Journal of The Korean Association
of Information Education Vol. 21, No. 1,
February 2017, pp. 77-87

http://dx.doi.org/10.14352/jkaie.2017.21.1.77
© 2017 KAIE

정보과 교육과정 표준모델을 위한 알고리즘 및 프로그래밍 영역 프레임워크 개발

성영훈* · 박남제** · 정영식***

진주교육대학교 컴퓨터교육과*, 제주대학교**, 전주교육대학교 컴퓨터교육과***

요 약

학교현장에서 활용할 수 있는 SW교육과정 구성을 위해 알고리즘과 프로그래밍 영역에서 개념이해와 학습활동을 구성된 프레임워크를 개발하였다. 이를 기반으로 영역별 5단계 성취기준 체계로 구성되어 있으며 알고리즘 영역은 18개, 프로그래밍 영역은 21개의 성취기준 요소로 구성되어 있다. 각 영역별 성취기준에 대한 내용의 중요도와 교육시기의 적절성을 검토하기 위해 전문가 델파이조사 검증을 통한 타당도를 제시하였다. 연구의 결과는 정보과 교육과정 표준모델 구성을 위한 연구 자료로 활용될 수 있으며 학교 현장에 적용을 위한 SW교육과정 개발을 위한 시사점을 제공하고 있다는 점에 의의가 있다.

키워드 : SW교육과정, 알고리즘, 프로그래밍, 프레임워크, SW교육모델

Development of Algorithm and Programming Framework for Information Education Curriculum Standard Model

Younghoon Sung*, Namje Park**, Youngsik Jeong***

Chinju National University of education*, Jeju National University**, Jeonju National University of education***

ABSTRACT

We developed a framework of consisting of concept understanding and learning activities in the area of algorithm and programming for constructing SW curriculum that can be used in school field. Based on this, it is composed of five levels of achievement standards by area. The algorithm area consists of 18 achievement standards elements, and the programming area consists of 21 elements. To examine the importance of content about achievement standards and the appropriateness of the education time of each area, its validity was suggested through expert verification by delphi survey. The present study could be utilized as the research data of the standard model of information education curriculum and provides important implications for the development of SW curriculum in the school field.

Keywords : SW curriculum, Algorithm, Programming, Framework, SW education model

---

이 논문은 한국정보교육학회 2016년 정보과 교육과정 표준모델 개발 연구보고서의 내용을 보완·확장하였음
교신저자: 정영식(전주교육대학교 컴퓨터교육과)
논문투고 : 2016-12-00
논문심사 : 2017-00-00
심사완료 : 2017-00-00

1. 서론

최근 SW교육이 강조되면서 국내에서도 2015 개정교육과정 속에 컴퓨팅 사고력(Computational Thinking)을 향상 시킬 수 있는 알고리즘, 프로그래밍 등 SW교육 내용요소들을 강화하였다[8].

특히 알고리즘과 프로그래밍 교육은 실생활에서 발견되는 다양한 문제들을 해결하기 위해 컴퓨터사고절차를 기반으로 문제에 대한 본질 이해와 더불어 학습자들의 논리적 능력과 창의성을 향상 시키는 데 필요한 교육이다[3][5][7][13][15]. 그러나 제시된 내용요소의 범위에서 학습자들의 문제해결력을 향상과 일상의 문제해결과정을 다루기에는 17시간이라는 제한된 시간 내에 다루기에는 어려운 점이 많다[11].

한국정보교육학회(KAIE)에서는 2015년에 국내외 정보과 교육과정을 분석을 통해 소프트웨어 교육과정 모델을 개발하여 7단계의 모델로 된 소프트웨어, 컴퓨터시스템, 융합활동 3가지 영역 내 10개의 하위영역으로 구성된 내용요소로 제시하였다[14]. 제시된 표준모델에서는 초등학생들을 위한 정보과 교육내용요소를 중심으로 다루고 있어 개정교육과정의 초, 중, 고등학교의 정보교과 내용 범위를 포괄하기에는 보완해야 할 부분이 많이 있다.

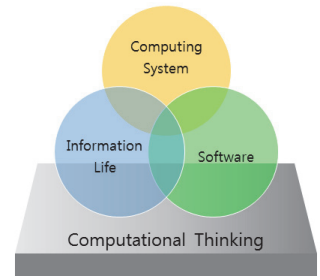
또한 교육과정의 구성 체계를 위해 2016년 미국 K12cs.org에서 제시하고 있는 컴퓨터과학교육에 대한 개념적 가이드라인을 살펴보면 기존 2011년 CSTA에서 제시하였던 5가지 영역(CT: Computational Thinking, CL: Collaboration, CPP: Computing Practice and Programming, CD: Computer and Communication Devices, CI: Community, Global, and Ethical Impacts)에서 크게 5가지 개념과 7가지 활동 중심으로 재구조화한 프레임워크를 바탕으로 4가지 학년 군별(K-2, 5, 8, 12)로 적용할 수 있는 성취수준으로 수정, 보완하였다. 또한 제시되고 있는 프레임워크를 활용하여 학교현장에서 이를 기반으로 다양한 교육과정을 설계하고 개발할 수 있도록 하였다[18].

본 연구에서는 KAIE에서 개발한 정보과 교육과정 표준모델의 소프트웨어 영역 중 알고리즘과 프로그래밍 영역을 중심으로 학년의 구분없이 학생들의 학습수준에 맞게 교사들이 학교현장에 활용할 수 있는 교육과정을 개발할 수 있도록 체계적인 프레임워크로 제시하고자 하였다. 이를 위해 전국의 교육대학교 컴퓨터교육과 교수들 16명을 대상으로 한 델파이조사 결과를 기반으로 재구성하여 영역별 핵심개념, 성취기준 및 성취기준에 대한 요약설명으로 제시하였다.

