

창의 · 컴퓨팅사고 교육내용 기본 설계 연구

오경선^{1*}, 서응교², 정혜진¹
¹단국대학교 SW중심대학사업단, ²단국대학교 교양교육대학

A study on development of educational contents about combining computational thinking with design thinking

Oh-Kyung Sun^{1*}, Suh-Eung Kyo², Haejin Chung¹
¹SW-Centric University Project Group, Dankook University
²School of General Education, Dankook University

요약 본 연구는 4차 산업혁명을 맞이한 현 시대에서 요구하는 창의 · 융합형 인재를 기르기 위해 대학에서 진행되는 비전 공자 대상 소프트웨어교육에 필요한 교육내용을 제안하는데 그 목적이 있다. 우선 선행연구를 통해 디자인씽킹 개념과 컴퓨팅사고요소간의 관계를 통해 창의적 문제해결능력을 길러내기 위한 소프트웨어교육 방향을 알아보고 디자인씽킹 기반의 컴퓨팅사고를 위한 두 개 수준별 교육내용요소들을 도출하고자 한다. 이를 위해 두 차례에 걸친 전문가 설문 조사와 포커스 그룹 인터뷰를 통해 내용들을 분석하였다.

연구 결과 1수준에서는 확산적 사고에 초점을 두고 디자인씽킹의 프로세스를 기반으로 전체 내용을 구성하되 프로토타입을 제작하는 프로세스부터는 문제해결에 필요한 컴퓨팅사고 내용들을 도출하였다. 2수준에서는 수렴적 사고에 초점을 두고 1 수준의 내용인 다양한 아이디어와 스토리를 컴퓨팅으로 구체화하여 실현하기 위한 컴퓨팅사고 내용들을 도출하였다. 본 연구를 통해 대학에서의 비전공 대상의 SW기초교양으로서 교육내용과 방법을 개발하는데 기초자료로 활용될 것으로 기대한다.

주제어 : 디자인씽킹, SW교육, 프로그래밍교육, SW교육모델, 컴퓨팅사고

Abstract The objective of this study is to suggest the contents of education for software for Undergraduates who are not majoring in IT to cultivate creative and convergent people to prepare for the 4th Industrial Revolution. In the background, this study investigated the relationship between design thinking and computing thinking factors and examined the direction of software education to develop creative problem solving abilities.

Two specialist questionnaires and focus group interviews were used to derive two levels of content elements for design thinking based computing thinking. According to the results of the analysis, This research is based on the process of design thinking, focusing on diffuse thinking at the first level, and the whole contents are composed. The process of producing a prototype was used to derive the computational thinking contents necessary for problem solving. This study focuses on convergent thinking at the second level and draws out the contents of computing thought to embody the first level contents of various ideas and stories into computing. It is expected that this study will be used as basic data to develop the content and method of education as SW education for non - major subject in university.

Key Words : Design thinking, SW education, Coding education, SW education model, Computational Thinking

*This research was supported by the MIST(Ministry of Science and ICT), Korea, under the National Program for Excellence in SW supervised by the IITP(Institute for Information & communications Technology Promotion) (2017-0-00091)

*Corresponding Author : Kyung-Sun Oh(skyl@dankook.ac.kr)

Received March 14, 2018

Revised April 13, 2018

Accepted May 20, 2018

Published May 28, 2018

1. 들어가는 말

세계 각 국에서는 최근 모든 산업이 소프트웨어 중심으로 변화함에 따라 소프트웨어 (Software: SW) 교육의 중요성을 인식하고 국가 차원의 교육과정을 마련하여 SW 교육을 강화하고 있다. 컴퓨터 과학적 원리를 토대로 효율적으로 문제를 해결 할 수 있는 사고능력을 의미하는 컴퓨팅 사고(Computational Thinking, CT)를 미래의 핵심역량으로 보기 때문에 SW교육을 강조하고 있는 것이다[1]. 뿐만 아니라 급변하는 사회에 대응하기 위해 창의적으로 문제를 해결하는데 있어서 컴퓨팅사고력을 적용할 수 있는 인재를 확보하는 것이 곧 국가경쟁력을 좌우하게 될 것으로 예상하고 있다[2,18]. 변화하는 시대에 발맞추어 세계 여러 나라는 컴퓨팅사고력 기반의 교육과정을 도입하고 있다.

국가교육과정을 운영하고 있는 영국은 2014년 9월부터 만 5-16세의 모든 학년이 '컴퓨팅(computing)' 과목을 필수로 지정하였다. 프랑스 교육부는 2016년 9월부터 SW과목을 중학교 정규 과목으로 지정함으로써 전통적으로 인문학 교육을 강조해 왔지만 세상이 SW중심으로 급속히 바뀌는 현실을 반영하였다. 핀란드의 경우는 2016년 가을학기부터 모든 초등학교 학생들이 프로그래밍을 필수로 이수하도록 했다. 각 주에 따라 상이한 교육과정을 가진 미국은 고등학교를 중심으로 컴퓨터과학(Computer Science, CS)과목을 정규 교육과정으로 채택하려고 노력하고 있다. 최근 세계 각국에서 시작하고 있는 컴퓨터교육은 SW중심의 디지털 산업사회 패러다임의 변화에 따른 새로운 인재역량 교육이 필수적이기 때문이라고 볼 수 있다[3].

