계산적·기술적 역량을 필요로 하는데, 이 과정이 SW 교육을 통해 집중적으로 학습되어야 할 ‘컴퓨팅 사고’이다.

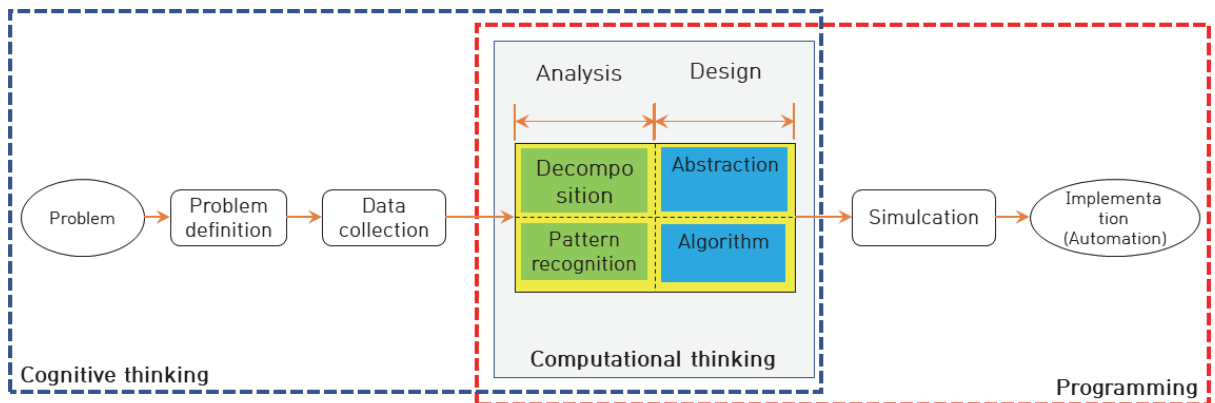
4.2 문제해결과정 중 컴퓨팅 사고

컴퓨팅 사고 교육에 대한 많은 선행 연구에서는 (Fig. 2)의 문제를 정의하는 단계부터 해결책을 프로그램으로 구현하는 전체의 과정을 컴퓨팅 사고 교육으로 정의하고 있다. 실제 SW가 필수교육으로 자리 잡은 국내의 초중등 교과과정에서도 (Fig. 2)의 문제 해결 전 과정을 교육에 포함하고 있다[2][3][6][12][28].

하지만 SW 교육이 아니더라도 학교와 대학에서의 다양한 교육은 사고력 배양을 목표로 한다. 문제해결과정의 전체 과정을 살펴보면, 문제를 정의하고, 자료를 분석하는 것은 여러 교육과정에서 가르치고 있다. 따라서 대학의 SW 교육은 SW를 활용한 문제 해결 과정의 고유한 측면이자, 인지사고 단계와 프로그래밍 기술 단계에 복합적으로 융합된 분석과 설계 단계를 컴퓨팅 사고로 집중 교육할 필요가 있다.

컴퓨팅 사고는 효율적이고 효과적인 문제해결과정을 지원하기 때문에 SW를 전공하든 그렇지 않든 필수적으로 갖출 필요가 있다. 컴퓨팅 사고 이후의 프로그래밍을 통한 구현은 전공에 따라, 또는 학생에 개인적인 선택에 따라 필요할 수도 있고 아닐 수도 있다. SW 교육에 대한 다양한 요구가 공존하고 있음을 고려할 때 프로그래밍 언어에 대한 교육은 선택적으로 적용할 필요가 있다.

정리하자면 대학의 비전공자 SW 교육은 교육의 유형이 기초교육이든, 융합교육이든 상관없이 SW를 활용한 문제 해결력 배양을 위한 컴퓨팅 사고가 교육의 중심이 되어야 하며, 이는 전공에 상관없이 누구에게 요구되는 역량이다.



(Fig. 2) Problem solving process via SW education

4.3 컴퓨팅 사고의 핵심 개념

컴퓨팅 사고 관련 연구에서 문제 분석 및 설계에 요구되는 핵심 개념으로 공통적으로 정의하고 있는 것은 분해, 패턴인식, 추상화 및 알고리즘이다.

(Fig. 3)은 분해, 패턴인식, 추상화, 알고리즘의 핵심 개념의 주요 특성, 각 개념 간의 관계 및 이들을 활용한 문제 해결 절차에서 얻을 수 있는 결과물의 유형을 도식화한 것이다.

