

예비교사의 프로그래밍 교수내용지식 향상을 위한 프로그래밍 교육프로그램 설계

안상진[†] · 이영준^{††}

요 약

본 연구는 프로그래밍 교육을 효과적으로 실시하기 위한 실제적 능력을 향상시키기 위하여 프로그래밍 교육 프로그램을 설계하고 적용하여 예비교사의 프로그래밍 교수내용지식의 변화를 확인한 연구이다. 제작된 교육 프로그램에서는 블록 기반 모바일 프로그래밍 언어인 앱 인벤터를 사용하였고, 문제 기반 학습 방법과 프로젝트 기반 학습 방법을 활용하였다. 개발된 교육프로그램을 예비정보교사에게 교육하고 프로그래밍 교수내용지식을 확인하기 위한 평가문항으로 자기평가를 실시하였다. 그 결과 문제 기반 학습 방법으로 프로그래밍을 학습한 후 프로그래밍 교수내용지식 점수와 교수법영역 지식의 점수가 유의하게 향상되었고, 프로젝트 기반 학습 방법을 적용한 후 내용지식영역, 교수법영역, 교육과정영역의 점수가 유의하게 향상되었다.

주제어 : 프로그래밍 교육, 정보교사교육, 교수내용지식

Designing Programming Curriculum for Developing Programming Pedagogical Content Knowledge of Pre-service Informatics Teachers

Sangjin An[†] · Youngjun Lee^{††}

ABSTRACT

This study is for developing a programming education course to improve pre-service teachers' pedagogical content knowledge(PCK) of programming education. A 40-hour training course was designed with App Inventor, a block-based mobile programming environment, and with problem-based learning method and project-based learning method. After the curriculum was adopted to 12 undergraduate students, the effect of education was tested with a programming PCK questionnaire. As a result, after a 20-hour problem-based learning class, overall score and teaching method score were enhanced significantly. After another 20-hour project-based learning class, content knowledge, teaching method, and curriculum score were improved..

Keywords : Programming Education, Informatics Teacher Education, Pedagogical Content Knowledge

[†] 정 회 원: 한국교원대학교 컴퓨터교육과 박사과정

^{††} 종신회원: 한국교원대학교 컴퓨터교육과 교수(교신저자)

논문접수: 2015년 9월 30일, 심사완료: 2016년 1월 15일, 게재확정: 2016년 2월 3일

* 본 논문은 2013년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행되었음(No. 2013R1A2A2A03068459)

1. 서론

전 세계적으로 정보 교육의 중요성이 점차 강조되고 있고, 그 중에서도 프로그래밍과 코딩에 관련된 관심과 교육 시도가 증가하고 있다. 이전까지의 컴퓨팅 교육이 컴퓨팅 기기, 그 중에서도 컴퓨터에서 사용할 수 있는 몇몇 소프트웨어를 활용하는 방식을 학습하는 것을 중심으로 이루어졌다면, 앞으로 진행될 컴퓨팅 교육은 다양한 컴퓨팅 기기를 활용하여 문제를 해결하는 교육을 중심으로 이루어지게 될 것이다. 컴퓨팅을 활용한 다양한 형태의 문제해결방식 중에서도 프로그래밍은 프로그래밍 언어라는 논리적 도구를 활용하여 학습자가 접할 수 있는 실세계의 복잡한 문제를 다양한 방식으로 해결하는 경험을 가질 수 있게 한다. 프로그래밍은 학습자에게 컴퓨팅이 가지고 있는 여러 가지 능력을 이해하게 하고, 실세계 문제 해결에서 컴퓨팅이 가지는 능력을 적극적으로 활용하도록 만드는 중요한 내용 요소이다 [1]. 프로그래밍을 교육하기 위한 다양한 교육 요소들 준비하는 과정에서, 프로그래밍 교육의 중요한 요소 중 하나인 정보교사의 프로그래밍 교육능력을 향상시키는 방안을 고민해야 할 필요가 있다.