우리나라의 경우 2015년 SW 교육운영지침과 2015년 개정 교육과정을 통해 SW 교육을 강조하기 시작하였고 그에 따라 2018년부터 중학교에 정보과목을 통한 최소 34시간이상의 SW 교육이 필수로 도입되며 2019년부터는 초등학교 5~6학년 학생들에게 17시간의 SW 교육이 의무화된다. 또한, 최근 SW 중심대학들을 통해 IT비전공 학부생들을 대상으로 대학 내 SW 기초교육을 확대하여 SW 기초 교육을 의무화하고 있다[1].

여기서 CT의 교육 목적을 생각해 볼 필요가 있다. CT의 교육의 목적은 창의적으로 문제 해결을 하는데 있어서 컴퓨팅 사고를 기반으로 효율적인 문제 해결을 찾는 것이다. 그렇다면 한다면 문제해결 절차와 교육내용이

컴퓨팅사고를 접목하여 창의력을 기를 수 있도록 역량중심으로 구성해야한다는 점이다. 무조건적인 프로그래밍 학습으로 문제를 해결하는 것이 아니라, 인간중심의 문제를 시작으로 해결의 실마리를 찾는 프로세스 안에서 CT를 접목하여 효율적인 문제 해결을 찾아 구현할 수 있는 교육 내용이 필요 있는 것이다. 이러한 교육내용은 학습자들의 맥락에서 문제를 발견하게 함으로 소프트웨어 교육에 동기 부여 뿐 아니라 학습에 적극적 동참을 할 수 있도록 한다[19,21].

따라서 본 연구는 비전공 학부생들의 SW교육에 있어서 단순히 코딩에 필요한 지식 습득과 능력보다는 문제 해결을 하는데 있어서 학습자 중심에서 해결의 실마리를 찾아가는 디자인씽킹(Design Thinking : DT) 프로세스로 확산적 사고를 유도하면서 구체적인 SW를 구현하도록 CT를 접목하여 창의력을 기를 수 있는 역량중심의 교육내용을 설계하고자 한다. 이 연구를 위해 제일 먼저 DT 프로세스 중 CT요소를 가지고 '무엇을 배워야하는지'에 관해 관련 자료와 전문가들의 의견을 토대로 내용을 분석하여 추출한다. 추출한 내용요소를 가지고 역량중심의 비전공자 대상의 창의·컴퓨팅 문제해결을 위한 SW교육을 개발하는데 방향을 제시할 수 있을 것으로 기대한다.

2. 이론적 배경

2.1 디자인씽킹(Design Thinking : DT)

애플(Apple)이 출시한 최초의 마우스, 마이크로소프트(Microsoft)의 두 번째 마우스, 팜 V PDA(Palm V PDA), 스틸케이스의 LeapChair개발한 미국의 이노베이션 기업인 아이데오의 팀 브라운은 디자인씽킹에 대한 정의를 '디자이너가 생각하는 방식으로 문제를 해결하는 방법으로 디자이너들이 자신의 디자인과정에서 최종 결과물을 만들어내기 위해 전 과정 안에 공감, 문제의 정의, 아이디어 발상, 아이디어 정교화, 실행 및 평가와 피드백 등 디자인 요소들을 선택 하고 편집하여 유용하게 사용하는 일련의 사고과정'이라고 하였다[4]. DT을 교육에 적용한 스탠포드 대학교에서는 '교육자에게 디자인 사고란 교실, 학교, 지역 사회에서 의미 있는 해결책을 설계하는 것'이라고 정의하였다[5]. 이러한 DT의 특성을 종합하여 이정열, 이주명은 모든 사람은 디자이너이고, 인

간의 전반적인 제반 문제를 시작으로 협업을 통하여 문제가 해결할 수 있도록 하고 더 나아가 다양한 학문 간의 융합을 통해 해결하는 과정이라고 하였다[6].

DT 프로세스는 문제 해결을 위한 아이디어를 확산적인 사고에 의해 여러 가지 해결방안을 도출하여 선택하고 적용하며, 그것을 평가하는 과정을 통해 최적의 아이디어를 실현하는 과정이라고 할 수 있다[7]. IDEO가 제안한 디자인 프로세스와 스탠포드 대학교 d-school에서의 DT 절차를 살펴보면 다음과 같다[8,9].