복잡한 문제는 문제의 규모, 기능, 역할, 조건 등을 기준으로 해결 가능한 수준의 단순한 문제로 세분화된다. 분해된 개별 문제들에서 일정한 경향, 반복적 규칙, 공통적 속성이 있는 지 식별하여 패턴을 인식한다. 패턴들 사이에 불필요한 사항을 제거하여 간소화시킴으로써 핵심 요소를 파악하는 추상화를 수행한다. 추상화를 통해 일반화된 문제는 순차적인 작업, 반복적인 작업, 조건에 의해 선택되는 작업 등의 문제 해결을 위한 세부 절차들의 알고리즘으로 기술된다.

각각의 개념을 구체적으로 살펴보면 다음과 같다.

4.3.1 분해

분해는 대량의 데이터 처리 혹은 복잡한 프로세스의 수행과 같이 한 번에 해결하기 어려운 복잡한 문제를 해결 가능한 수준의 작은 부분 혹은 단계로 나누는 것을 말한다[2][3][28]. 이 때 분해의 단위는 문제의 규모(크기), 요구 사항에 근거한 기능, 수행되어야 하는 역할, 문제가 갖는 조건 등을 기준으로 한다. 분해를 통해 얻어

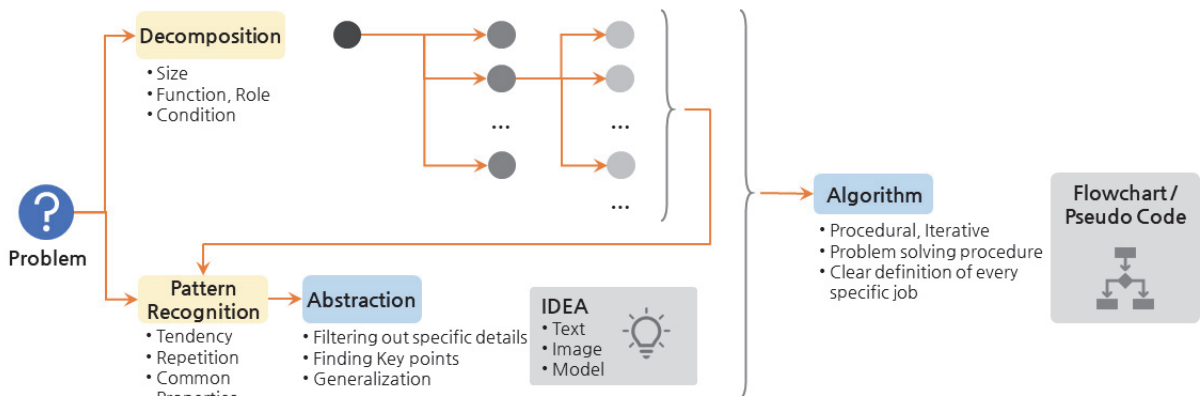
지는 결과물은 유한개로 구성된 작은 규모의 문제 집합이며, 집합을 구성하는 개별 문제는 직관이나 간단한 연산만으로 해결할 수 있는 수준의 단순한 문제가 된다. 다만, 분해된 결과로 생성된 작은 문제들을 하나하나 해결하면 궁극적으로 최초에 주어진 문제가 해결되어야 한다. 즉 최초 주어진 문제의 본질이 분해 과정에서 왜곡되지 않도록 일관성이 유지되어야 함이 중요하다.

4.3.2 패턴 인식

복잡한 문제를 보다 효율적으로 해결하기 위한 목적으로 분해된 개별 문제들이 공유하는 속성, 경향, 규칙 등의 유사성을 찾는 과정을 패턴인식이라 한다[2][3][28]. 이 과정에서 찾아지는 패턴의 유형은 복수 개의 개별 문제가 공통적으로 포함하는 구성 요소 혹은 유사성, 반복 적용 가능한 규칙 및 일정한 경향 등이다. 따라서 이 과정을 거치며 반복 수행이 요구되는 동일한 성격의 세부 작업이 정리되거나 기존의 유사 문제에 적용되었던 해결책이 그대로 혹은 약간의 변경을 통해 적용될 수 있는지에 대한 검토가 수행되므로 최초 주어진 전체 문제의 규모 감소 및 난이도 경감이 이루어질 수 있다.

4.3.3 추상화

추상화는 수행되어야 하는 내용의 일반적인 특성을 찾기 위해 발견된 패턴들로부터 이들에 내재된 핵심적인 개념을 추출하는 과정이다[2][3][28]. 세부적, 부수적인 특성들을 정제하여 개별 패턴들에 포함된 일반성이 추출되면 주어진 문제를 해결하는데 요구되는 개념을



(Fig. 3) Relationship between key concepts of computational thinking

정리할 수 있다. 추상화의 결과물은 주어진 문제의 유형 및 문제를 해결에 활용되는 도구 혹은 환경에 따라 텍스트, 이미지 혹은 모델 등 다양한 형태로 생성될 수 있으며, 생성된 결과물에는 작성자가 제시하는 문제 해결을 위한 아이디어가 함축되어 표현된다.

