

Computational Thinking 교육에서 나타난 컴퓨터 비전공 학습자들의 어려움 분석

김수환[†]

요 약

본 연구의 목적은 Computational Thinking(CT)을 모든 대학생들에게 가르치기 위한 기초연구로 컴퓨터 비전공 초보학습자들이 CT 교육 초기에 겪는 어려움을 조사, 분석하여 시사점을 제시하는 것이다. 최근 국내외적으로 CT를 기반으로 한 융합형 인재 양성의 중요성이 대두되면서 국내 대학에서도 CT 관련 강좌를 필수교양으로 채택하고 있다. 본 연구에서는 CT 교육의 도구로 스크래치를 활용하여 서울 소재 C대학에서 2014년 1, 2학기 동안 교양강좌를 진행하고 각 수업내용에서 비전공 학습자들이 어떤 부분을 어려워하는지에 대해 조사, 분석하였다. 연구결과 CT 학습과정에서 비전공 초보학습자들은 변수, 리스트의 개념을 가장 어려워하며, 아이디어를 생각하고 구현하는 과정, 어떤 명령어를 선택해야 하는지에 대한 고민 순으로 어려워하는 것으로 나타났다. 학습에 대한 재미와 흥미는 self-프로그래밍 능력과 self-CT 역량에 유의미한 상관관계가 있는 것으로 나타나 학습자의 어려움을 감소시키는 요소로 활용할 수 있다. 또한, 스크래치와 같이 쉽고 직관적인 교육용프로그래밍 언어의 경우에도 명령어의 사용과 응용에 대한 적절한 학습 시간 제공을 고려하는 것이 필요하다는 시사점을 도출하였다.

주제어 : SW 교육, 컴퓨팅 사고력, 스크래치, 초보 학습자, 어려움

Analysis of Non-Computer Majors' Difficulties in Computational Thinking Education

SooHwan Kim[†]

ABSTRACT

The purpose of this study is to provide considerations through investigation and analysis about non-computer major learners' difficulties in computational thinking education. In recent, the importance of human resources development in convergence based on computational thinking is increasing internationally and a Korean university is selecting CT as a mandatory subject. I taught CT with Scratch at C university in Seoul for two semesters in 2014 and investigated and analyzed what difficulties non-Computer majors felt in the process of CT education. The result showed they felt the following some difficulties in order: the concept of variable and list; to think a idea and implement it; which commands should be selected. The pleasure and the interest can be apply to decrease difficulty, because they affect self-programming ability and self-CT capability each other statistically. Although Scratch is an easy and an intuitive programming language, it is needed to consider to provide appropriate learning time to student for using and applying commands

Keywords : SW Education, Computational Thinking, Scratch, Novice, Difficulty

[†] 중신회원: 충신대학교 교양교직과 교수
논문접수: 2015년 2월 22일, 심사완료: 2015년 4월 27일, 게재확정: 2015년 5월 5일

1. 서론

미래사회는 디지털 문제해결력을 기반으로 한 융합 사고력을 갖춘 인재를 요구하고 있다[24]. 국내외적으로는 디지털 문제해결력의 기반이 되는 사고력으로 Computational Thinking(이하 CT)을 강조하고 있으며, CT를 모든 학생들에게 가르치려는 노력을 시도하고 있다[15][19][21][22].

영국의 경우 2000년부터 컴퓨팅 교과를 만들어서 초중등학교에 적용하고 있으며, 미국에서도 시카고를 비롯한 여러주에서 CT 교육을 초중등학교에서 실시하고 있다[15][22][26]. 영국 컴퓨팅교과의 목표는 CT를 기르는 것이며, 학생들이 CT를 기반으로 한 문제해결력을 길러주는 것을 목표로 하고 있다[14][22].

우리나라에서도 2014년 정부에서 SW교육의 초중등학교 적용을 공표하고, 2014년부터 시범학교를 운영하고 있다[16]. 초중등 학생들이나 비전공자를 위한 SW교육의 목표는 전공자를 대상으로 한 고등교육에서의 목표와는 다르게 설정되어야 한다. 전공자를 대상으로 한 고등교육에서의 SW교육 목표는 SW를 개발하는 개발자나 설계자를 목표로 이루어지고 있으며, SW관련 산업분야로 진출하고자 하는 인력을 대상으로 한다. 이와는 달리 초중등 학생이나 비전공자를 대상으로 한 SW교육의 목표는 미래 인재로서의 역량을 길러주는 것이 목표이므로, 학습자들의 사고력과 창의성, 문제해결력을 길러주는 방향으로 설정되어야 한다[2][9][16]. 따라서 초중등 학생이나 비전공자들을 대상으로 한 SW교육에서는 단순한 코딩이나 프로그래밍 언어를 통한 SW개발에 초점을 맞추기 보다는 SW를 개발하는 과정을 통하여 학습자들의 사고력을 향상하는 부분에 초점을 맞추어야 한다[2][6].