Table 1. Design Thinking Process

IDEO & d-school	Explanation
Empathy	Sympathy is to look at a problem by placing the center of the problem on people
Problem Definition	Problem definition is a step for finding the most core and essential problem by defining the essence of a problem
Ideas	Idea creation is a step for deriving the optimal idea by connecting, combining, transforming, and modifying various ideas coming out through collaboration
Prototype	Prototyping is a step to produce a model from the derived ideas
Test	Testing is a step in which the created model is actually applied

2.2 컴퓨팅사고(Computational Thinking : CT)

CT의 교육의 목적은 창의적으로 문제 해결을 하는데 있어서 컴퓨팅 사고를 기반으로 효율적인 문제 해법을 찾는 것이다. CT를 유 · 초 · 중학교에 효과적으로 적용할 수 있는 기반을 마련하고자 미국의 National Standards for K-12 Computer Science 에서는 CT의 조작적 정의를 다음과 같이 제시하였다[12].

- 컴퓨팅으로 해결할 수 있는 정형화된 문제
- 논리적으로 자료를 조직하고 분석하기
- 시뮬레이션과 모델들처럼 추상화를 통해 데이터를 표현하기
- 알고리즘 사고를 통한 자동화 솔루션
- 식별과 분석 그리고 가장 효과적이고 효율 으 로 자원 과 단계를 조합하여 목표달성을 할 수 있는 해결책 을 구현
- 일반화

이러한 조작적 사고인 CT를 9가지 조작적 요소(자료 수집, 자료표현, 자료분석, 문제분해, 추상화, 알고리즘, 자동화, 병렬화, 시뮬레이션)로 구분하여 다음과 같이 설명하였다[12,13].

Table 2. Computational Thinking

Computing Thinking Factors	Explanation
Collecting data	Ability to gather information
Data representation	Ability to visualize and organize data with appropriate graphs, charts, words, images, etc.
Data analysis	Ability to create meaning with collected data, discover patterns, draw conclusions
Problem resolution	Ability to work with small units that can manage big problems
Abstract	Ability to tune or handle complexity by eliminating parts that are common to many objects and that are not relevant to other objects
Algorithm	Ability to list steps in order to solve problems or achieve certain results
Automation	Ability to perform repetitive or tedious tasks with a computer or machine
Parallelism	At the same time, the ability to organize resources to perform tasks that reach a common goal
Simulation	Ability to predict 'What if' through process modeling and representation

당면한 어떤 문제를 해결하는 과정에서 위의 표에서 제시한 CT의 여러 조작적 요소들을 경험 할 수 있다. 이러한 조작적 요소들 가운데 Wing(2006;2008)은 문제를 해결 가능한 형태로 표현하기 위한 사고과정인 추상화 (abstracting)와 이를 통해 제시한 아이디어를 자동화 (automating)를 CT의 핵심 조작적 요소로 설명하였다. 위의 내용을 종합하여 보면 CT 기반의 문제해결과정에서의 ‘문제’는 컴퓨팅으로 해결할 수 있어 명확하고 모호 하지 않는 특성을 가지고 있다고 할 수 있다[14].

2.3 창의적 문제해결과정

송해덕(2007)은 ‘창의적 문제해결력의 구성요인과 교수설계원리의 탐색’의 연구에서 일반적 문제해결과정은 현재 상태와 목표 상태를 파악한 후 목표 상태로 가기 위한 해결방안을 마련하는 과정이지만 창의적문제해결과정은 문제를 해결하는 과정에서 아이디어를 발견하고 다양한 해결방안들을 찾아내면서 확산적사고와 수렴적 사고가 동시에 활성화 되어 진행된다는 점에서 차이가 있다고 하였다[17]. Isaksen과 Treffinger(1985)또한 문제 해결하는 과정에서 수렴적사고와 확산적사고가 작용하여 해결안을 만들어가는 과정을 창의적 문제해결이라 하였다[10]. 한순미(2004)는 연구를 통해 창의적 문제해결은 인지적사고기능에서 확산적사고와 수렴적사고가 작용하여 나타난다고 하였다. 창의적 문제해결을 종합하여 정의하여 보면 수렴적사고와 확산적사고의 작용을 통해 이뤄진다고 할 수 있다[11].

2.4 비전공자를 위한 국내 SW교육 현황

SW교육의 목적은 정답을 맞추는 것이 아니라 생각하는 방법을 알아 문제를 창의적으로 해결하는데 있어서 CT가 바탕이 되도록 하는 것이다. 4차 산업혁명에 적극적으로 대응하기 위해 CT기반의 창의적 문제해결능력의 중요성이 대두됨에 따라 중학교는 2018년부터 최소 34시간이상의 SW교육을 필수로 하고 초등학교는 2019년부터 17시간의 SW교육을 의무화한다. 또한, 최근 SW 중심 대학으로 선정된 대학들은 IT비전공 학부생들을 대상으로 대학 내 SW 기초 교육을 필수적으로 하고 있다. 그러나 SW 중심대학 사업에 선정된 대학들의 SW기초교육은 특정 프로그래밍 언어(예: Python, 스크래치, 앱인벤터)의 커리큘럼으로 구성되어있으며, 코딩 문법 학습과 예제풀이가 대다수이다. 하지만 IT비전공자 학생들이 SW학습을 하는데 어려움을 느끼는 사고는 프로그래밍 언어교육을 통해 더욱 심화되기에 비전공자 대상의 SW 교육 내용에 대한 연구가 필요한 실정이다[13,22].