2. 관련연구

2.1 알고리즘과 프로그래밍 영역 내용체계 분석

KAIE에서 제시하고 있는 정보과 교육과정의 영역은 컴퓨팅시스템, 소프트웨어, 정보생활, 컴퓨팅 사고력 영역으로 크게 4개의 영역으로 구성되어 있다[10]. 이 중에서 (Fig 1.)과 같이 본 연구와 관련된 소프트웨어 영역은 알고리즘, 프로그래밍 및 로봇과 컴퓨팅 영역을 포함하고 있다.



(Fig. 1) KAIE Information Education Curriculum

알고리즘과 프로그래밍 영역 내용체계 구성을 위해 국내외 관련 교육과정을 분석해 보면 다음과 같다.

첫째, 내용 체계 면에서 국내의 2015 개정 교육과정과 소프트웨어 교육 운영지침에서 다루고 있는 알고리즘과 프로그래밍 영역의 내용 수준과 시기가 서로 달라 체계적인 교육과정 적용에 어려움이 있다[8][9][16].

둘째, 프로그래밍 교육내용에 대한 학습도입 시기 면에서 살펴보면 영국은 초등학교 3학년, 미국은 초등학교 1학년부터 시작하고 있는 반면 우리나라의 경우 초등학교 5~6학년부터 도입되고 있어 SW교육의 중요성에 비해서 상대적으로 너무 늦은 면이 있다[16].

셋째, 교육과정 내용 체계 구성을 위한 프레임워크면

에서 살펴보면 K12cs.org에서는 컴퓨터 과학에 대한 이해와 학습을 위해 개념(Concepts)과 활동(Practices)으로 구조화하여 학년 군으로 구분된 학습 단계별로 각각의 개념들이 성취수준과 연계되어 학습할 수 있도록 제시하고 있다[11][18].

넷째, 2015 개정교육과정 알고리즘과 프로그래밍 영역의 내용과 성취수준을 고려할 때 주당 1시간 이상해야 하는 미국, 영국, 독일 교육과정에 비해 우리나라는 17시간으로 너무 적은 시수로 운영해야하는 부담이 있다[1][12].

다섯째, 2015 KAIE의 SW교육 표준모델에서 제시하고 있는 내용체계는 학년 구분 없이 7단계로 구성되어 단계별 2개의 성취기준 형태로 제시하고 있다[14]. 그러나 내용체계에 대한 타당성 확보를 위한 전문가의 의견을 충분히 반영하지 못한 점이 있고 실제 학교현장에 활용할 수 있는 정보과 교육내용의 위계성을 고려할 필요성이 있다.

이에 KAIE 소프트웨어 교육과정 표준모델 분석연구 결과를 기반으로 알고리즘과 프로그래밍 영역을 구성하는 하위 영역들을 각각 다음과 같이 구성하여 제시하였다[16].

첫째, 알고리즘 영역에서는 크게 6개의 하위 영역으로 구성하였으며 세부적인 내용은 다음과 같다. 일의 순서와 표현 및 알고리즘과 프로그램과의 관계를 다루고 있는 알고리즘 개념 영역, 문제해결과정에 대한 표현과 패턴 및 순서도 표현과 알고리즘 분해를 다루고 있는 알고리즘 표현 영역, 다양한 알고리즘의 구조를 순서도로 표현해 보는 알고리즘 구조 영역, 알고리즘의 작동 결과를 예측하고 오류 수정 및 개선과정을 담고 있는 알고리즘 결과 영역, 탐색과 정렬 및 알고리즘을 협력해서 작성해 보는 알고리즘 실제 영역, 알고리즘의 장단점을 비교하고 성능에 대한 평가를 통해 최적 알고리즘을 선택하여 적용해 보는 알고리즘 평가로 구분하였다.

둘째, 프로그래밍 영역에서는 크게 7개의 하위 영역으로 구성하였으며 세부적인 내용은 다음과 같다. 프로그램의 의미를 이해하고 언어의 필요성과 종류에 대해 알아보는 프로그램의 개념 영역, 간단한 알고리즘과 연산자 및 텍스트 프로그래밍 언어로 작성하는 프로그램의 작성 영역, 다양한 형태의 조건과 반복 구조가 포함되는 제어구조 영역, 변수와 배열에 대한 이해와 프로그램 작성을 다루는 자료형 영역, 함수와 입출력 및 라이브러리를 활용하는 프로그램 작성하는 함수와 라이브러리 영역, 정렬과 탐색 프로그램 작성 및 오류수정을 통한 프로그램 성능 평가 과정을 포함하는 프로그램의 평가 영역, 실제 프로젝트 수행을 위한 문서작성과 협력해서 개발하는 방법 및 개발환경 구성과 디버깅에 대해서 다루고 있는 프로그램의 실제 영역으로 구성하였다.

2.2 문제해결과 컴퓨팅 사고력

2015년 개정 정보과 교육과정에서 강조하고 있는 역량 가운데 하나가 협력적 문제해결력이다[9]. 특히 정보과에서 문제해결력은 컴퓨터과학의 기본개념과 원리에 대한 이해를 바탕으로 실생활의 문제들을 해결하는 것으로 이는 컴퓨팅 사고력과 그 의미적 연계성이 있다. 또한 CAS, CSTA, K12cs.org 등에서 제시하고 있는 컴퓨팅 사고력과 관련된 주요 자료를 재정리하여 2016년 KAIE 정보과 교육과정 표준모델에서 크게 다음과 같이 5가지 요소로 요약하여 구분하였다[2][4][10][17][18].