미국의 카네기 멜론 대학[27]이나 하버드 대학[28]에서도 컴퓨터 비전공자들을 대상으로 한 CT를 활발하게 진행하고 있으며, 우리나라에서도 K대학교에서 전공필수 과목으로 지정하여 모든 학생들에게 교육을 진행하고 있다[25]. 컴퓨터 비전공자들을 대상으로 교육하려면 비전공학생들이 CT 교육을 어떻게 받아들이는지, 어떤 부분을 어려워 하는지에 대한 부분에 대한 연구가 필요하

다. 따라서 본 연구에서는 이전 연구에 나타난 초보 프로그래밍 학습자들의 어려움을 토대로 [13][23], CT 교육에서의 비전공 학습자들이 보이는 인식 및 어려움에 대한 분석을 연구하였다. CT 교육 과정에서 나타난 학습자들의 인식과 행위에 대한 분석이므로 양적 및 질적 연구 방법론을 혼합하여 혼합 연구 방법론을 적용하여 분석하였다. 학습자들의 재미, 흥미, CT에 대한 인식은 양적 분석을 실시하고, 학습자들의 CT에 대한 긍정·부정 인식, 어려움에 대한 특성은 질적연구를 병행하여 실시하였다.

2. 이론적 배경

2.1 SW 교육과 CT

SW 교육은 일반 산업계에서 SW 개발자를 양성하기 위해 이루어지는 교육을 의미하나, 최근 SW가 미래산업의 중요한 분야로 대두되면서 모든 학생들에게 가르쳐야 하는 교육으로 부각되고 있다[27]. 미국과 영국을 비롯한 선진국에서도 SW교육을 강화하고 있으며, 모든 학생들을 위한 SW교육의 목표를 CT로 설정하고 있다 [19][21][22]. 우리나라에서도 초중등 SW교육의 목표로 설정되었다[16].

영국의 컴퓨팅 과목의 교육목표는 ‘컴퓨팅적 표현방식으로 문제를 분석하고 이러한 문제를 해결하기 위해 컴퓨터 프로그램을 기술하는 실제적 경험을 반복해 본다’라고 설정하고 있다[14][22]. 미국의 CSTA에서 발표한 컴퓨터 과학 교육과정에서는 단계별로 CT 역량을 길러주는 목표에 초점을 맞추고 있다[19]. 또한 AP 코스에서 발표한 교과 내용에서도 Computational Thinking Practices를 소개하고 있다[18].

국내외에서 이루어지고 있는 CT 교육은 4가지 분야로 나타나는데[2], 교육용프로그래밍 언어를 사용한 교육, 컴퓨터 없이 컴퓨터과학의 원리를 가르치는 언플러그드 교육, 센서를 이용하여 물리세계와 프로그램을 연결하는 피지컬 컴퓨팅, 다른 학문에서 CT를 활용하는 융합 컴퓨팅으로 구분할 수 있다.

교육용프로그래밍 언어를 통한 CT 학습에서는

자신의 아이디어를 표현하고 프로그램으로 구현하는 프로그래밍 과정을 통해 학습이 이루어진다. 따라서 본 연구에서는 스크래치를 활용한 CT 교육에서의 학습자 인식에 대한 연구를 진행하였다.

2.2 CT 교육에서 초보학습자의 어려움

모든 학생들에게 CT 교육을 실시하려면 학생들이 무엇을 어려워 하는지, 어떤 부분에서 거부감을 가지고 있는지에 대한 연구가 선행되어야 한다. 앞 절에서 밝힌 것처럼 교육용 프로그래밍 언어를 활용한 교육에서는 프로그래밍 과정이 필수적으로 이루어지는데, 프로그래밍 과정에서 나타난 초보 학습자의 어려움에 관한 연구를 분석하면 다음과 같다.

장혜선 외(2007)은 두리틀을 활용한 교육에서 나타난 오류를 구문에러, 의미에러, 논리에러로 구분하였다[10]. 구문에러는 일반적으로 문법에러로 볼 수 있으며, 의미 에러는 작성한 코드의 해석과 관련된 에러이다. 구문 자체만 보면 에러가 없지만 변수의 타입, 선언하지 않은 변수 사용, 잘못된 매개변수로 메소드 호출 등의 경우에 발생하는 에러로 정의하였다.

<표 1> 초보학습자들의 오류 유형

분류	내용
구문 기반 문제	자연언어 문제
	인간-중심 해석
	불일치
계획 조합 문제	요약
	최적화
	선 경험
	특수화
	경계구분
	예상을 벗어난 경우
	인식 부족 문제

정란(1996)은 <표 1>과 같이 초보학습자들이 프로그래밍 학습에서 보이는 오류 유형을 3가지로 분류하였다[11]. 첫째, 언어가 제공하는 기능을 제대로 이해하지 못하여 생기는 구문-기반 버그, 둘째, 프로그램 내의 정보 흐름을 파악하지 못함으로써 계획을 적절하게 통합시키지 못해 생기는 구문-기반이 아닌 버그, 셋째, 구문-기반인 듯하면서도 반드시 구문-기반이 아닌 버그이다.

Jenkins(2002)는 프로그래밍의 어려움은 다음과 같이 7가지로 분류하였다[23].