3. 연구방법

창의적 문제해결능력을 창의적 문제해결 역량이라는 관점으로 보고 내용을 개발하고자 하였다.

비전공자 학부생 대상의 SW교육 목적은 단순히 코딩만을 익혀 반복적인 기계학습이 아니라 다양한 문제들 가운데 창의적인 사고를 바탕으로 논리적 수렴과정을 거쳐 SW코딩을 통해 실제 구현할 수 있는 DT 기반의 CT역량인 창의적 문제해결능력 함양이다. 특히 창의적문제 해결과정은 선행연구를 통해 확산적 사고와 수렴적 사고를 통해 이뤄진다는 것을 알 수 있다. 따라서 비전공자 대상의 SW교육은 학습자 중심에서 문제를 발견하여 창의적으로 해결하는 DT기반의 CT역량을 기르기 위해 ‘무엇을 가르쳐야하는가?’ 에 초점을 두고 실천적 SW능력을 함양할 수 있는 교육 내용 구성에 관한 연구를 하고자 하였다.

제일 먼저 창의적 문제 해결 능력을 길러내기 위해 DT 프로세스를 선택하고 이 과정 안에서 CT를 적용하는 DT SW교육모형을 정의하였다. 전문가 1차 측정도구를 개발하기 위한 창의적 문제해결능력을 향상시킬 수 있는 단계를 도출하기 위해 국내·외 관련 연구 자료와 단국대학교 SW교양필수과목인 ‘창의적사고와 코딩’을 기반으로 Fig. 1, Table 4와 같이 정의하였다.

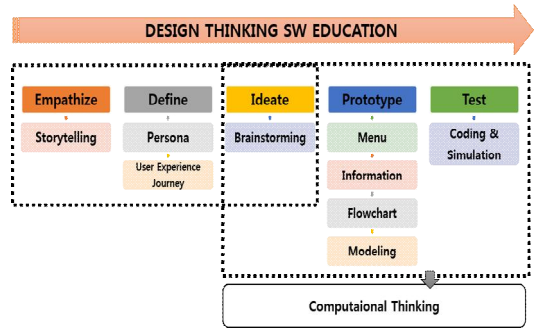


Fig. 1. Design Thinking SW Education Process

Fig. 1의 프로세스 중 3단계인 ‘아이디어 만들기’부터 CT 역량을 접목하여 문항의 초안을 개발하였다. 여기서 CT 역량은 인간의 사고와 관련 있는 추상화, 이것을 절차화한 알고리즘적 사고를 가지고 효율적으로 자동화하고, 시뮬레이션을 통해 테스트 할 수 있는 내용으로 축약하였다. 또한 1수준 단계에서는 DT 프로세스에 대한 이해 및 적용과 이 프로세스 안에서의 간단한 CT의 이해 및 적용을 통해 창의적으로 문제를 해결할 수 있는 확산적 사고에 초점을 두었다. 그리고 2수준 단계에서는 창의적 아이디어를 SW로 개발하기 위해 필요한 CT(추상화, 알고리즘, 자동화, 시뮬레이션)에 무게를 두어 수렴적 사고를 할 수 있는 방향으로 문항 초안을 제시하였다.

Table 3. DT&CT Generalized knowledge

Factor	Generalized knowledge
DT	It is a creative problem solving methodology devised to actively solve problems and maximize influential changes
CT	Computational thinking is the thought processes involved in formulating a problem and expressing its solution(s) in such a way that a computer-human or machine-can effectively carry out ¹⁾

DT SW교육 모형을 기반으로 창의적인 CT역량을 길러내기 위한 내용을 추출하기 위한 본 연구를 위해 두 차례에 걸친 전문가 조사를 실시함으로 합의점을 도출하여 타당성을 검증하였다.

본 연구 내용 타당성 설문 조사에 참여한 14인 중 SW관련 업계 전문가는 경력 10년 이상으로 제한하였고 대학의 교육현장에서 IT비전공대상 SW관련 교과를 담

1) Wing, Jeannette (2014). "Computational Thinking Benefits Society". 40th Anniversary Blog of Social Issues in Computing.

당하고 있는 대학교수는 컴퓨터교육 전공 박사학위 소지자로 5년 이상의 관련 업계 경력자로 제한하였다. 또한 SW교육 현업 전문가들 중 컴퓨터교육 석사 학위 소지자의 경우 10년 이상의 경력자로 제한하였다. 이들을 대상으로 모바일과 인터넷을 이용하여 조사하였으며 각 단계별 참여 인원과 응답자 수는 다음과 같다.