4.3.4 알고리즘

알고리즘은 문제를 해결하거나 주어진 작업을 수행하는데 필요한 단계적 절차 혹은 규칙의 나열 혹은 정의이다[2][3][28]. 따라서 알고리즘 정의 단계에서는 수행되어야 하는 개별 작업의 명확한 정의가 이루어지며 이들이 수행되어야 하는 순서에 대한 계획이 이루어진다. 따라서 정의된 알고리즘 내에는 해석 및 실행 방법에 대한 모호함이나 고려되지 못한 예외 상황이 남아있어서는 안 되며, 알고리즘을 구성하는 각 단계는 문제 해결이라는 목표를 지향하도록 정의되어야 한다. 정의된 알고리즘은 자연어로도 기술될 수 있으나 일반적으로 순서도나 의사코드(pseudo-code) 형태를 취하며 이는 대체로 컴퓨터를 활용한 자동화 과정의 시작점이 된다.

5. 대학 SW 기초교육에서 SW 문해

한국교양기초교육원에서 제시한 교양교육표준안[30]에 따르면, 대학에서 가르쳐야 하는 기초교육의 한 유형으로 ‘디지털 문명의 자료·정보(SW)를 해독할 수 있는 SW(정보) 문해’를 제시하고 있다.

현재 정보 문해에 대한 명확한 정의는 미흡한 상황인데, 정보 문해를 하나의 연구 분야로 포함하고 있는 정보문헌학에서는 ‘정보화 기기에 의해 생산되거나 유통되는 정보를 읽고 사용할 수 있는 능력’으로 정의하고 있다[26][27]. 정의에서 보듯이 정보문헌학에서의 SW 문해는 디지털 ‘정보(information)’에 대한 문해로 국소적 해석이라 할 수 있다.

중·고등학교의 SW 관련 교과인 정보 과목에선 ‘정보(informatics)는 컴퓨터과학의 기본 개념과 원리 및 기술을 바탕으로 실생활과 다양한 학문 분야의 문제를 창의적이고 효율적으로 해결하기 위한 학문 분야’로 언급하고 있다[16]. 이를 토대로 대학의 SW 기초교육에서의 SW(정보) 문해는 ‘IT 기술문명 세계의 다양한 인간·사

회·문화적 현상을 SW 시각으로 체계적으로 탐구하고 이해할 수 있는 능력’으로 일상생활에서의 컴퓨터(SW) 및 정보의 활용은 물론 SW가 만들어낸 사회 현상 및 기술 발전을 이해하고, 미래 환경에서 SW를 통해 나타날 변화를 예측하고 대응하는 능력을 포함한다.

본 논문에선 SW 문해를 컴퓨팅 사고와 함께 대학의 SW 기초교육을 통해 학습되어야 할 역량으로 정의한다. 컴퓨팅 사고가 컴퓨터와 유사한 사고방식으로 문제를 효율적·효과적으로 해결할 수 있도록 하는 일종의 문제해결과정이라면, SW 문해는 IT 기술문명 세계와의 효과적인 의사소통을 위해서 어떠한 방식으로 컴퓨터(SW) 및 정보를 사용할 것인지, SW가 인간생활에 어떤 영향을 미쳐왔고 사회를 어떻게 변화시켰는지, 시대를 주도하는 기술이 SW와 어떤 연관이 있는지와 같이 SW로 인한 개인적, 사회적, 문화적 현상을 식별하고 이해하며 이에 대응하는 것을 의미한다.

SW 문해의 핵심 요소를 정리하면 다음과 같다.

SW 및 정보 활용 - 컴퓨터와 SW의 기본적 사용 방법과 이를 통해 생성된 정보를 이해하고 응용할 수 있는 능력

IT 기술문명 이해 - 컴퓨터와 SW의 발전을 이끈 사회·문화·기술적 환경에 대한 역사적 이해를 바탕으로 시대적 변화 흐름에 대한 이해

SW 미래기술 이해 - 빅데이터, 인공지능, 가상현실과 같은 발전하는 미래기술과 SW의 역할에 대한 이해

정보 보안 및 윤리 - 디지털 정보의 다양성과 정보 보안의 중요성 및 인터넷 시대를 살아가는 인간이 지켜야 할 윤리에 대해 이해 및 준수

6. SW 기초교육 효과성의 모형

6.1 SW 교육 효과성 관련 연구

효과란 사전적으로 ‘어떤 목적을 지닌 행위에 의해서 드러나는 보람이나 좋은 결과’를 의미한다[34]. 효과성이란 논리적 인과관계로 인해 효과가 나타나는 것으로, 효과의 성질이나 상태의 존재여부를 추상적으로 나타낸 말이다. 따라서 교육 효과성이란 교육을 통해 나타나게 되는 성과를 지칭하는 다분히 성과지향적인 용어이다

[19]. 따라서 교육을 통해 달성하고자 하는 성과를 효과성의 구인으로 개발하고 이를 측정함으로써 교육의 성과를 확인하고 개선의 방향을 모색한다.