- 다양한 기술 : 선택스 기본, 점차 시맨틱, 구조, 스타일로 옮겨감
- 다양한 과정 : 알고리즘을 세분화하는 과정, 그것을 코드로 옮기는 과정
- 언어 : 프로그래밍 언어 문제
- 교육적인 새로움 : 새로운 과목에 대한 부담감
- 관심 : 학생들은 지루해 함. 최선의 프로그래밍은 재미있고, 창의적인 활동으로 구성해야 함
- 평판과 이미지 : 어렵다는 이미지
- 학습시간 : 대학에서 배우기 때문에 시간 스케줄의 문제

최정원과 이영준(2014)은 <표 2>와 같이 CT의 요소로 볼 수 있는 순차, 반복, 조건, 이벤트를 학습하는 과정에서 나타난 초보 학습자들의 오류를 분석하였다[13].

<표 2> 요소별 초보학습자들의 오류 분석

오류	세부 설명
순차	학습자들은 블록의 순서를 제대로 나열하지 못하거나 필요한 블록을 생략 혹은 불필요한 블록을 삽입하는 등의 문제 해결에서의 어려움을 보임
반복	학습자들은 반복되어야 할 블록 순차적 나열, 반복문 내에 불필요한 블록을 삽입, 반복 횟수 입력 오류, 문제 해결에 적합하지 않은 반복 블록 선택, 중첩 구조를 만들지 못하는 등의 어려움을 느낌
조건	학습자들은 조건 개념 내에 반복 기능이 포함되어 있다고 인식하거나 조건 블록 내에 삽입되어야 할 블록을 찾아내지 못하는 어려움을 겪음. 이는 논리 구조와 사용 방법에 대한 인지 부족
이벤트	이벤트 학습자들은 함수 대신에 함수에 들어가야 할 블록들을 반복해서 절차적으로 나열하거나 반복 블록을 사용하였으며 함수의 호출 위치 파악이나 매개 변수 삽입 위치 파악, 함수에 포함되는 블록을 선정하는 데에 어려움을 느끼고 있음. 이는 함수에 대한 개념과 사용 방법에 대한 이해가 충분히 이루어지지 않은 점, 문제 해결 과정의 전체 맥락에서 함수에 포함되어야 하는 패턴을 정확하게 인지하지 못한 데 그 원인이 있음

여러 학자들의 연구결과에 따라 CT 교육에서 공통적으로 나타나는 초보학습자의 어려움은 새로운 프로그래밍언어의 어려움에 대한 문제, 컴퓨터과학의 개념에 대한 어려움, 프로그래밍 과정에 대한 지루함, 프로그래밍이 어렵다는 인식 등이

다. 따라서 본 연구에서는 기존의 연구에서 학습자가 인식하는 어려움에 대한 분석과 CT 개념 요소에서의 어려움을 분석하고, 재미, 흥미와의 관계를 분석하여 시사점을 제시하였다.

2.3 내용분석법

내용분석법은 커뮤니케이션학에서 비롯된 사회과학연구방법론의 하나로 역사적, 비교론적 연구가 가능할 뿐만 아니라 절차적 오류나 연구방법의 모순 등이 존재하면 언제나 재검토가 가능할 정도로 안전한 방법론이기 때문에 사회과학 분야에서 유용성이 높은 것으로 알려져 있다. 내용분석법은 질적연구의 한 방법으로 비구조화된 커뮤니케이션의 자료를 면밀한 해석을 통하여 어떠한 결론을 추론하는 과정으로서 커뮤니케이션 자료의 특성을 체계적, 객관적으로 규명하는 방법으로 정의된다. 그 예를 들면 공문서, 교과서, 교과지도서, 학교의 규칙과 규율, 생활기록부, 성적표, 시험점수, 학생들에 의하여 만들어진 모든 산물들, 포트폴리오, 수업시간표, 교육과정 지침서, 교실일지, 학급회의록, 연구자 반성일지, 상담일지 등과 같은 것들이다[5]. Dictionary of Statistics & Methodology(2005)의 정의에 따르면 내용분석법은 책이나 뉴스기사, 텔레비전 프로그램, 인터뷰 내용과 같이 문서나 음성으로 기록된 커뮤니케이션 자료나 그림으로 표현된 커뮤니케이션의 내용을 체계적으로 분석하는 연구 기법이다. 본래 의사소통의 과정에서 발생하는 중요한 문제인 누가, 무엇을, 어떻게 전달하며 그 효과는 무엇인가라는 문제를 확인하며 분석하기 위해서 개발된 방법이다. 그러나 사회과학 일반에서 이 방법의 중요성이 인식되면서 교육이나 심리학에서도 개인의 지적 및 정의적 특성을 연구하기 위한 방법으로 이용하게 되었다[3].

내용분석의 절차는 다음과 같다[1][20].

첫째, 내용을 선정하고 표집하여야 한다. 본 연구에서는 CT 수업과정에서 학습자들이 작성한 성찰 저널을 표집하였다.