Table 4. General Properties of Research Participants

Factor	1st		2st	
	Distribute (persons)	Response (persons)	Distribute (persons)	Response (persons)
Computer Education	5	5	5	4
Computer Science	9	8	8	6

DT SW교육 모형 단계와 정의를 통해 작성한 문항에 대한 내용타당성을 검증할 필요가 있었다. 각 문항에 대하여 적합 또는 부적합을 선택하는 폐쇄형 질문과 개방형 질문을 함께 하는 1차 전문가 설문조사를 하였다. 이후 2차 설문 문항개발은 1차 전문가 설문응답을 분석하여 작성하였다.

Table 5. Composition of Questionnaire

Level	Factor	Questionnaire		Question Contents
		1st	2st	
1	DT	7	7	<ul style="list-style-type: none"> • Society members • Empathy • Problem definition • Explore ideas • Convert ideas • Select a solution
	CT	6	6	<ul style="list-style-type: none"> • Prototyping • Core element extraction • Algorithm design • Automation • Testing and evaluation
2	DT	3	3	<ul style="list-style-type: none"> • DT Process • Persona analysis • Create flowcharts based on user experience
	CT	12	12	<ul style="list-style-type: none"> • CT concepts & elements • Flowchart • Input and Output • Variables and Operations • Control structure • Screen Design • Algorithm design • Coding • Test • Analysis and evaluation

설문조사는 리커트 7점 척도로 실시하였다. 그 결과를 가지고 측정하고자 하는 내용들이 각 역량에 적합한지 알아보기 위해 타당도 분석을 실시하였으며, 측정하는

내용들의 신뢰도는 Cronbach's α 값으로 검증하였다. 이상의 분석을 위하여 한글 PASW 18.0을 사용하였다.

4. 연구결과와 해석

전문가 조사 방법에서는 신뢰도, 내용타당도, 안정도 값으로 검증한다. 본 연구에서 전문가 설문조사방법으로 진행함에 따라 타당도는 Lawshe(1975)이 제시한 내용 타당도 비율(CVR)로 판별하고 신뢰도는 문항 내적 일관성 신뢰도(Cronbach's α)로 판별한다. 또한 반복되는 문항에 대한 일치성을 알기 위해 표준편차를 산술평균으로 나눈 변이계수를 사용하는 안정도는 로 판별한다 [15,16]. 이때 변이계수가 0.5이하인 경우에는 합의가 이루어진 것으로 판단해 추가 설문이 필요하지 않음을 의미한다. 만약 0.8이상인 경우에는 추가적인 설문조사를 실시해야함을 의미한다.

본 연구에서는 1차 설문의 경우 13명의 전문가이기 때문에 13명을 기준으로 CVR값이 최소 0.54 값 이상이면 타당하다고 보았다. 또한 문항 내적 일관성 신뢰도의 값이 사회과학분야에서는 0.6 이상이면 신뢰도가 있다고 보기 때문에 응답분석결과 0.6이상인 문항들만 선정하였다.

2차 설문의 경우 전문가들의 의견을 종합하여 내용을 수정하여 문항을 구성하여 실시했다. 따라서 2차 설문 10명의 응답자를 기준으로 내용 타당 비율(CVR) 값이 최소 0.62 값 이상이 되는 문항과 문항의 안정도를 확보하기 위한 변이계수가 0.5 이하의 값을 가지면서 문항 내적 일관성 신뢰도의 값이 0.6 이상인 문항을 최종 선정하였다 [15].

4.1 1차 전문가 설문 결과 분석

1차 설문의 결과 폐쇄형 응답의 경우 모든 문항이 0.54 이상으로 내용타당도를 확보했고 신뢰도 또한 모든 문항이 Cronbach's α 값이 0.95이상으로 문항의 내적 일관성을 확보했다.

1수준에서는 창의적으로 문제를 해결하는데 필요한 사고과정을 디자인씽킹을 통해 배우는 내용들로 구성하였다. 이에 전문가들은 디자인씽킹에 대한 내용요소가 적절하다고 판단하였다. 특히 디자인씽킹의 단계 중 프로토타입 제작 부분에서는 무엇보다도 컴퓨팅으로 구현이 가능한 것을 판별하기 위해 핵심요소추출을 컴퓨팅사고로 보는 것을 적합하다고 판단하였다.