SW 교육의 효과성을 무엇으로 볼 것인가에 대해서는 아직 합의된 것은 없다. 그러나 2015개정교육과정에 정보 교과가 필수교과로 포함되면서, 미래창조과학부나 한국교육학술정보원 등 정부의 지원을 받아 SW 교육이나 SW 연구학교의 교육 효과성을 분석하려는 시도가 이루어졌다[5][22][23]. 대표적으로 이환철 외[5]와 이원규 외[23]의 연구를 살펴보면, 컴퓨팅 사고력과 관련된 인지적 변화와 SW에 대한 태도나 인식 등 정의적 변화를 중심으로 SW 교육 효과성 구인을 제시하고 있다.

이들 연구 외에도 SW 교육의 인지적 효과를 컴퓨팅 사고력에 국한하여 언급하고 있는 연구가 상당수이다 [1][9][15][17][20][24][25].

반면 SW에 대한 인식 변화 등 정의적 영역을 교육의 효과로 다룬 연구는 많지 않다. 그러나 SW 비전공자를 대상으로 하는 교육이라면, 위의 연구[5][23]와 같이 SW 인식이나 태도 변화 등도 SW 기술 사회를 준비하기 위해 교육의 효과로 나타나야 할 부분임에 틀림없다.

6.2 SW 기초교육 효과성 모형을 위한 시사점

이상의 연구를 통해 대학의 SW 교육의 효과성 모형 개발의 시사점을 정리하면 다음과 같다.

첫째, SW 기초교육 효과성 모형은 대부분의 대학에서 교양필수로 운영되는 SW 기초교육의 결과로 나타난 직간접적인 효과를 측정하는데 국한할 필요가 있다. 비전공자 대상 SW 교육이라 하더라도 목적에 따라 SW 기초교육과 SW 융합교육으로 분류될 수 있으며, 교육 내용과 성격도 차이가 있다. 이 연구에서는 전공에 상관없이 모든 학생들이 배워야 하는 기초교육에 한정하여 SW 교육의 효과성 모형을 개발한다.

둘째, SW 기초교육의 인지적 효과는 컴퓨팅 사고력과 SW 문해력의 함양이다. SW 기초교육에 해당하는 교육 내용은 도구 활용과 컴퓨팅 사고이나(Fig. 1) 최근 도구 활용은 비교과, 또는 개인 차원에서 학습하기 때문에 교육 효과성 모형에 포함하는 것이 적절하지 않다. 반면, SW 기초교육을 통해 디지털로 된 자료와 정보를 활용하고 SW의 영향을 인식하는 등 SW 전공자와 소

<Table 2> Related work comparison of SW education effectiveness

HwanChul Lee et al. (2015)	WonGyu Lee et al. (2017)
[Cognitive area]	[Computational thinking]
Computational Material & Output: Algorithm, Data, Program, Automation, Abstraction	Analytical skills: Data collection, data analysis, data expression
Computational Concepts: Sequence, Repeat, Selection, List, Variable, Function, Computation, Input&Output	Modeling skills: Problem decomposition, abstraction, algorithm
Computational Practice: Collection, Expression, Design, Implementation, Error collection, Decomposition	Automation, testing
[Justice area]	[Attitudes and perceptions on SW education (students)]
Value: Usefulness, importance, convenience, necessity, class satisfaction, career search	SW education experience
Attitude: immersion, communication, cooperation, accountability	Social change due to SW
Effectiveness of CT : Self-directness, confident	SW education usefulness
Interest: learning interest	Internet and computer control
	SW education and further study
	Training time of SW education

통할 수 있는 SW 문해력을 직간접적으로 배우기 때문에 이를 교육 효과로 볼 수 있다.

셋째, SW 기초교육의 정의적 효과는 SW의 가치, 중요성, 영향력 등을 인식하고, SW 활용에 흥미와 자신감을 갖는 것이다. 초·중등의 교육 효과성 도구(<Table 2>)의 정의적 영역과 비교해 보면 일부 항목에서 차이가 나타난다. 초·중등과 다르게 전공을 이미 선택한 대학생이 학습 주체임을 감안하여 SW 관련 진로나 진학 관련된 내용은 제외하였으며, 수업만족도 등 교수자의 개인적 역량에 따라 변화될 수 있는 내용도 배제하였다.