둘째, 내용분석 자료를 부호화하여야 한다. 내용분석에서 대상자료를 부호화하는 것은 연구대상의 내용특성을 정확하게 기술할 수 있도록 원

자료를 체계적으로 변환하거나 결합하는 과정이다. 자료의 부호화라는 원칙은 내용분석 연구의 필수적인 요소로서 연구자는 내용분석의 범주와 내용분석의 단위, 그리고 계수 체제를 결정함으로써 자료를 부호화 하는 원칙을 설정해야 한다.

셋째, 자료를 분석한다. 부호화한 자료를 다각도에서 분석한다.

본 연구는 ‘학습자 성찰 저널’을 분석하는 것이므로 주제범주의 방법을 사용하여 분석하였다. 주로 문장 분석이나 내용분석에도 사용되고 있는 범주로[1][8], 본 연구에서는 성찰 저널 문장 분석이 목적이므로 내용분석법을 적용하여 문장 진술을 분석하였다.

3. 연구의 방법 및 절차

3.1 연구과정 및 교육내용

본 연구에서는 CT 교육 과정에서 비전공 초보 학습자들이 겪는 어려움 및 인식을 조사, 분석하는 것이 목적이므로 연구의 절차는 양적 분석과 질적 분석단계로 이루어졌다. 먼저 <표 3>과 같이 2014년도 1, 2학기에 수도권 소재 C대학교에서 스크래치를 통한 CT 교육을 실시하고 각 수업내용별로 학습자의 인식 및 어려움을 조사하였다. 각 주차별로 강좌시간은 3시간씩 진행하였다.

<표 3> 연구의 과정

단계	활동 내용	참가 학생	비고 및 수집 데이터
준비	컴퓨터 비전공 인문사회 계열 학생 모집		
실험 1	스크래치 활용 CT 교육 -CT 개념 -CT 실습 -자율 프로젝트 제작	72명	2014년 1학기/ 재미, 흥미, self-프로그래밍 능력, self-CT 능력/ 성찰 일지
실험 2	스크래치 활용 CT 교육 -CT 개념 -CT 실습 -자율 프로젝트 제작	75명	2014년 2학기/ 재미, 흥미, self-프로그래밍 능력, self-CT 능력/ 성찰 일지

연구에 참여한 학생들은 모두 컴퓨터 비전공자로 1-4학년 학생들 147명이었으며, 데이터가 제대로 수집된 112명의 데이터를 대상으로 양적 분석을 실시하였다. 양적 분석을 위해서는 매 시간 수업마다 학습내용에 대한 재미, 흥미, 난이도를

Likert 5점 척도로 조사하였다. 수집한 데이터를 통해 빈도분석으로 각 요소의 추이를 분석하였다. 또한 교육 전후를 통해 재미, 난이도, self-프로그래밍 능력, self-CT 역량을 측정하여 t-검정 및 상관분석을 실시하였다. self-프로그래밍 능력은 스스로 프로그래밍을 잘한다고 인식하는 문항으로 수집하였으며, self-CT는 Brennen(2015)이 제시한 CT 관점을 문항으로 제작하여 설문하였다 [17].

질적 분석은 매 수업 시간마다 학습자 성찰 저널을 작성하도록 하여 수업 내용을 통해 좋았던 점이나 어려웠던 점에 대한 내용을 기록하도록 하였다. 내용 분석법을 적용하여 각 성찰 저널을 통해 나타난 수업 내용에 대한 긍정, 부정을 조사하여 CT 교육에 대한 이미지를 분석하였다. 비전공 학습자들의 CT 교육에 대한 어려움의 분석은 어려움에 대한 요소 분석을 통하여 실제 어떤 요소를 어려워하고 있는지에 대한 분석을 실시하였다.

스크래치는 초보자들이 자신의 아이디어를 다양한 멀티미디어 프로그램으로 쉽게 작성할 수 있는 교육용프로그래밍 언어로써, CT 교육에서 많이 사용되고 있다[17].

<표 4> 교수-학습 활동 주제

강좌	활동 주제	CT 요소
1강	스크래치 소개 애니메이션	인터페이스 설명 절차, 반복/ 표현하기
2강	조건에 따른 애니메이션 상호작용 애니메이션	절차, 반복, 분기/ 재사용 & 재구성/ 표현하기
3강	미로게임 바나나 게임	변수, 난수, 객체 통신/ 테스팅 & 디버깅/ 표현하기, 연결하기
4강	음악 프로그램 리스트 활용	데이터 입출력, 불린 로직, 연산자/ 추상화 & 모듈화/ 연결하기, 질문하기
5강	수치계산 프로그램 퀴즈 프로그램	불린 로직, 연산자, 객체 통신/ 추상화 & 모듈화/ 연결하기, 질문하기
6~7강	개인 프로젝트 제작하기	모든 요소

본 연구의 주별 교육내용은 <표 4>와 같다. 마지막 과제는 기존의 프로그래밍 과제들이 아닌 개인별로 자신이 구상한 애니메이션, 게임, 시뮬레이션 등을 개발하게 하였다. Jenkins(2002)는 프

로그래밍의 어려움을 극복하기 위한 방안으로 재미있고 창의적인 활동을 통해 학습하기를 제안하였다[23]. 따라서 본 연구에서는 학습자의 재미, 흥미, 어려움에 대한 인식을 양적 데이터로 수집하였다.