첫째, 문제분석(Pr: Problem Analysis)으로 문제에 대한 이해와 정의, 그리고 주어진 문제에 대한 분해를 통해서 더 작은 테스크(Task) 단위로 나누는 과정을 의미한다.

둘째, 자료분석(Da: Data Analysis)은 문제해결을 위한 자료를 수집하고 다양한 형태의 데이터로 표현하며 분석된 결과에 대해 구조화하여 표현하는 과정을 의미한다.

셋째, 추상화(Ab: Abstraction)는 패턴분석, 논리적 추론, 모델링, 절차와 방법을 단순화하여 표현함으로써 문제해결과정을 알고리즘으로 나타내는 과정을 의미한다.

넷째, 자동화(Au: Automation)는 문제를 해결하기 위해 추상화하여 만들어진 알고리즘을 실제 프로그램으로 작성해보고 오류를 개선하는 과정 속에서 문제를 해결하는 방법을 확인한다.

다섯째, 일반화(Ge: Generalization)는 보다 나은 문제해결과정으로 개선하는 최적화, 문제해결을 위한 효과적인 목적 달성에 대한 평가, 유사한 문제에 적용 가능한 사례적용 요소로 구성된다.

이에 알고리즘과 프로그래밍 영역에서 컴퓨팅 사고력과 관련된 대표적인 구성 요소를 각각의 성취기준에 제시하여 학교현장에서 교육과정 개발 및 적용에 손쉽게

참고할 수 있도록 제시하였다.

2.3 시사점

KAIE에 제시하고 있는 정보과 교육과정 표준모델을 기반으로 소프트웨어 영역의 하위 영역 중 알고리즘과 프로그래밍과 관련된 프레임워크 개발을 위해 얻은 시사점은 다음과 같다.

첫째, 연구에서 제시하고 있는 각 영역별 프레임워크 내 개념, 성취기준을 기준으로 학교현장의 교육과정 뿐만 아니라 다양한 형태의 소프트웨어교육활동에서도 교사들이 학습자들의 수준에 맞게 재구성하여 활용할 수 있도록 개념들을 위계화하고 단계적으로 제시한다.

둘째, 내용체계는 각각의 하위영역별로 개념이해 부분은 핵심개념과 성취기준으로 구성하고 학습활동 부분은 교수학습방법과 평가방법으로 구성한다.

셋째, 문제해결을 위한 컴퓨팅 사고력 향상을 위해서 컴퓨팅사고력 영역의 하위요소를 개념이해 부분의 핵심개념 요소들과 연계하여 학습할 수 있도록 제시한다.

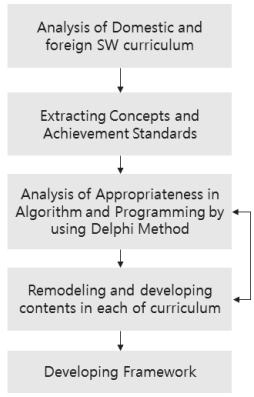
3. 개발 내용

3.1 연구절차

(Fig. 2)와 같이 KAIE의 소프트웨어 교육과정 표준모델을 기반으로 알고리즘과 프로그래밍 영역의 프레임워크 개발 연구 진행절차는 다음과 같다.

영역별 내용체계 구성을 위해 기존의 국내외 교육과정을 분석하여 알고리즘과 프로그래밍 교육에 대한 주요 개념들을 추출하였다.

2016년 전국 교육대학교 컴퓨터교육과 교수들을 대상으로 델파이 조사를 실시하여 알고리즘과 프로그래밍 영역에 대한 개념요소와 교육시기의 적절성을 분석하여 내용에 대한 타당성을 확보하였다[16]. 이를 기반으로 본 연구에서는 2차 검증에 참여한 전국교대 16명의 교수들을 대상으로 의견검토과정을 거쳐 각 영역별 성취기준에 대한 단계별 수정 및 재조정을 거쳐 최종적인 프레임워크를 개발하였다.



(Fig. 2) A Process of study

3.2 영역별 성취수준 검토

연구에서 제시하는 핵심개념과 성취기준 요소들에 대한 검토방법은 응답한 참여자에 대한 타당도 비율(CVR)과 합의도 계산방법을 적용하여 영역별 성취수준에 대한 내용의 중요도와 적용시기에 대해서 분석하였다[6][16].

$$CVR = \frac{N_e - \frac{N}{2}}{\frac{N}{2}}$$

N_e : 적절하다, 매우 적절하다고 응답한 수

N: 전체 응답자 수

$$\text{합의도} = 1 - \frac{Q3 - Q1}{Mdn}$$

Q_1, Q_3 : 제1사분위 값과 제3사분위 값

Mdn : 중앙값

성취수준별 내용의 중요도는 1점(전혀 적절하지 않다)~5점(매우 적절하다)까지의 5단계 리커트 척도로 평가하고 참여자가 16명이므로 CVR 값이 0.42 이상인 값을 타당한 것으로 분석하였다. 성취수준별 적용 시기는 초1~2, 초3~4, 초5~6학년, 중학교, 고등학교, 대학교의 6단계로 적용시기를 선택하도록 하였으며 성취수준에 대한 시기적인 적절성은 합의도 0.75 이상을 기준으로 하

였고 세부적인 내용은 다음과 같다.

3.2.1 알고리즘 영역

첫째, 교육내용의 중요도는 <Table 1>과 같이 영역별 내용의 중요성은 평균 4.50, 전체적인 평균 합의도는 0.78로 나타났다.