넷째, SW 교육의 목적과 내용에 대한 면밀한 선행 연구 검토를 바탕으로 효과성 구인과 지표를 개발할 필요가 있다. 예컨대, 현재 많은 연구에서 분해, 패턴인식, 추상화, 알고리즘을 컴퓨팅 사고의 핵심 개념으로 제시하지만, 이것의 구체적인 내용이 무엇이며, 어떻게 측정될 수 있는지 자세하게 분석한 것은 많지 않다. (Fig. 3)과 같이 선행 연구에 근거하여 논리적으로 각 구인을 명세화할 필요가 있다.

다섯째, SW 기초교육 효과성 모형은 SW 기초 수업의 내용을 구성하는 기준이 되거나 추후 진단도구 개발로 이어질 수 있기 때문에 비교적 상세하게 세부 지표

를 구성할 필요가 있다. 세부 지표를 선정할 때 이것이 학습 내용으로 구체적으로 활용되며 측정될 수 있는지를 면밀히 검토하여야 한다. 효과성 모형을 통해 SW 기초교육의 목적을 분명히 하고 이것이 수업을 통해 얼마나 달성되었는지 진단하는 과정은 SW 기초교육의 개선을 위해 중요하다.

6.3 SW 기초교육 효과성 모형

위의 시사점을 바탕으로 대학의 SW 기초교육의 효과성을 모형을 제안하면 <Table 3>과 같다.

SW를 전공하지 않더라도 SW 기초교육을 받는 대학생이라면 누구나 갖추기를 기대하는 결과는 컴퓨팅 사고력과 SW 문해력의 함양과 SW 인식 및 태도의 변화이다. 컴퓨팅 사고력은 분해, 패턴인식, 추상화 및 알고리즘

의 세부 영역으로 구성되며, 세부 지표는 (Fig. 3)을 기반으로 설명했던 각 개념 별 주요 활동을 기준으로 정의한다.

SW 문해력은 SW 교육을 통해 IT 기술문명 세계에 대한 넓어진 이해를 평가하기 위한 것으로 SW 및 정보 활용, IT 기술문명 이해, SW 미래기술 이해, 정보 보안 및 윤리로 세분화 한다.

SW 인식은 개인적, 사회적 차원에서 SW의 가치와 중요성을 인식하고, 교육의 필요를 느끼게 되는 것으로 개인적·사회적 중요성에 대한 인식 변화, SW 교육의 필요성에 대한 인식 변화를 살펴본다.

SW 태도는 SW에 대한 흥미 변화뿐만 아니라 자신감의 향상을 포함한다. 구체적으로 SW 흥미, 컴퓨팅 사고 효능감, SW 문해 효능감을 포함한다.

<Table 3> Effectiveness model of SW basic education

Categories	Definition	Metrics
Computational Thinking	Decomposition Decompose complex problems into several simple problems that can be solved	Decomposition of the scale Decomposition of the function Decomposition of the role Decomposition of the condition Consistency of Decomposition
	Pattern Recognition Explore patterns of similarities, trends, and rules shared by individual problems for efficient solution of problems	Constant trend recognition Common Attribute Recognition Iterative Rule Recognition
	Abstraction Eliminate unnecessary content from simplified problems or patterns and extract and generalize core content	Remove unnecessary content Core Content Extraction Simplification Generalization
	Algorithm The generalized problem-solving process is divided into stepwise, iterative, and conditional procedures.	Step-by-Step Procedure An iterative procedure Conditional procedures Clarify the process of problem solving Definition of detailed work
SW Literacy	Ability to systematically explore and understand the various phenomena of the IT technological civilization world from the SW perspective	SW and information utilization Understanding of IT Technology Civilization Understanding SW future technology Information security and ethics
SW Recognition	Recognize the value of SW in society and understand its importance and impact on personal life	Social significance Personal importance Need for education
SW Attitude	Interested and confident in the activities using SW	SW interest Effectiveness of computing thinking Effectiveness of SW literacy

6.4 모형의 타당성 검토

문헌고찰을 통해 도출한 모형은 총 3개의 4년제 대학에서 SW 기초교육을 담당한 경험이 있는 교수 총 13명을 대상으로 타당성 검토를 수행하였다. 모형의 타당성, 이해도, 설명력, 보편성, 유용성에 대해 5점 척도로 점수를 부여하게 하였다.

그 결과, <Table 4>와 같이 모든 항목에서 4.0이 넘었다. 상대적으로 점수가 낮은 영역은 이해도로 4.23이었고, 유용성이 4.46, 타당도, 설명력, 보편성은 동일하게 4.54였다. 신뢰도 계수로 확인한 평정자 간 일치도는 .829로 높은 편이었다. 내용타당도(Contents Validity Ratio: CVR) 값은 전문가 패널 수(13명)에 따라 최소 0.54 이상이면 적절하다고 판단할 수 있는데, 모든 항목에서 적절한 것으로 나타났다.