3.2 내용분석 범주화

본 연구에서 사용하는 내용 분석법의 범주는 주제범주이므로, 기존의 프로그래밍 과정에서의 어려움에 대한 범주와 초보학습자의 코딩조직[7]을 사용하기로 한다.

앞 절에서 분석한 바와 같이 여러 학자들의 연구를 바탕으로 본 연구에서는 CT 교육에서의 학습자 어려움을 분석하기 위해 주제 범주를 <표 5>와 같이 설정하였다. 본 연구에 참여한 147명이 매 시간 작성한 687개의 성찰 저널을 내용분석법으로 분석하였다.

각 범주별로 분석하기 위한 세부요소를 설정하기 위해 성찰저널을 1차 분석한 결과 <표 5>와 같은 세부요소들이 나타났다. 따라서 각 세부요소가 포함된 문장을 범주로 분류, 분석하였다.

<표 5> 내용분석법에 따른 범주 구성

유형	범주	세부요소
수업에 대한 이미지	긍정	감사, 힘들지 않음, 재미, 흥미
	부정	모르겠음, 힘들다, 어렵다
수업에 대한 어려움	명령어	명령어 사용, 명령어 선택
	객체사용	음악/그림
	CT개념	연산, 방송, 변수/리스트
	CT실습	알고리즘, 코드응용, 오류수정
	사고력	생각, 창의성

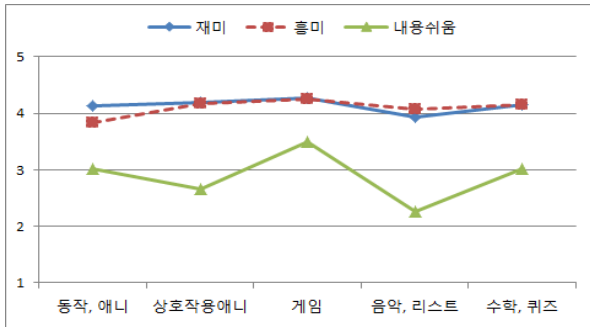
4. 연구결과

4.1 학습자 인식 변화

Jenkins(2002)의 제안[23]에 따라 CT 교육 과정에서의 학습자의 어려움을 극복하는 요소로 재미, 흥미, 난이도를 설정하고 각 주별로 <표 4>의 수업내용에서의 요소를 측정한 결과 <그림 1>과 같이 나타났다.

학습자들은 모든 수업 내용에 대한 재미나 흥미가 3.8점 이상으로 나타나 긍정적으로 인식하고

있는 것으로 나타났다. 수업 내용에 대한 난이도에서는 W형태로 나타났는데, 분석결과 상대적으로 새로운 개념이나 어려운 기능을 배우는 차시인 2, 4장에서 어렵다고 생각하였으며, 3, 5장에는 이전 주에서 배우 내용의 확장이 이루어지므로 상대적으로 쉽다고 인식하는 것으로 나타났다.



<그림 1> 재미, 흥미, 난이도에 대한 인식변화

학습자들의 인식 변화에 대한 사전, 사후 t-검정 결과 재미와 난이도의 인식에서는 <표 6>과 같이 유의미한 결과가 나타나지 않았다. self-프로그래밍 능력 및 self-CT 역량에서는 유의미한 결과가 나타났다($t=2.142, p=.034$; $t=6.947, p=.000$). 스스로 프로그래밍을 잘한다고 생각하는 부분에서는 사전($m=2.67$)로 보통보다 낮은 수준이었으며, 사후($m=2.88$)로 보통보다 낮은 수준이었으나 사전점수보다는 유의미하게 상승하였다.

<표 6> 학습자의 인식 변화 (n=112)

요소	항목	평균	표준 편차	t	p
		사전	사후		
재미	사전	4.11	.824	1.565	.120
	사후	4.25	.765		
쉬움	사전	3.18	1.015	.897	.372
	사후	3.09	.885		
self-프로그래밍	사전	2.67	.924	2.142*	.034
	사후	2.88	.931		
self-CT	사전	3.94	.700	6.947***	.000
	사후	4.44	.577		

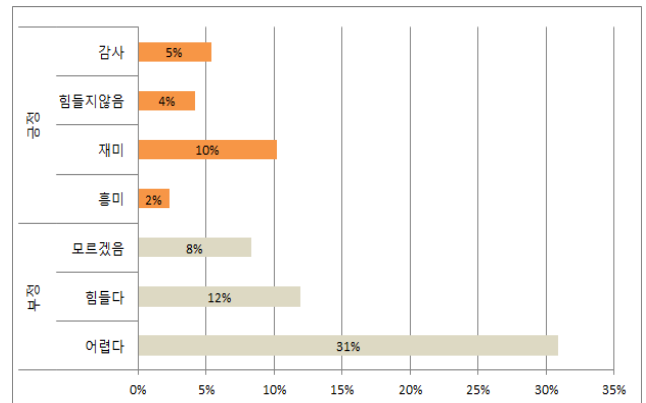
*($p<.05$), ***($p<.001$)

비전공 학습자들은 7주간의 학습에도 불구하고 스스로 프로그래밍에 대한 능력 인식이 낮게 나타났다. 이는 7주간의 CT 학습이 부족함을 보여주며, Jenkins(2002)가 제기한 학습속도의 문제 [23]가 동일하게 나타남을 보여준다.