<Table 1> Validity Analysis of Algorithm Curriculum

Div.	M	CVR	Validity
Concepts of AL.	4.63	0.94	0.76
Representation of AL.	4.54	0.92	0.66
Constructure of AL.	4.58	0.85	0.81
Result of AL.	4.52	0.94	0.72
Practice of AL.	4.38	0.79	0.90
Evaluation of AL.	4.36	0.78	0.81
Mean	4.50	0.87	0.78

다만, 알고리즘 표현과 알고리즘 결과 부분에 대해서는 다소 낮은 합의도를 보인다.

둘째, <Table 2>와 같이 성취기준요소별 적용시기를 고려한 최종 단계(Lvl.) 표시는 교육과정에서 다루고 있는 초, 중, 고등학교 수준 5단계만 고려하였다. 전문가 의견수렴결과(단위: %)를 바탕으로 전체 단계균형을 고려하여 필요시 특정한 성취기준 요소의 경우 <Table 2>에서 음영 처리한 부분처럼 가장 많이 선택된 단계와 인접한 한 단계 전후로 재구성하여 제시하였다.

<Table 2> Achievement Standards of Algorithm

Div.	Achievement Standards	CT	Lvl.	Primary			M	H	U
				1,2	3,4	5,6			
Concepts of AL.	Understanding the sequence of work	PR	1	37.5	50.0	12.5	0.0	0.0	0.0
	Describing the sequence of work (picture, sentence)	PR	1	12.5	68.8	18.8	0.0	0.0	0.0
	Meaning of algorithm	PR	1	12.5	62.5	18.8	6.3	0.0	0.0
	Relationships of Algorithm and Program	DA	1	0.0	43.8	43.8	0.0	12.5	0.0
Representation of AL.	Expression of problem-solving process (Words and Writings)	DA	2	18.8	56.3	25.0	0.0	0.0	0.0
	Expression of problem-solving process (pictures, symbols)	DA	2	18.8	37.5	43.8	0.0	0.0	0.0
	Finding Patterns and Representing	AB	2	6.3	37.5	50.0	6.3	0.0	0.0
	Expression solutions according to conditions	AB	2	0.0	56.3	31.3	12.5	0.0	0.0
	Algorithmic representation *Flowchart, Pseudo code	AB	3	0.0	25.0	56.3	15.6	3.1	0.0
	Algorithm decomposition and correlation * Algorithm decomposition, algorithm relation	PR	3	0.0	12.5	37.5	46.9	3.1	0.0
Construcure of AL.	Represent various flowcharts * Sequence, iteration, condition, Mixing, nested iteration, Control structure	AB	3	1.0	12.5	44.8	39.6	2.1	0.0
Result of AL.	Problem description (current and target status)	PR	3	0.0	18.8	50.0	18.8	12.5	0.0
	Algorithm improvements * Prediction, Modification, and improvement	GE	3	0.0	4.2	25.0	45.8	22.9	2.1
Practice of AL.	Understanding the search algorithm	PR	4	0.0	12.5	12.5	56.3	6.3	12.5
	Understanding sorting algorithms	PR	4	0.0	12.5	6.3	68.8	6.3	6.3
	Collaborative writing for algorithms	GE	4	0.0	6.3	6.3	50.0	25.0	12.5
Evaluation of AL.	Algorithm analysis	DA	5	0.0	6.3	0.0	56.3	25.0	12.5
	Algorithm Evaluation * Execution time, complexity, performance	AB	5	0.0	3.1	3.1	25.0	46.9	21.9
	Application of Optimal Algorithm	GE	5	0.0	6.3	0.0	6.3	56.3	31.3

3.2.2 프로그래밍 영역

첫째, 교육내용의 중요도는 <Table 3>과 같이 영역별 내용의 중요성은 평균 4.47, 전체적인 평균 합의도는 0.77로 나타났다. 다만, 프로그램 개념, 제어구조, 프로그램 실제에서는 합의도가 다소 낮게 나타났다.

<Table 3> Validity Analysis of Programming Curriculum

Div.	M	CVR	Validity
Concepts of Prg.	4.53	0.88	0.59
Writing of Prg.	4.65	0.94	0.78
Control Structure of Prg.	4.59	0.96	0.73
Data Types	4.52	0.88	0.83
Functions and Library	4.39	0.85	0.89
Evaluation of Prg.	4.38	0.81	0.86
Practice of Prg.	4.25	0.80	0.70
Mean	4.47	0.87	0.77

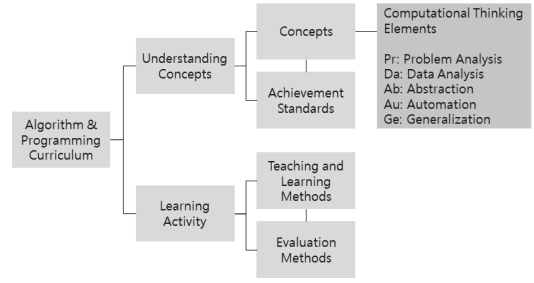
둘째, <Table 4>와 같이 성취기준요소별 적용시기를 고려한 최종 단계는 알고리즘 영역에서 설정한 기준과 같이 대학교에 표시된 부분은 제외하고 5단계로 설정하고 전문가 의견수렴결과를 기준으로 필요시 특정 성취기준요소는 가장 많이 선택한 단계를 기준으로 인접한 한 단계 전후로 재구성하여 음영으로 표시하여 제시하였다.