Table 6. Results of the first survey response

Level	Factor	Content Element	CVR	Cronbach's α
1	DT	Understanding DT	1.00	.961
		sympathy	1.00	.961
		Problem Definition	1.00	.959
		Explore ideas	1.00	.959
		Converting Ideas	0.85	.958
		Select a solution	0.85	.957
		Understanding the problem	0.85	.959
	CT	Abstraction	0.85	.959
		Algorithm design	0.85	.978
		Algorithm Selection	0.67	.956
		Coding	0.69	.956
		Test	0.69	.956
		Evaluation	0.69	.956
2	DT	DT Process	0.85	.923
		sympathy	0.85	.922
		Problem Definition	0.85	.923
	CT	Algorithm Thinking Process	0.85	.924
		Understanding the algorithm	1.00	.941
		Algorithm design	0.85	.923
		Execution of the algorithm	0.85	.923
		Input and Output	1.00	.942
		Variables and Operations	1.00	.942
		Control structure	1.00	.942
		Screen design	0.85	.922
		Algorithm design	0.85	.922
		Coding	1.00	.940
		Test	1.00	.942
		Analysis and Evaluation	0.85	.942

2수준에서는 창의적으로 해결 가능한 문제를 찾아 아이디어를 탐색하는데 있어서 컴퓨팅으로 구현이 가능한가에 대한 판별을 시작으로 Level1에서와는 다른 좀 더 논리적이고 컴퓨터과학적인 접근의 내용요소가 적합한 것으로 나타났다.

1수준에서 발산적 사고를 위해 알고리즘 선택이나 구현, 테스트, 평가에 대한 항목은 다른 내용요소에 비해 타당도가 낮게 나왔다.

컴퓨팅사고 내용 요소에 대한 대다수의 검토의견은 2수준에서 다뤄야하는 발산적 사고에 대한 목적과 컴퓨팅 사고가 가지고 있는 수렴적 사고에 대한 상치되는 개념으로 인한 것이다. 이외에도 문제이해의 내용 요소에 대한 검토의견을 반영하여 최종적으로 2차 설문 문항을 작성하였다.

Table 7. Review opinion and reflection (1st)

Review comments	Reflect review
<ul style="list-style-type: none"> Algorithm selection, implementation, testing, and evaluation at Level 1 require approach to problem solving based on design thinking. Algorithm design seems to be somewhat difficult to deal with divergent thinking 	<ul style="list-style-type: none"> Algorithm selection involves choosing a logical solution based on creative ideas it means. This will be driven by team collaboration activities. The programming implementation is to solve the best ideas in a storytelling programming language. It is not unreasonable to test and evaluate ideas that have been solved through storytelling.
<ul style="list-style-type: none"> To understand the problems through empathy and problem definition. Therefore, the 'Understanding the Problem' content element is unnecessary. 	<ul style="list-style-type: none"> 'Understanding the Problem' content element removed.

4.2 2차 전문가 설문 결과 분석

2차 설문 응답결과는 모든 문항이 1차 결과와 같이 0.54이상으로 내용타당도를 확보했다. 특히 1차 설문조사의 검토의견을 통해 전문가 인터뷰를 진행하였기 때문에 설문 내용의 이해가 높아져 내용타당도가 0.69의 값들이 0.80으로 증가한 경향을 알 수 있다. 또한 Cronbach's α 값이 0.95이상으로 모든 문항의 내적 일관성을 확보했다. 2차 설문 결과 후 계속해서 설문이 필요한가에 대한 판별을 제시해주는 변이계수가 < .5 조건을 만족한 항목들이 안정도가 있다고 보고 이러한 항목들을 추출하였다.

2차 항목에 대한 검토 의견에서는 1수준에서의 발산적 사고에 대한 목적에 해당하는 DT의 세부적인 내용의 개발을 요구했고 이러한 사고를 길러내기 위한 프로그래밍 언어의 신중한 선택을 요구하기도 하였다.

또한 2수준에서는 수렴적 사고에 대한 목적에 해당하는 컴퓨팅사고의 세부적인 내용의 개발, 전공별 특성에 맞는 예제들의 활용을 함께 개발 할 것을 제안하였다. 이러한 검토의견들은 내용요소가 만들어지고 내용체계를 구성한 후 세부적인 내용을 개발하여 현장에 적용하기 위한 교재 개발에 검토의견을 반영하기로 하였다.

Table 8. Results of the second survey response

(* C : CVR, S : Stability)

Level	Factor	Content Element	C	Cronbach's α	S
1	DT	Understanding DT	1.00	.982	0.12
		sympathy	1.00	.981	0.10
		Problem Definition	1.00	.978	0.12
		Explore ideas	0.80	.977	0.15
		Converting Ideas	0.80	.977	0.15
	CT	Select a solution	0.80	.976	0.16
		Abstraction	1.00	.978	0.12
		Algorithm design	0.8	.976	0.20
		Algorithm Selection	0.80	.978	0.25
		Coding	0.80	.978	0.25
		Test	0.80	.978	0.25
2	DT	Evaluation	0.80	.978	0.25
		DT Process	1.00	.971	0.25
		sympathy	1.00	.974	0.16
	CT	Problem Definition	0.80	.969	0.12
		Algorithm Thinking Process	0.85	.971	0.10
		Understanding the algorithm	1.00	.973	0.10
		Algorithm design	1.00	.971	0.09
		Execution of the algorithm	1.00	.971	0.11
		Input and Output	1.00	.971	0.13
		Variables and Operations	1.00	.969	0.13
		Control structure	1.00	.969	0.11
		Screen design	1.00	.968	0.12
		Algorithm design	1.00	.969	0.14
		Coding	1.00	.969	0.12
Test	1.00	.969	0.13		
Analysis and Evaluation	1.00	.970	0.13		