4.2 CT 교육에 대한 이미지

비전공 학습자들이 CT 교육에 대해 가진 이미지는 여러 연구에서 나타난 것처럼 ‘생소하고 지루하며 어렵다’는 것이다.

따라서 본 연구에서는 학습자들의 특성에 맞는 학습내용으로 다양한 멀티미디어 콘텐츠를 제작하도록 한 결과 양적분석에 나타난 것처럼 재미, 흥미 요소는 매우 긍정적으로 나타났다. 양적결과와 비교하여 학습자들의 실제 성찰 저널에서도 같은 결과가 나타나는지 <표 5>의 범주에 따라 분석하였다. 실제 성찰 저널을 내용 분석법으로 분석한 결과 총 687개의 성찰 저널에서 긍정적인 이미지는 22%로 나타났으며, 부정적인 이미지는 51%로 나타났다. 이는 <그림 2>에서 나타난 난이도의 결과와 동일하게 나타났다. 학생들은 재미와 흥미는 있지만 학습내용이 쉽지는 않다고 느끼는 것으로 나타났다.

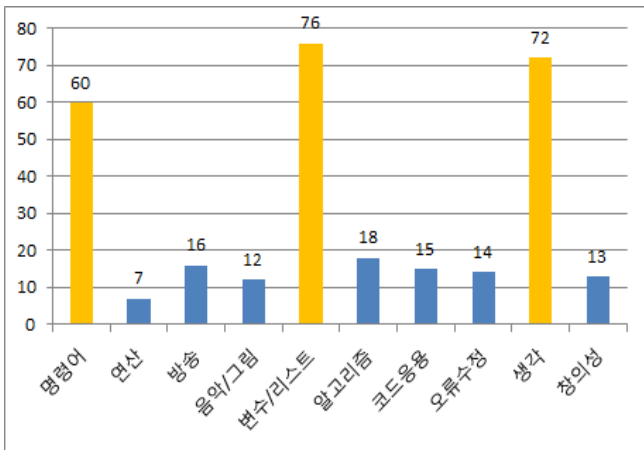


<그림 2> CT 교육에 대한 이미지 인식

<그림 3>에서와 같이 실제 비전공 학습자들이 느끼는 어려움 중에서 가장 높은 빈도를 보인 부분은 변수와 리스트에 대한 내용이었다. 4장과 5장에서 학습한 변수와 리스트를 통한 학습에서 학습자들이 가장 많은 어려움을 느끼는 것으로 나타났다. 두 번째로는 ‘생각’으로 분류된 요소인데, 이는 설계부터 해결방안 모색에 대한 부분이다. 기존의 프로그래밍 교육에서는 계획 및 알고리즘 부분이라고 볼 수 있다. 학습자들은 명령어 사용법이 익숙해진 후 자신의 아이디어를 구현하기 위해서는 어떤 코드로 작성해야 하는지에 대한 어려움과 어떤 프로그램을 만들지에 대한 아

이디어 단계에서의 어려움을 호소하였다.

세 번째로 명령어에 어려움을 호소한 문장이 많이 나타났는데, 대부분의 프로그래밍 학습에서 초보 학습자들이 어렵다고 느끼는 명령어 기능, 명령어의 응용에 대한 내용이다. 스크래치와 같은 직관적인 교육용프로그래밍 언어에서도 명령어의 기능이나 어떤 명령어를 선택해야 하는지에 대한 문제가 동일하게 나타나는 것으로 분석된다. 따라서 명령어 학습에 대한 교수학습 전략이 필요함을 시사한다.



<그림 3> 어려움에 대한 인식

4.3 상관관계 분석

흥미, 재미, 난이도와 self-프로그래밍 능력, self-CT 능력을 상관관계 분석 결과 <표 7>과 같이 유의미한 결과가 나타났다. self-CT와 self-프로그래밍은 약한 상관관계가 나타났으며, self-CT와 흥미, 재미와의 관계에서는 보통 상관관계가 나타났다(상관계수 = .561; .519). 이는 학습자들이 느끼는 흥미와 재미가 스스로 CT 능력이 높다고 인식하는 것과 상관관계가 있는 것으로 분석할 수 있다.

self-프로그래밍 능력과 난이도, 흥미, 재미와의 관계에서는 self-프로그래밍 능력과 난이도와의 관계가 보통 상관관계를 보였고(상관계수=.571), 흥미와 재미와의 관계에서는 약한 상관관계를 보였다(상관계수 = .389; .360). 이런 결과는 학습자들이 쉽다고 생각하는 인식과 스스로 프로그래밍을 잘한다고 인식하는 것이 상호 영향을 주는 것

으로 분석할 수 있다. 즉, 학습 내용이 쉬운 경우 스스로 프로그래밍을 잘한다는 자신감을 가질 수 있음을 시사한다.