<Table 4> Review the results of validity

Category	Contents	Avg.	Std. Dev.	CVR
Feasibility	The structure and contents of the above model is valid as the effectiveness model of SW basic education.	4.54	0.660	0.85
Understanding	The terms and explanations used in the above models are easy to understand.	4.23	0.599	0.85
Explanation	The above model comprehensively explains the effectiveness of SW basic education.	4.54	0.660	0.85
Universality	The above model universally reflects the situation of many universities that offer SW basic education.	4.54	0.660	0.85
Usefulness	The above model is useful for explaining the effectiveness of SW basic education.	4.46	0.877	0.54

7. 결론

본 논문은 대학의 비전공자 대상 SW 교육이 무엇을 목적으로 어떻게 구성되어 있으며, 구체적인 내용은 무엇인지 선행 연구에 근거하여 논하고, 구체적인 지표가 포함된 SW 기초교육의 효과성 모형을 제안하였다.

이제까지 비전공자 대학 SW 교육이 필요하다는 공감대에 비해 무엇을 가르쳐야 하는가에 대한 논의는 충분치 않았다. 본 논문에선 비전공자 대상 SW 교육은 목표에 따라 기초교육과 융합교육으로 구분하였다. 미래사회를 살아갈 대학생들은 전공과 상관없이 누구나 SW를 이해하고 컴퓨팅 사고를 활용할 수 있도록 하는 SW 기초교육을 받아야 하지만, SW의 구체적인 방법론인 프로그래밍을 배우는 것은 전공에 따라 개인에 따라 선택할 수 있어야 한다. 즉, 학생들은 SW 기초교육을 의무적으로 수강한 후 개인이나 전공의 필요에 따라 SW 융합교육을 추가적으로 선택할 수 있도록 해야 한다.

비전공자 대상 SW 교육은 다양한 배경지식을 가진 구성원이 SW 교육 목적과 범위에 대해서 제대로 이해하였을 때 교육의 효과가 제대로 발휘될 수 있다. 이 연구에서는 SW 교육의 핵심 목표로 언급되는 컴퓨팅 사고 기반 문제해결 과정의 매커니즘과 분해, 패턴인식, 추상화, 알고리즘의 단계별 내용을 구체적으로 제시하였다. 이는 SW 기초교육을 담당하는 교수는 물론, SW 교육을 받는 학생과 해당 학과의 교수에게 SW 교육의 목적과 내용을 이해하도록 함으로써 상호 원활한 소통을 유도하는데 도움이 될 것이다.

대학에서 기초교양으로 SW를 가르치는 목적은 학생들이 미래 기술기반 사회를 준비하도록 하기 위함이다. 학생들은 디지털 자료를 효과적으로 활용하는 정보 문해를 넘어 변화하는 기술기반 사회에 적극적으로 대응할 수 있는 조망 능력을 필요로 한다. 이를 정보 문해를 확장한 SW 문해력으로 개념화하였다는 점도 의미가 있다.

SW 기초교육 효과성 모형은 구성요소들의 개념 뿐만 아니라 구체적인 지표를 제시하고자 노력하였다. 이는 SW 기초교육의 내용을 일관되게 구성하는데 도움이 될 수 있다. 더 나아가 측정 지표로 개발됨으로써 SW 기초교육의 효과를 측정하는데 기여할 수 있을 것이다. SW 기초교육은 대학의 모든 학생들을 대상으로 하기 때문에 많은 인적, 물적 자원이 소요된다. SW 기초교육

의 효과를 검증하는 것은 수업의 개선을 넘어 교육의 필요성을 교내외 구성원들과 공감하기 위해서도 중요하다. 모형에 기반 한 진단도구의 개발과 타당화는 후속 연구로 이루어질 것으로 기대한다.

비전공자를 대상으로 한 SW 교육이 5년째에 접어들면서 대학가에서 어느 정도 자리를 잡아가고 있지만, 이에 대한 학술적 논의는 사례 연구를 제외하고는 상대적으로 활발하지 않다. 이 연구가 비전공자 대상 SW 교육에 대한 학술적 논의를 활성화하는데 기여할 수 있기를 기대해 본다.