<Table 4> Achievement Standards of Programming

Div.	Achievement Standards	CT	Lvl.	Primary			M	H	U
				1,2	3,4	5,6			
Concepts of Prg.	Meaning of programming	PR	1	12.5	62.5	12.5	6.3	6.3	0.0
	Relationship of program and algorithm	DA	1	12.5	56.3	25.0	0.0	6.3	0.0
	Program operation	DA	1	37.5	56.3	6.3	0.0	0.0	0.0
	The need for a programming language	PR	2	0.0	62.5	31.3	6.3	0.0	0.0
	Types of programming language	DA	2	6.3	18.8	37.5	25.0	6.3	6.3
Writing of Prg.	Simple block programming	AU	2	25.0	68.8	6.3	0.0	0.0	0.0
	Simple algorithm implementation	AU	2	12.5	75.0	12.5	0.0	0.0	0.0
	Writing an arithmetic operation program	AU	2	0.0	62.5	37.5	0.0	0.0	0.0
	Writing Comparison and logic programs	AU	3	0.0	31.3	68.8	0.0	0.0	0.0
	Writing text programs	AU	3	0.0	6.3	56.3	31.3	6.3	0.0
Control Structure of Prg.	Writing control structure programs * Sequence, iteration, event, various iteration, condition, mixing, nested iteration	AU	3	0.9	29.5	50.0	17.9	1.8	0.0
Data Types	Writing variable-use programs * Using variables, scope of variables	AU	3	0.0	28.1	56.3	15.6	0.0	0.0
	Writing an array and list-based program	AU	4	0.0	0.0	28.1	59.4	12.5	0.0
Functions and Library	Create function-based programs * Simple functions, parameters, user-defined functions, standard I / O, file I / O, library calls	AU	4	0.0	4.2	16.7	63.5	13.5	2.1
Evaluation of Prg.	Creating sorting and searching programs	AU	4	0.0	0.0	15.6	62.5	15.6	6.3
	Program improvement	AU	3	0.0	6.3	43.8	37.5	6.3	6.3
	Evaluation of program performance	GE	5	0.0	6.3	12.5	25.0	43.8	12.5
Practice of Prg.	Understanding and writing program development plan documents	DA	4	0.0	12.5	43.8	18.8	18.8	6.3
	Development the program of collaboration	GE	4	6.3	0.0	43.8	25.0	18.8	6.3
	Understand debugging and use methods * Concept understanding, debugging method	AU	5	3.1	3.1	18.8	50.0	15.6	9.4
	Configuring the program development environment	DA	5	0.0	0.0	18.8	31.3	43.8	6.3

3.3 프레임워크 개발

알고리즘 및 프로그래밍 영역별 성취기준 검토과정을 거쳐서 재구성한 내용을 기반으로 KAIE 소프트웨어 교육과정 표준모델을 위한 최종 프레임워크를 (Fig. 3)과 같이 개발하였다.



(Fig. 3) Algorithm & Programming Curriculum Framework for KAIE SW Curriculum Standard Model

개발한 프레임워크는 크게 영역별로 개념이해와 학습활동 부분으로 구성되어 있으며 주요 사항은 다음과 같다.

첫째, 프레임워크를 구성하고 있는 개념이해 부분은 영역별 주요 내용요소인 개념들과 성취기준을 포함하고 있고 학습활동 부분은 교수학습방법과 평가방법에 대한 내용을 포함하고 있다.

둘째, 영역별 성취기준에 대한 체계적인 위계성 구성을 위해 전문가 조사결과를 기반으로 재구성한 5단계의 구조로 제시하였다.

셋째, KAIE 소프트웨어 교육과정 표준모델 적용을 위해 각 성취기준 요소별로 대표적인 컴퓨팅사고력과 관련된 하위 문제해결요소를 연관 지어 교육과정 개발과 교수학습에 참고할 수 있도록 하였다[10].

3.4 영역별 성취기준

개발한 프레임워크를 기반으로 알고리즘과 프로그래밍 영역에 대한 세부적인 성취기준은 다음과 같다[4][13].

3.4.1 알고리즘 영역

알고리즘의 세부 개념요소는 크게 알고리즘의 개념, 표현, 구조, 결과, 실제, 평가로 구성되며 성취기준들과 설명요소들을 요약정리하면 다음과 같다[4].

[1단계]

1. (일의 순서 이해, Pr) 일에는 순서가 있음을 이해, 생활에서 발생 가능한 일 알기, 일을 해결하기 위한 순서 설명

2. (일의 순서 표현, Pr) 일의 순서에 따른 그림, 문장 나열, 생활 속의 간단한 일 생각하여 순서대로 그림이나 짧은 문장으로 나열

3. (알고리즘의 의미, Pr) 문제해결을 위해 필요한 일의 순서나열이 알고리즘임을 알기, 문제해결을 위한 과정과 방법의 다양성 이해와 다양한 알고리즘 존재이해

4. (알고리즘과 프로그램 관계, Da) 알고리즘 의미 이해와 정보기기(컴퓨터 등)가 이해할 수 있는 언어로 된 명령어들로 자동화하여 검증하는 것이 프로그램임을 이해

[2단계]

5. (문제해결과정을 말과 글로 표현, Da) 문제해결과정의 말, 글로 표현, 일상 속 발생되는 문제해결과정의 말이나 글로 표현하고 각각의 장단점 이해

6. (문제해결과정을 그림과 기호로 표현, Da) 문제해결과정의 그림이나 기호 표현, 말이나 글로 표현하는 것과의 차이점과 장단점 이해

7. (패턴 찾기와 표현, Ab) 반복적인 일에서의 패턴찾기와 표현, 문제해결과정 속의 공통패턴을 찾아 말, 글, 그림 및 기호 등을 사용하여 표현

8. (조건에 따른 해결 방법 표현, Ab) 조건에 따라 달라지는 해결방법 표현과 문제해결방법 이해, 조건 만족여부를 포함한 다양한 형태의 방법으로 표현

[3단계]