<표 7> 상관관계 분석

		self-CT	self-프로그래밍	난이도 (쉬움)	흥미	재미
self-CT	Pearson 계수	1	.254**	.181	.561***	.519***
	p		.007	.056	.000	.000
self-프로그래밍	Pearson 계수		1	.571***	.389***	.360***
	p			.000	.000	.000
난이도 (쉬움)	Pearson 계수			1	.272**	.339***
	p				.004	.000
흥미	Pearson 계수				1	.801***
	p					.000
재미	Pearson 계수					1
	p					

** (p<.01), *** (p<.001)

또한 흥미와 재미와의 관계는 매우 강한 상관관계를 보여(상관계수 = .801), 학습내용이 흥미 있는 내용으로 구성되면 재미도 증가하는 것으로 분석할 수 있다.

5. 결론 및 제언

본 연구에서는 컴퓨터 비전공자들을 위한 CT 교육 과정에서 나타난 어려움을 분석하고, 이를 해결하기 위한 방안을 모색하였다. 본 연구를 통해 나타난 시사점은 다음과 같다.

첫째, 비전공자들이 CT 교육에서 가장 어려워하는 부분은 변수와 리스트의 학습, 아이디어를 생각하고 구현하는 과정, 적합한 명령어의 사용의 순으로 나타났다.

둘째, Jenkins(2006)의 주장[23]과 같이 학습내용에 대한 재미와 흥미는 self-CT 역량과 유의미한 상관관계가 있는 것으로 나타났다. 따라서 CT 교육의 내용은 학습자의 흥미와 재미를 유발할 수 있도록 구성하여야 한다.

셋째, CT 교육을 위해서는 비주얼 교육용프로

그래밍 언어이거나 직관적인 도구라 하더라도 기본적인 명령어를 응용하기 위해서는 일정시간의 학습시간이 필요하다. 따라서 정규 공교육에서 교육용프로그래밍 언어를 활용한 CT 교육을 실시할 때는 명령어에 대한 학습시간과 이를 응용한 CT 능력 증진을 위한 학습시간을 확보해야 하는 문제를 해결해야 한다.

본 연구의 결과를 바탕으로 비전공자 초보학습자를 대상으로 한 CT 교육에서의 제언을 다음과 같이 도출할 수 있다.

첫째, 교육내용의 구성이 나선형 교육과정의 형태로 이루어져야 하며, 사용하는 프로그래밍 언어의 명령어 학습 시간이 필요하다. 나선형 교육과정은 핵심이 되는 개념과 원리를 반복적으로 확장하는 형태로 교육과정이 구성되므로, 학습자들의 인지부담을 줄여줄 수 있다. 최정원과 이영준(2014)의 연구[13]에서도 나타난 바와 같이 학습내용의 반복적이고 확장적인 구성이 필요함을 시사한다.

또한 명령어의 사용법을 가르치는 구성보다는 구성주의 원리에 입각하여 스스로 탐색하여 기능을 발견하는 방식이나 동료와의 교류를 통해 익히는 방법을 제시하여 학습시간을 확보해야 한다.

둘째, 명령어 학습에 모든 시간을 사용할 수 없으므로 개인 또는 그룹이 스스로 학습할 수 있는 명령어 쿼북과 같은 형태의 교수학습 자료가 필요하다. 스크래치 사이트에서 제시하는 명령어 카드를 이용하여 학습하는 방법도 대안이 될 수 있다[28].

셋째, CT 교육은 기존의 아날로그 정보를 다루는 교육에서 경험하지 못한 내용을 학습하게 되므로 디지털 정보를 기반으로 한 문제해결력과 고등사고력으로의 전환에 대한 다양한 기회와 시간을 제공해야 한다. 즉, 컴퓨터과학의 개념과 원리를 학습하기 위해 프로그래밍만을 학습 내용으로 구성하는 것이 아니라 활동중심의 학습내용을 구성하는 전략도 고려해야 한다[2].

마지막으로 SW 교육의 목표는 CT, 창의력, 문제해결력, 협업능력이므로, 이를 기반으로 하는 통합 프로젝트형 주제가 필요하다. 다양한 예제의 구현을 통하여 사용 언어의 한계점과 구현가능성

에 대한 충분한 탐색이 필요하다. 프로젝트 학습을 실시하기 위해서는 교과간의 연계나 교육시수의 확보 등 다양한 전략이 고려되어야 한다.