프로그래밍은 컴퓨터 과학에서 중요하고 필수적인 부분이지만 학습자가 학습하기 어려워하는 교육내용이기도 하다. 프로그래밍 학습의 어려움은 다양하게 보고되고 있지만 많은 경우 프로그래밍 언어가 가지고 있는 프로그래밍 문법을 습득하기가 쉽지 않고, 알고리즘이나 프로그래밍 전략을 직관적으로 이해하기 어렵기 때문에 프로그래밍을 학습자가 어려워한다고 알려져 있다[2][3]. 프로그래밍 학습의 어려움은 곧 프로그래밍 교육의 어려움이기도 하다. 프로그래밍을 가르치는 교사가 프로그래밍 언어를 습득하고 프로그래밍의 절차적 지식을 갖추고 있다고 하더라도, 이것이 학습자에게 프로그래밍을 효과적으로 가르칠 수 있는 능력을 보장하는 것은 아니다. 효과적인 프로그래밍 교육이 되기 위해서는 프로그래밍에 대한 내용지식의 기반 위에 교육학지식과 상황지식 등의 교육적인 지식이 결합되어야 한다. 이렇게 효과적인 교육을 위해 필요한 교사의 지식을 교수내용지식(pedagogical content knowledge) 라고

부르는데, 프로그래밍을 교육하기 위한 교사의 능력을 프로그래밍 교수내용지식이라 부를 수 있다. 일반적으로 교수내용지식은 실제 교실에서 해당 내용을 교육하면서 발달하는 것으로 생각되지만, 이러한 교수내용지식이 발달하기 위해서는 교사 준비기부터 교수내용지식 발달을 염두에 둔 형태로 교육이 이루어져야 한다. 따라서 예비정보교사의 경우 효과적인 프로그래밍 교육을 위해 프로그래밍이라는 영역을 학습할 때 프로그래밍에 대한 내용지식을 습득하는 것과 동시에 프로그래밍을 효과적으로 교육하기 위해 필요한 지식 또한 습득해야 한다. 하지만 아직까지 대부분의 예비정보교사를 위한 프로그래밍 교육프로그램은 프로그래밍 내용지식의 향상에 더욱 비중을 두고 있다. 효과적인 프로그래밍 교육 능력을 갖춘 정보교사를 양성하기 위해서는 교육프로그램이 내용 지식과 함께 교육 능력을 갖추도록 변화할 필요가 있다.

따라서 본 연구에서는 예비정보교사의 프로그래밍 교수내용지식을 향상시킬 수 있도록 프로그래밍 내용지식과 교수방법을 학습할 수 있는 교육프로그램을 설계하였다. 설계한 교육프로그램을 학습자에게 교육하고 학습자의 프로그래밍 교수내용지식을 측정하여 예비정보교사의 프로그래밍 교수내용지식의 향상을 측정하였다.

2. 이론적 배경

2.1 프로그래밍 교수내용지식

교사가 가져야 하는 전문성에 대한 연구 중 교사 지식의 구조에 대한 논의가 있다. 일반적으로 교사가 가져야 하는 지식을 분류하였을 때 교과 지식(subject matter knowledge), 교육학 지식(pedagogical knowledge), 맥락 지식(knowledge of context)으로 나누어서 논의하고 있다[4][5]. 교과 지식은 교사가 특정한 내용을 가르치기 위해서 가지고 있어야 하는 해당 분야에 대한 지식을 말한다. 현대의 교육에서는 단순히 해당 분야의 사실이나 절차적 지식을 암기하는 것뿐만 아니라 교육하고자 하는 분야의 핵심 개념, 내용간의 관계, 추론능력, 활용능력 등의 지식 또한 내용 지

식에 포함되는 것으로 판단하고 있다[5]. 교육학 지식은 일반적인 교육학적 지식으로, 교수학습전략, 학습이론, 학생 발달에 대한 지식 등을 포함한다. 일반적인 교육학 지식은 다양한 교육 분야에서 발전된 이론이 일반화된 것이기 때문에 교과에 특이적이지 않은 부분이 있다. 따라서 일반적인 교육학 지식이 개별 교과에 적용되면 교과마다 조금씩 다른 형태를 보일 수 있다. 맥락 지식은 교사가 학교에서 교육을 진행하면서 체득하게 되는 학생, 학교, 공동체에 대한 지식을 말한다. 맥락 지식은 객관적인 지식이라기보다는 주관적인 지식으로 교사가 다양한 형태의 교육현장에 대한 경험이 축적되어 가며 발전해나가는 지식으로 볼 수 있다.