9. (순서도와 의사코드, Ab) 알고리즘의 순서도와 의사코드 표현, 문제해결과정의 약속된 기호표현인 순서도 이해, 알고리즘 또는 용도에 맞는 순서도 기호 사용, 의사코드 의미 이해 및 의미 전달 가능한 간결하고 효과적인 알고리즘 표현

10. (알고리즘 분해와 관계, Pr) 복잡한 알고리즘 분해, 간단한 표현, 분해된 알고리즘 관계 파악, 알고리즘 속의 논리적 사고흐름을 다양한 방법으로 간결하고 효과적 표현, 분해된 알고리즘 간 논리적 흐름이해와 서로 간 관계 파악

11. (다양한 순서도 표현, Ab)

[순차구조] 다양한 알고리즘 구조의 순서도 표현, 생활 속 발생되는 일의 순서도 표현, 순차구조 표현시의 단점이해

[반복구조] 생활 속 반복적인 일에 대한 순서도 표현과 문제해결과정 목표 도달을 위한 절차의 시작과 끝 이해

[조건구조] 작성한 알고리즘 기반 다양한 형태의 조건 수행절차 이해와 순서도 표현, 조건 만족여부에 따라 문제해결방법 다름 이해

[반복, 조건구조] 반복과 조건 혼합된 일상의 일 찾고 이를 알고리즘과 순서도로 표현

[중첩반복구조] 생활 속 중첩된 반복 구조 일 찾고 순서도로 표현 및 간결한 표현 방법 이해

[제어구조] 일상 속 문제해결을 위한 순차, 반복, 조건의 복합된 복잡한 알고리즘의 순서도 표현

12. (문제의 현재와 목표 상태, Pr) 문제의 현재 상태와 목표 상태 설명, 문제분석 단계에서 현재 주어진 상황인 초기상태와 자신이 얻고자 하는 결과나 상태의미의 목표상태 설명

13. (알고리즘 개선, Ge) 알고리즘 작동 결과 예측, 오류수정을 통한 알고리즘 개선, 알고리즘 실행과정 이해, 주어진 조건에 따라 다른 결과의 알고리즘 찾기와 원인 설명 및 개선

[4단계]

13. (탐색 알고리즘 이해, Pr) 여러 가지 탐색 알고리즘(선형탐색, 이진탐색 등)이해와 차이점 표현

14. (정렬 알고리즘 이해, Pr) 여러 가지 정렬 알고리즘(버블정렬, 선택정렬, 삽입정렬, 병합정렬, 힙정렬 등)의 종류와 사용법이해와 장단점 파악

15. (협력적 알고리즘 작성, Ge) 친구들과 협력하여 탐색, 정렬 알고리즘 등 복잡하고 다양한 알고리즘 작성

[5단계]

16. (알고리즘 분석, Da) 문제해결 목표상태 도달을 위한 다양한 알고리즘 특징과 사용법 이해와 적용 및 장단점 비교와 알고리즘별 장단점 공존에 대한 이해

17. (알고리즘 평가, Ab) 알고리즘의 수행 시간이나 복잡도를 다양한 상황과 데이터 활용을 통한 평가, 데이터 양 증감에 따른 알고리즘 성능 차이변화에 대한 평가

18. (최적 알고리즘 적용, Ge) 일상 문제해결을 위해 구현하려는 알고리즘 성능 결과 예측을 통한 최적의 알고리즘 선택 및 유사 문제 적용

3.4.2 프로그래밍 영역

프로그래밍 영역의 세부 개념요소는 프로그램 개념, 작성, 제어구조, 자료형, 함수와 라이브러리, 평가 및 실제로 구성되며 구체적인 성취기준과 설명요소들을 요약하여 정리하면 다음과 같다[4][13].

[1단계]

1. (프로그램 의미, Pr) 컴퓨터가 처리해야할 작업에 대한 명령어들로 구성된 프로그램의 의미 이해

2. (프로그램과 알고리즘 관계, Da) 알고리즘은 수행절차의 추상화, 프로그램은 이를 자동화한 것임을 이해, 서로 간 관계 파악

3. (프로그램 작동, Da) 프로그램에 의해 작동되는 컴퓨터 이해와 명령어의 필요성 및 프로그램은 소프트웨어의 하위 관계 이해

[2단계]

4. (프로그래밍 언어 필요성, Pr) 프로그램을 작성하는 과정이 프로그래밍임을 이해 및 프로그램 언어의 필요성 이해

5. (프로그래밍 언어 종류, Da) 프로그래밍 언어의 종류와 장점이해, 인간이 이해하는 언어표현 방식의 프로그래밍 장단점 이해

6. (간단한 블록 프로그래밍, Au) 블록 프로그래밍 언어로 간단한 프로그램 작성, 스프라이트 이동 또는 회전하는 블록 프로그램 작성

7. (간단한 알고리즘 구현, Au) 생활 속 발견 문제에 대해 한 가지 목표달성을 위한 간단한 알고리즘이 담긴 프로그램 작성

8. (산술 연산자, Au) 산술 연산자 사용 프로그램(예:간단한 계산) 작성과 산술 연산자 우선순위 이해 및 다양한 계산식의 연산블록 조합 표현

[3단계]

9. (비교 연산자와 논리 연산자, Au) 비교 연산자, 논리연산자(and, or, not 등) 연산결과 이해 및 간단한 입력을 받아 이를 활용한 블록 프로그램 작성

10. (텍스트 프로그래밍 언어, Au) 텍스트 프로그래밍 종류 이해, 설치 및 간단한 언어와 문장으로 구성된 프로그램을 작성

11. (제어 구조, Au)

[순차구조] 순서대로 진행되는 순차구조 형식 프로그램 작성, 표현, 출력

[반복구조] 순차구조 형식 프로그램 기반으로 반복 수행되는 공통부분 추출을 통한 반복구조 프로그램 작성, 순차구조와 반복구조의 장단점 이해와 순차문의 반복문 변환