참 고 문 헌

- [1] 강선미 (2004). 7차 가정교과에서의 양성평등 교육내용분석, 중앙대학교교육대학원, 석사학위논문
- [2] 경인교대 미래인재연구소 (2015). **창의컴퓨팅 이슈리포트**, 2015-1. <http://computing.or.kr>
- [3] 고아연 (2013). 주얼리 의미에 대한 연구 : 내용분석법의 활용, 서울과학기술대학교 산업대학원, 석사학위 논문
- [4] 김민자, 김자미, 김현철 (2015). 에스토니아의 ProgeTiger를 통한 SW교육 교사지원 방향 고찰. **2015년 한국컴퓨터교육학회 동계 학술발표논문지**, 19(1), 11-14.
- [5] 김영천 (2001). **교과수업과 수업에서의 질적 연구**. 문음사
- [6] 김수환, 한선관 (2012). Computational Thinking 향상을 위한 디자인기반 학습, **정보교육학회 논문지**, 16(3), 315-326.
- [7] 김수환, 한선관, 김현철 (2011). 프로그래밍 과정에서 나타나는 초보학습자들의 행동 및 사고과정 분석. **컴퓨터교육학회 논문지**, 14(1), 13-21.
- [8] 백남진 (2005). 교육과정 문서에서의 교육내용 제시 방식 분석 -한국·미국 과학(생물) 교육과정 내용 진술문을 중심으로- 이화여자대학교 박사학위 논문
- [9] 이영준 외 (2014). **초중등 단계 Computational Thinking 도입을 위한 기초 연구**. 한국과학창의재단
- [10] 장혜선, 최숙경, 전수진, 염용철, 이원규 (2007). 에러 피드백 기반의 초보자를 위한 프로그래밍 학습 지원 시스템, **한국컴퓨터교육학회 논문지**, 10(6), 1-10.
- [11] 정란 (1996). 초보자 프로그래밍 과정의 특성 분석과 지원 방안. **삼척대학교 논문집**, 29(1), 387-411
- [12] 최숙영 (2011). 21st Century Skills와

- Computational Thinking 관점에서의 ‘정보’ 교육과정 분석. **한국컴퓨터교육학회 논문지**, 14(6), 19-30.
- [13] 최정원, 이영준 (2014). 프로그래밍 학습에서 학습자의 어려움 분석. **한국컴퓨터교육학회 논문지**, 17(5), 89-98.
- [14] 최정원, 이은경, 이영준 (2015). 초등 정보 교육을 위한 영국 컴퓨팅 교과서 분석. **2015년 한국컴퓨터교육학회 동계 학술발표논문지**, 19(1), 19-22.
- [15] 한국과학기술학림원 (2014). **K-12 Computer Science Education in Korea**. KAST's 20th Anniversary International Symposium. Seoul, Korea
- [16] 한국컴퓨터교육학회, 정보통신산업진흥원. (2014). **국내외 SW교육 운영현황 및 요구사항 조사 보고서**.
- [17] Brennan, K., Balch, C., Chung, M. (2015). **창의컴퓨팅 가이드북** (한선관, 김수환 외 역). 서울: 퍼플. (원서출판 2014).
- [18] College Board, “AP Computer Science Principles Draft Curriculum Framework”, 2015. 2. 20일 검색 <https://advancesinap.collegeboard.org/stem/computer-science-principles>
- [19] CSTA (2011). CSTA K-12 Computer Science Standards, 2011, 2015. 2. 20 검색 <http://csta.acm.org/Curriculum/sub/K12Standards.html>
- [20] Elo, S., Kyngas, H. (2007). *The qualitative content analysis process*. *Journal compilation*, Blackwell
- [21] Estonian Ministry of Education and Research, “National Curricula for Basic Schools and Upper Secondary Schools 2011”, Retrieved 2014. 4. 1 검색 <http://www.hm.ee/index.php?1511576>
- [22] Naace, Havering education service, and Computing At School. (2014). *Switched on Computing Year1*. Rising Star UK Ltd.
- [23] Jenkins, T. (2002). On The Difficulty of Learning to Program. *3rd Annual LTSN-ICS Conference, Loughborough University*. 53-58.
- [24] Resnick, M. (2002). *Rethinking Learning in the Digital Age*. In *The Global Information Technology Report: Readiness for the Networked World*, edited by G. Kirkman. Oxford University Press.
- [25] 국민대 대학보도. 2015. 2. 20일 검색 <http://press.kookmin.ac.kr/site/main/view.htm?num=11798>.
- [26] 시카고 CS 교육 기사. 2015. 2. 20일 검색 http://www.koreadaily.com/news/read.asp?art_id=2182615
- [27] Computational Thinking Center. 2015. 2. 20일 검색 <http://www.cs.cmu.edu/~CompThink/>
- [28] scratchEd. 2015. 2. 20일 검색 <http://scratch.ed.gse.harvard.edu/>

김수환



1999 인천교육대학교(교육학학사)
 2006 경인교육대학교
 컴퓨터교육과(교육학석사)
 2011 고려대학교
 컴퓨터교육과(이학박사)

2013~2014 경인교육대학교 겸임교수
 2014~현재 총신대학교 교양교직과 조교수
 관심분야: 컴퓨터교육, EPL, 컴퓨터적사고, CSCL,
 Computational Literacy
 E-Mail: skim@csu.ac.kr