참고문헌

- [1] ChungSoo Na, Hyun Joo, JinJu Lee, DongSik Kim(2018). Inducing Computational Thinking in Korean SW Education: Synthesizing Standardized Mean Changes through Meta-analysis. *Journal of Educational Technology*, 34(3), 775-815.
- [2] CSTA & ISTE(2011). *Computational Thinking in K-12 education - teacher resources, 2nd edition*, CSTA & ISTE.
- [3] DaeSu Kim(2016). *Software and Computational Thinking*. Life and Power Press.
- [4] GeumJu Park, YungJoon Choi(2018). Exploratory study on the direction of software education for the non-major undergraduate students. *Journal of Education & Culture*, 24(4), 273-292.
- [5] HwanChul Lee, HyunSook Lee, JungSook Sung, SoonWon Jeong, SooHwan Kim, SeokHee Kim, JongHye Kim, JooYeon Park, ByungMo Lee(2015). *A Study on Surveying the Actual Conditions and Evaluating the Effectiveness of SW Education in Elementary and Secondary Schools*. KERIS(Korea Education and Research Information Service), 15-JinHeung-077.
- [6] HyoMin Park(2014). *Global Trend of SW Education Status and Educational Methodology*. KISA Internet & Security Focus 2014-9. 40-56.
- [7] HyunSeok Shin, YoungHyio Joo, Yoon Jung Yang, SuHyeon Jung(2016). Analyzing Trends of Literature Review Methods in 『The Korean Journal of Educational Administration』. *Korean Journal of Education Administration*, 34(4), 1-24.
- [8] InHwan Yoo(2016). Software Engineering: A Study on SW Development Process for Increasing Computational Thinking. *KIPS Transactions on Software and Dta Engineering*, 5(2), 51-58.
- [9] JaeKyung Kim(2017). Development of Rubric for Assessing Computational Thinking Concepts and Programming Ability. *The Journal of Korean Association of Comuter Education*, 20(6). 27-36.
- [10] Jeannette M. Wing(2006). Computational Thinking, *Communication of ACM*, 49(3), 33-35.
- [11] JeongEun Nah(2017). Software Education Needs Analysis in Liberal Arts. *Korean Journal of General Education*, 11(3), 63-89.
- [12] JinSook Kim, SunKwan Han, SooHwan Kim, SoonWon Jung, JaeMyung Yang, UiDeok Jang, JungNam Kim, MiYoung Ryu, JinTae Lee, SooJin Jeon, SangHong Kim(2015). *Development of Teaching and Learning Model for SW Education*. Korean Education Development Institute and Korea Education & Research Information Service.
- [13] JooYoung Seo(2017). A Case Study on Programming Learning of Non-SW Majors for SW Convergence Education. *Journal of Digital Convergence*, 15(7), 123-132.
- [14] JooYoung Seo, SeungHun Shin, EunHee Goo(2018). A Study on Non-Majors Students' Perception of the SW Liberal Education in University, *Journal of Digital Convergence*, 16(5). 21-31.
- [15] JungCheul Oh, JongHoon Kim(2016). A Development of a Puzzle-Based Computer Science Instruction Model and Learning Program to Improve Computational Thinking for Elementary School Students. *The Journal of Fisheries and Marine Sciences Education*, 28(5), 1183-1197.
- [16] Korea Institute for Curriculum and Evaluation(2015). *Information in the National Curriculum of Korea 2015*.