[이벤트] 이벤트 개념 이해, 블록 프로그래밍 언어 활용 사용자 입력이나 특정 이벤트 처리가능한 프로그램 작성

[다양한 반복구조] 다양한 형태의 반복구조 예를 들고 반복문 표현 및 프로그램 작성

[조건구조] 산술, 비교연산자 등이 포함된 조건 구조 프로그램 작성, 비교연산자 결과에 따른 조건 만족여부 판단과 조건 만족시 특정 구문 수행하는 간단한 조건구조 프로그램 작성

[혼합구조] 다양한 반복구조(횟수, 무한반복 등) 기반의 특정 조건 만족까지의 다양한 반복문과 조건문이 포함된 수행되는 혼합형태 프로그램 작성

[중첩반복구조] 다양한 연산자가 포함된 중첩된 반복구조 프로그램 작성, 반복 구조내 조건 만족 여부에 따른 이벤트 처리 가능한 조건문 사용

12. (변수 활용, Au) 변수의미, 변수 범위(전역변수 지역변수)이해, 변수활용 프로그램 작성, 이벤트(방송) 처리를 통한 결과의 차이점 이해, 목적에 맞는 변수 제작 및 사용

13. (프로그램 개선, Au) 프로그램 실행에 따른 오류 형태(컴파일, 런타임, 논리 등) 의미 이해, 오류 찾기와 수정을 통한 프로그램 개선

[4단계]

14. (배열과 리스트 활용, Au) 배열, 리스트 장점 이해를 통해 다양한 변수활용 구성과의 비교, 배열을 통한 데이터처리 장단점 이해, 리스트를 통한 데이터 추가 삭제시 배열과의 처리속도와 비교한 장점 이해 및 프로그램 작성

15. (함수 활용, Au) 함수와 라이브러리 활용 프로그램 작성

[간단한 함수사용] 함수 의미 이해, 간단한 내장함수(화면 입출력, 랜덤 함수 등) 호출 프로그램 작성, 이벤트내 객체 간 메시지 전달방법 이해와 사용

[매개변수와 인수] 메시지 전달을 위한 매개변수, 인수 사용 함수 제작과 이벤트처리 포함된 프로그램 작성 시 사용

[사용자 정의함수] 반복, 특정 기능수행을 위한 함수 정의, 반복과 조건문이 결합된 사용자 정의함수 제작과 프로그램 작성

[표준 입출력 함수] 변수인자, 사용자 정의함수 및 키보드 입력 데이터처리를 통한 화면출력 프로그램 작성

[파일 입출력 함수] 파일 입출력 함수 사용한 프로그램 작성
[라이브러리 함수] 라이브러리 함수 이해와 이를 활용한 다양한 형태의 입출력 프로그램 작성

16. (정렬과 탐색, Au) 문자열, 숫자, 파일 등 다양한 형태의 데이터를 특정 기준에 의해 순차, 이진탐색 알고리즘 적용하는 정렬프로그램 작성

17. (개발 계획서 작성, Da) 프로그램 개발을 위한 프로그램 개요, 내용, 기대효과 등 전반적 개발 계획서의 필요성 이해와 목표, 기능정의, 개발도구, 순서도, 테스트 방법 등 세부 설계계획 작성

18. (협력적 프로그램 개발, Ge) 생활 속 여러 가지 문제해결을 위해 개발 계획서 기반의 프로젝트 수행을 위한 개발절차 정의, 과제수행 점검, 목표 기능 구현방법, 테스트 시나리오와 체크리스트 작성, 결과발표와 공유를 포함한 과제전반에 대해 협력하여 프로그램 설계와 개발

[5단계]

19. (프로그램 개발 환경 구성, Da) 알고리즘 구현과 생활 속 문제해결을 위해 자신에 맞는 프로그래밍 언어 선택 및 사용을 위한 프로그램 개발 환경 구성

20. (디버깅 이해와 활용, Au) 다양한 프로그래밍 언어와 사용법 이해를 기반으로 디버깅 방법 이해와 프로그램 소스코드 실행절차 속 중지점에 대한 이해 및 사용법

21. (프로그램 성능 평가, Ge) 프로그램 실행에서 얻은 반응시간, 자원사용 효율성, 데이터 전송 효율성 등 프로그램 효율성, 문제해결성, 오류발생 등과 관련된 프로그램 신뢰성과 성능평가

4. 결론 및 제언

본 연구에서 개발한 알고리즘과 프로그래밍 영역의 내용체계와 성취수준은 전국 교육대학교 컴퓨터교육과 교수 16명의 델파이 조사결과에 따른 타당성을 확보하였으며 세부적인 연구결과는 다음과 같다.

첫째, 프레임워크 개발을 위한 성취기준과 내용체계에 대한 델파이 조사 결과 대부분 중요성과 합의도에서 타당하다고 나타났다. 다만 알고리즘 표현, 알고리즘 평가, 프로그래밍 개념, 제어구조, 프로그램 실제 핵심개념과 관련된 성취기준 요소들은 일부 적용시기에 대해 다소 낮은 합의도를 보였다.

둘째, 프레임워크는 영역별로 전문가의 검토결과를 반영하고 최종적으로 학년구분 없이 적용할 수 있는 5단계의 성취기준 체계로 구성하였고 세부 성취기준 요소들은 대표적인 컴퓨팅사고의 하위요소와 연계하였다. 세부적으로 개념이해 부분에서 알고리즘 영역은 6개의 핵심개념과 18개의 성취기준요소 구성되어 있으며 프로그래밍 영역은 7개의 핵심개념과 21개의 성취기준요소로 구성되었다.