교수내용지식의 개념은 위에서 말한 내용 지식, 교육학 지식, 상황 지식이 서로 상호작용하여 발달하는 역동적인 지식으로 정의할 수 있다. 교수내용지식은 교사가 교육하고자 하는 교육내용을 적절하게 교육하기 위한 실제적인 지식을 말한다. 교수내용지식은 특정한 지식을 알고 있는지, 모르고 있는지에 관련된 것이 아니라 교사가 알고 있는 지식을 어떻게 활용할지에 관련된 지식이다. 교수내용지식에 어떠한 영역이 포함되는지에 대한 많은 논의가 있지만 많은 경우 교과내용, 교육과정, 학생이해, 교수전략, 교육목적, 교육 상황, 평가에 대한 지식을 필요로 한다[5][6][7].

교수내용지식의 관점에서 교사가 프로그래밍을 교육하기 위해 필요한 교수내용지식의 영역으로는 프로그래밍에 대한 내용지식, 프로그래밍의 교육 목적과 교육과정의 구성, 프로그래밍을 가르치기 위한 교수방법, 학생이 프로그래밍을 학습하는 과정에 대한 이해, 학생의 프로그래밍 학습을 평가하기 위한 지식 등이 포함될 것이다. 프로그래밍 교수내용지식은 프로그래밍에 대한 내용지식과 교육학지식, 또한 프로그래밍 교육 상황에 대한 지식과 프로그래밍을 교육하면서 나타나는 학습자의 다양한 변화에 대한 지식이 융합되어 교사에게 형성되는 지식이라 할 수 있다.

2.2 예비정보교사의 프로그래밍 학습

교사발달 연구의 접근법에서는 교직의 경력이

나 연령을 중심으로 단계를 구분하거나, 교사의 발달단계 이론에 따라 인지발달적 연구로 단계를 구분하는 접근법이 주로 사용된다. 다양한 연구자들이 교사발달 단계에 대하여 논의하고 있지만 일반적으로 교사발달단계는 크게 준비기, 교직초기, 중기, 후기로 나누어서 살펴보고 있다. 일반적으로 교사발달에 관련된 연구는 교사가 실제 교직에 입문하고 나서 개발하는 교수방법, 학생지도, 교육과정에 관련된 직무기능에 초점을 맞추고 있다[8]. 하지만 현재 우리나라의 교사양성과정은 교사를 양성하기 위한 교육기관이 따로 존재할 정도로 예비교사양성에도 중요성을 부여하고 있다. Jaworski와 Gellert(2003)에 의하면 예비교사들은 초기 교육프로그램에 의해 자신의 교육적 선입관과 암묵적 지식, 교육적 정체성을 갖게 되기 때문에 예비교사 교육은 중요하며 교육프로그램을 효과적으로 설계하는 것이 관건이라고 하였다[9]. 다시 말해 교직 준비기는 교수내용지식을 실제 교육현장에서 발달시키기 위한 기초적인 기반을 다지는 시기이다. 예비교사가 준비기에 충분한 내용지식과 교육학지식을 습득하지 못하면 교직에서 교수내용지식을 발달시키기가 어려워지고, 이는 높은 수준의 교육이 이루어질 수 없는 원인이 된다. 또한 예비교사교육에서도 교수내용지식에 대한 학습이 이루어질 수 있어야 한다. 대부분 예비교사교육에서는 내용지식의 습득을 위한 교육을 진행하고 있는데, 실제적인 교육을 수행하기 위해서는 예비교사도 단순히 해당 영역의 지식을 습득하는 것뿐만 아니라 그 지식을 실제 교실에서 어떤 교육 방법으로 표현하고, 그에 따른 학습자의 변화가 어떻게 일어나는지에 대한 구체적인 지식의 습득이 필요하다. 따라서 예비정보교사가 프로그래밍을 학습할 때는 내용지식과 함께 앞에서 말한 프로그래밍 교수내용지식인 교수법지식, 프로그래밍 교육단계에 대한 지식, 평가에 대한 지식 등을 함께 학습하고 이를 실제 교육상황에서 적용할 수 있도록 교육과정을 구성할 필요가 있다.