Table 9. Review opinion and reflection (2st)

Review comments	Reflect review
<ul style="list-style-type: none"> It seems necessary to develop the details of design thinking related to divergent thinking at level 1. 	<ul style="list-style-type: none"> In this study, we extracted the content elements for raising divergent thinking and convergent thinking. We will develop detailed contents of each element after this study with the elements of education contents derived from this study
<ul style="list-style-type: none"> It is necessary to develop detailed contents of computing thinking to convergent thinking. It would be good if we could develop the use of examples that fit the characteristics of each major. 	<ul style="list-style-type: none"> Through the subsequent research, we will reflect on the contents and education methods that can select and apply the programming language to the applicable level in the education field.

1차, 2차 전문가 조사 분석을 통해 도출한 단계별 DT 기반의 CT교육 내용을 두 개의 수준으로 나누어 Fig. 2 처럼 제시할 수 있다.



Fig. 2. D · C Thinking educational contents

5. 결론

5.1 연구요약

4차 산업혁명 시대는 단순 지식 습득에 초점이 맞춰진 교육에서 벗어나, ‘창의성’, ‘융합성’, ‘문제해결능력’과 같은 ‘역량’에 초점을 맞춰진 교육을 요구하고 있다[20]. 이를 위해 본 연구에서는 SW교육에 있어서 CT와 DT 사고를 갖춘 창의·융합 인재의 육성을 위한 역량 중심의 교육 내용을 제안하고자 하였다. 이를 위해 창의적 문제해결능력을 기르는데 많이 사용하는 DT와 CT를 융합하기 위한 수준을 나누고 각 단계별 학습해야할 내용들이 무엇인지에 대한 두 차례에 걸친 조사를 통해 전문가들의 의견을 종합적으로 분석하여 최종적인 내용 요소를 도출하고자 하였다. 비전공자 대상의 창의·컴퓨팅사고 SW교육의 방향과 내용요소를 다음과 같이 도출하였다.

첫째, 1수준에서는 확산적 사고에 초점을 두고 DT의 프로세스를 기반으로 전체 내용을 구성하되 프로토타입을 제작하는 프로세스부터는 문제해결에 필요한 CT 내용들로 구성하였다. 여기서의 CT내용들을 단순히 프로그래밍 언어를 배우는 내용이 아니라 아이디어를 실현하

는 과정을 보여주는 스토리텔링을 적용 가능한 사고중심의 알고리즘체계, 선택, 구현, 테스트, 평가로서 내용 요소를 추출하였다.

둘째, 2수준에서는 수렴적 사고에 초점을 두고 1수준에서 하였던 다양한 아이디어와 스토리를 컴퓨팅으로 구체화하여 실현하기 위한 단계로 보고 내용을 구성하였다. 이러한 이유에서 DT에서의 ‘아이디어 제안’부분부터 CT 영역의 내용 요소에 내포하여 다양한 아이디어 중 컴퓨팅으로 구현할 수 있는 아이디어를 추려내어 논리적으로 설계하고 구현할 수 있는 내용들로 구성하였다. 특히 인간의 사고를 컴퓨터적 관점으로 해석하여 모델링하고 이것을 알고리즘으로 절차화하여 설계하고 자동화로 구현하는 내용적 요소를 추출할 수 있었다.

5.2 연구의 한계 및 향후 연구 방향

이 연구는 단순히 지식을 습득하는 능력보다는 컴퓨팅사고 기반의 창의적 문제해결력을 요구하는 미래 역량으로서 창의성과 컴퓨팅이라는 발산적사고와 수렴적 사고들 간의 연결 고리를 찾아 창의적이면서 컴퓨팅사고적인 문제해결능력을 기르기 위한 비전공자대상의 SW교육의 내용을 제안하였다. 변화하는 대학 교육현장에서 SW기초교육 운영방향 정립에 기초자료로 활용될 것으로 기대한다. 또한 이 연구 결과 및 결론을 통해 도출한 내용구성을 바탕으로 후속연구로 SW교육내용을 체계화하여 구체적인 교육내용 개발할 필요가 있다. 더 나아가 SW교육에 있어서 비전공자들의 계열별 다양한 요구를 파악하고 그 결과를 반영한다면 보다 실질적인 SW교육을 할 수 있는데 도움이 될 것으로 보인다.