본 연구에서 개발한 KAIE 소프트웨어 표준모델을 기반의 알고리즘과 프로그래밍 영역에 대한 프레임워크를 통해 학교현장 뿐만 아니라 다양한 소프트웨어 교육 활동에서 학년 구분 없이 학습자의 수준에 맞는 SW교육과정을 재구성하여 활용할 수 있을 것이다. 향후 개발한 프레임워크의 일반화를 위해 제시하고 있는 성취기준에 따른 교수학습방법과 평가방법을 개발하여 실제 현장에서 활용 할 수 있는 교수학습모델에 대한 연구가 필요하다.

참고문헌

[1] Kapsu Kim (2016). An Implications of Computer Education in Korea from the U.S., U.K. and Germany Computer Curriculums. *Journal of The Korean association of information education, 20*(4), 421-432. The Korean Association of Information Education.

[2] CAS (2013). Computing in the national curriculum: A guide for primary teachers. Computing at school.

[3] Chul Kim (2015). A Study on Algorithm Teaching and learning Methods and Assessment for Elementary School Students, *Journal of The Korean Association of Information Education, 19*(4), 489-498. The Korean Association of Information Education.

[4] Chul Kim, Namje Park, Younghoon Sung, Soobum Shin, Youngsik Jeong (2016). Development of Information Education Curriculum Standard Model. Research Report 2016. The Korean Association of Information Education.

[5] Hyungshin Choi (2014). Development of a Holistic Measure of Learning Effects in Robotics Program: Connecting Sociocultural Context and

Computational Thinking, *Journal of The Korean Association of Information Education, 18*(4), 541-548. The Korean Association of Information Education.

[6] Injoong Ju, Dongyeol Park, Misug Jin(2010). The Study of Core Competency's Domains and Levels. Korea Research Institute for Vocational Education & Training.

[7] Lye, S. Z., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12?, *Computers in Human Behavior, 41*, 51-61.

[8] Ministry of Education (2015). 2015 Revised Curriculum. Korea Ministry of Education.

[9] Ministry of Education (2015). Software Education Guidelines. Korea Ministry of Education.

[10] Soo-Bum Shin, Chul Kim, Namje Park, Kap-Su Kim, Young-Hoon Sung, Young-Sik Jeong (2016). Convergence Organization Strategies of the Computational Thinking in Informatics Curriculums, *Journal of The Korean Association of Information Education, 20*(6), 607-616. The Korean Association of Information Education.

[11] Younghoon Sung, Youngsik Jeong, Namje Park (2016). Development of Network and Digital creation tools Curriculum Model to consider understanding concepts and learning activity, *Journal of The Korean Association of Information Education, 20*(6), 563-574. The Korean Association of Information Education.

[12] Youngsik Jeong (2014). A Study on the Content Framework of Algorithm Education in Primary and Middle Schools. *Journal of The Korean association of information education, 18*(2), 275-284. The Korean Association of Information Education.

[13] Youngsik Jeong (2015). A Development of Algorithm and Programing Curriculum Model for Elementary School Students, *Journal of The Korean Association of Information Education, 19*(4), 459-466. The Korean Association of Information Education.

[14] Youngsik Jeong, Kapsu Kim, Inkee Jeong, Hyunbae Kim, Chul Kim, Jeongsu Yu, Chongwoo Kim, Myunghui Hong (2015). A Development of the Software Education Curriculum Model for Elementary Students, *Journal of The Korean Association of Information Education, 19*(4), 467-480. The Korean Association of Information Education.

[15] Youngsik Jeong, Jeongsu Yu, Jinsuk Lim, Yukyoeng Son (2015). Thesis of Software Education. CMASS publication.

[16] Young-Sik Jeong, Soo-Bum Shin, Young-Hoon, Sung (2016). Analysis of Appropriateness in Information Curriculum for Algorithm and Programming Education, *Journal of The Korean Association of Information Education, 20*(6), 575-584. The Korean Association of Information Education.

[17] CSTA (2011). The 2011 CSTA K-12 Computer Science Standards. Retrieved from http://www.csteachers.org/page/CSTA_Standards.

[18] K12cs.org (2016). K - 12 Computer Science Framework. Retrieved form http://K12cs.org.

저자소개

성 영 훈

2000. 진주교육대학교(학사)

2002. 진주교육대학교 교육대학원 컴퓨터교육 전공(석사)

2010. 경상대학교 대학원 컴퓨터과학(공학박사)

2011～2015. 한국교육학술정보원 연구원

2015～현재 진주교육대학교 컴퓨터교육과 조교수

관심분야 : SW교육, 컴퓨팅융합교육, 국가행정정보시스템

e-mail : yhsung@cue.ac.kr

박 남 제

2008 성균관대학교 컴퓨터공학과(공학박사)

2003～2008 한국전자통신연구원 정보보호연구단 선임연구원

2009 University of California at LA(UCLA) Post-doc.

2010 Arizona State University (ASU) Research Scientist

2010～현재 제주대학교 교육대학 초등컴퓨터교육전공 교수

관심분야 : 컴퓨터교육, STEAM, 정보보호, 암호이론 등

E-Mail: namjepark@jejunu.ac.kr

정 영 식

1996 춘천교육대학교 수학교육학과(교육학학사)

2001 한국교원대학교 컴퓨터교육과(교육학석사)

2004 한국교원대학교 컴퓨터교육과(교육학박사)

2004～2011 한국교육개발원 연구위원

2004～현재 전주교육대학교 컴퓨터교육과 교수

관심분야: 컴퓨터교육, 프로그래밍, 이러닝

E-Mail: nurunso@jnue.kr