- [17] KyungMin Kim(2017). A Study on Information Literacy Education for Enhancing Computational Thinking. *Journal of Korean Association of Computer Education*, 20(4), 59-66.
- [18] MinSeok Kim(2018). *Status and Implications of Domestic and International SW Education for Human Resource Development*. 4th Industrial Revolution and Soft-power Issue Report 2018-53, National IT Industry Promotion Agency, Korea.
- [19] MinWon Seo(1994). Conceptionalization and Measurement of College/University Educational Effectiveness: Exploring Conceptual Framework and its Application. *Korean Journal of Higher Education*, 6(2). 97-136.
- [20] OakYoung Han, JaeHyoun Kim(2017). A Study on the Improvement in Computational Thinking for Nonmajors Through Software Education. *Proceedings of The Korean Association of Computer Education*, 21(1), 139-141.
- [21] SooHwan Kim(2015). Analysis of Non-Computer Majors' Difficulties in Computational Thinking Education. *The Journal of Korean Association of Computer Education*, 18(3), 49-57.
- [22] SungHoon An, JongMin Kim, TaeOak Song, CheolHyun Lee, SungHyun Kim, JungHwan Lee(2017). *Research on Analysis of the Effectiveness for SW Education Research School in 2016*. KERIS(Korea Education and Research Information Service), KR 2016-4.
- [23] WonGyu Lee, JaeMyung Yang, JaMi Kim, JungHee Seo(2017). *Research on Tool Development for Measuring SW Education Effectiveness in 2017*. KERIS(Korea Education and Research Information Service), CR 2017-8.
- [24] YongMin Kim, JongHoon Kim(2015). Development and Application of Software Education Program of App Inventor Utilization for Improvement of Elementary School Girls' Computational Thinking. *Journal of The Korean Association of Information Education*, 19(4), 385-398.
- [25] YoungHo Seo, MirYeong Yeom, JongHoon Kim(2016). Analysis of Effect that Pair Programming Develops of Computational Thinking and Creativity in Elementary Software Education. *Journal of The Korean Association of Information Education*, 20(3), 219-234.
- [26] YoungJoon Nam, SoonHee Kim, SangWon Jung, JungA Shin, JungSim Kim(2010). *A Study on the Development and Management of Digital Information Education Program*. The National Library of Korea.
- [27] YoungMan Ko(2004). A Study on Curriculum Design for the Information Literacy in Higher Education. *Korea Society for information Management*, 21(4), 67-87.
- [28] Google(2019). Exploring Computational Thinking. <http://goo.gl/oG0w56>
- [29] IITP, Korea(2019). Human Resource Development through the National Program for Excellence in SW. <https://www.iitp.kr/kr/1/notice/reportAndClarify/view.it?ArticleIdx=3561&count=true>
- [30] Korean National Institute for General Education(2019). Standard Model for College Liberal Arts Education. http://konige.kr/files/sub0208/Konige_Standard_Model.pdf
- [31] Ministry of Education, Ministry of Science, ICT and Future Planning(2015). Human Resource Development Plan for SW-centric Society. <https://www.msit.go.kr/SYNAP/skin/doc.html?fn=075964f3c916f54dc5b36afbd4bfd256&rs=/SYNAP/sn3hcv/result/201908/>
- [32] Ministry of Science, ICT and Future Planning(2015). Plan for the National Program for Excellence in SW. <https://www.msit.go.kr/web/msipContents/contentsView.do?cateId=mssw11211&artId=1272942>
- [33] MinSeok Lee(2018). Status and Crisis of University SW Education. <https://spri.kr/posts/view/22496>
- [34] National Institute of Korean Language(2019). Standard Korean Dictionary. <https://stdict.korean.go.kr/search/searchView.do>

[35] WooYong Kim(2018). <https://www.zdnet.co.kr/view/?no=20180209110815>

저자소개



홍 성 연

2009 서울대학교 교육학과 (교육학 박사)
2009 서울대학교 CTL 연구원
2010~2011 가톨릭대학교 CTL 책임연구원
2014년~ 현재 아주대학교 다산학부대학 부교수
관심분야 : 학생지원, 역량, 질 관리, 효과성
E-Mail : hsy06@ajou.ac.kr



서 주 영

2009 이화여자대학교 컴퓨터공학과(공학박사)
2009 이화여자대학교 컴퓨터공학과 연구교수
2009~2015 아주대학교 소프트웨어학과 강의교수
2016~현재 아주대학교 다산학부대학 교수
관심분야 : SW공학, SW테스트, SW융합교육
E-Mail : jyseo@ajou.ac.kr



구 은 희

2009 단국대학교 전자컴퓨터공학과(공학박사)
2011~2013 서일대학교 정보통신

공학과 강의교수

2013~2014 (주)도넛시스템 LSI 이미징사업부 책임 연구원
2014~2016 (주)이너트론 이동통신연구소 수석연구원
2016~현재 아주대학교 다산학부대학 교수
관심분야 : 정보보호, 암호 알고리즘, 서비스로서의 보안 (ASCaaS), 소프트웨어 교육
E-Mail : ehgoo@ajou.ac.kr



신 승 훈

2011 아주대학교 정보통신공학과(공학박사)
2011~ 2015 아주대학교 소프트웨어융합학과 강의교수
2016~ 현재 아주대학교 다산학부대학 교육중점교수
관심분야 : SW공학, SW테스트, 컴퓨터 네트워크
E-Mail : sihsh@ajou.ac.kr



오 하 연

2013 서울대학교 컴퓨터공학과(공학박사)
2013 ~ 2016 숭실대학교 IT대학 전자정보공학부 교수
2010 U.C. Berkeley 방문 연구원
2016년~ 현재 아주대학교 다산학부대학 교수
관심분야 : 소셜정보망, 데이터분석 및 인공지능
E-Mail : hyoh79@gmail.com



이택균

2010 State University of New York at Buffalo, Computer Science and Engineering (공학박사)

2011 ~ 2013 아주대학교 소프트웨어학과 연구교수

2014 ~ 2016.8 아주대학교 소프트웨어학과 강의교수

2016.9~ 현재 아주대학교 다산학부대학 교수

관심분야 : SW융합교육, SW기초교육

E-Mail : taeklee@ajou.ac.kr