REFERENCES

- [1] Education Department. (2018). *2015 Revised National Curriculum*. Retrieved from <http://ncic.go.kr/mobile.dwn.ogf.inventoryList.do>
- [2] J. Sung & H. Kim. (2015). Analysis on the international comparison of computer education in schools. *The Journal of Korean association of computer education*, 18(1), 45-54.
- [3] *Examples of foreign software education and implications.(2018)*. Retrieved from http://edzine.chedi.re.kr/2015/summer/article/world_02.jsp
- [4] LitingLee, Sanxin, Huangzheng, & HuangmingWay. (2014). *Notes from Stanford University's Design Thinking Lecture*. Seoul : insert.
- [5] Design Thinking for Educators. (2018). Retrieved from <https://designthinkingforeducators.com>.
- [6] J. Y. Lee & J. M. Rhi. (2010). A Comparative Study on the Meaning of Design Thinking: In a view of Herbert Simon's design thinking and IDEO's design thinking. *Proceedings of Korean Society of Design Science*, 10, 62-63.
- [7] M. H. Kyung. (2010). *Using design process creative ideas development research on education: Ideas for the development of Work Sheet*. Master dissertation, Daegu University, Daegu.
- [8] S. S. Jeon. (2005). A Investigative Study on Design Process Based Design Instruction Models. *The Journal of Art Education*, 19, 355-385.
- [9] Design Thinking. (2018). Retrieved from <https://www.ideou.com/pages/design-thinking>
- [10] S. G. Isaksen & D. J. Treffinger. (1985). *Creative problem solving. The Basic Course*. New York: Bearly Limited.
- [11] S. M. Han, S. H. Kim , & W. L. Kyun. (2005). *Creativity*. Seoul: hakjisa.
- [12] Computational Thinking for all.(2018). Retrieved from <https://www.iste.org/explore/articleDetail?articleid=152>.
- [13] K. S. Oh. (2016). *A study on the contents of computational thinking for programming education*.Ph.D. dissertation, Sungkyunkwan University, Seoul.
- [14] K. S. Oh & S. J. Ahh. (2013). A study on the probelm-solving process model based on the computational thinking to improve creativity. *The journal of korean association of computer education*, 17(1), 183-186.
- [15] K. S. Oh & S. J. Ahh. (2016). A study on development of educational contents about computational thinking. *The journal of korean association of computer education*, 19(2), 11-20.
- [16] C. H. Lawshe. (1975). A quantitative approach to content validity. *Personnel Psychology*, 28(4), 563-575.
- [17] H. D. Song. (2007). Instructional Design Principles for Enhancing Creative Problem Solving Skills. *The journal of Korea Association of Yeolin Education*, 15(3), 55-73.
- [18] J. H. Chun & J. Y. Yoo. (2014). Trials and effects of a learner-centered creative training technique on undergraduate education of medical record information management. *Journal of Digital Convergence*, 12(3), 277-288.
- [19] J. Y. Seo, S. H. Shin,& E. H. Goo. (2017). Analysis of

Changes of Digital Mind by the Class Type in Basic Software Education for the Students of Humanities. *Journal of Digital Convergence*, 15(9), 55-64.

- [20] J. H. Kim. (2018). *KISTEP*. KISTEP InI. Retrieved from http://www.kistep.re.kr/c3/sub3_2.jsp?
- [21] Y. S. Lee. (2018). Python-based Software Education Model for Non-Computer Majors. *Journal of the Korea Convergence Society*, 9(3), 73-78
- [22] S. J. Kim & D. E. Cho. (2018). A Study on Learning Model for Effective Coding Education. *Journal of the Korea Convergence Society*, 9(2), 7-12.

오 경 선(Oh, Kyung Sun) [정회원]



- 2002년 8월 : 상명대학교 컴퓨터교육전공(교육학석사)
- 2016년 8월 : 성균관대학교 컴퓨터교육과(교육학박사)
- 2017년 9월 ~ 현재 : 단국대학교 SW중심대학사업단 조교수
- 관심분야 : 컴퓨터교육, 컴퓨팅사고, SW교육
- E-Mail : skyal@dankook.ac.kr

서 응 교(Suh, Eung Kyo) [정회원]



- 2003년 2월 : 연세대학교 산업정보경영학과(경영학 석사)
- 2008년 8월 : 연세대학교 경영학과(경영학박사)
- 2012년 9월 ~ 현재 : 단국대학교 교양교육대학 교수
- 관심분야 : 컴퓨터 매개 커뮤니케이션, 조직내 창의적 아이디어 발상
- E-Mail : eungkyosuh@dankook.ac.kr

정 혜 진(Chung, Hae Jin) [정회원]



- 2011년 2월 : 단국대학교 정보처리전공 (공학석사)
- 2017년 2월 : 단국대학교 정보처리전공 (공학박사)
- 2017년 9월 ~ 현재 : 단국대학교 SW중심대학사업단 조교수
- 관심분야 : 가상화, 빅데이터, 데이터분석, Docker
- E-Mail : haejini.chung@gmail.com