2.3 국내외 관련 연구

현재 국내외에서 진행 중인 앱 인벤터 관련 연

구에서 살펴볼 수 있는 점은 국내 연구에서는 초·중등 교육에 앱 인벤터를 활용하는 방법에 대한 초점을 맞추고 있고, 국외에서는 대학생 및 교사를 대상으로 하는 연구에 조금 더 비중을 두고 있다는 점이다[10]. 이는 국내와 국외의 컴퓨팅 교육 대상과 교육 목적의 차이로 판단할 수 있는데, 국내에서는 소프트웨어 교육을 위시한 컴퓨팅 교육이 초·중등학교로 확산됨에 따라 초·중등 학생을 대상으로 하는 연구가 계속해서 진행되고 있다 [10][11]. 국외에서는 아직까지 초·중등학교에 컴퓨팅 교육이 원활하게 진행되고 있지 않고, 앱 인벤터를 대학교 프로그래밍 교육의 입문 도구로 활용하거나 프로그래밍 교육을 보조할 수 있는 도구로써의 가치를 연구하고 있는 것으로 판단된다 [12][13]. 국내에서는 앱 인벤터를 프로그래밍을 교육하기 위한 도구로 인식하고 있으나 이를 교사나 예비교사에게 교육하고 프로그래밍 교수내용지식에 미치는 효과를 검증한 연구가 아직 실시되지 않았다.

3. 연구 방법

본 연구는 예비정보교사 교육에서 프로그래밍 교수내용지식을 향상시키기 위한 교육프로그램을 설계하고 그 효과를 검증한 연구이다.

3.1 연구 대상

본 연구는 K 대학에 개설된 컴퓨터교육학과 전공과목에서 실시하였다. 연구에 참여한 예비정보교사는 총 12명으로 남학생 6명, 여학생이 6명이다. 참여 학생은 모두 컴퓨터 교육을 전공하는 학부생들이고 3학년 8명, 4학년이 4명이다. 학생들은 모두 프로그래밍을 학습한 경험이 있었다. 학생들이 학습하고 활용해 본 프로그래밍 언어는 스크래치, C 언어, Java 언어가 있었다.

3.2 평가 도구

교수내용지식은 객관적으로 측정할 수 있는 지식이라기보다는 교사가 실제 교실에서 해당 내용을 교육할 수 있는 능력에 대한 자기효능감에 가

깝다고 할 수 있다. 따라서 교수내용지식을 측정하기 위한 방법은 객관적인 시험보다는 자기평가지 형태의 질문지법을 활용하는 것이 적절하다. 현재 정보교사의 프로그래밍 교수내용지식을 평가하기 위해 연구가 많지 않아 객관적으로 제작된 평가지가 없기 때문에 연구자들은 교수내용지식과 관련된 선행 연구들을 토대로 프로그래밍 교수내용지식을 평가하기 위한 문항을 개발하였다[14][15]. 개발된 평가 도구는 Likert 5점 척도로 이루어진 자기질문지 형태의 문항지이고, 답변은 ‘매우 그렇지 않다’부터 ‘매우 그렇다’까지 답하도록 하였다. 개발된 문항은 총 20문항이고, 프로그래밍 내용지식, 교수법 지식, 교육과정 지식과 평가 지식에 관련된 문항으로 구성하였다. 개발된 문항은 <표 1>과 같다.

<표 1> 프로그래밍 교수내용지식 평가문항

영역	평가 내용	문항수
내용지식	프로그래밍 개념 이해 프로그래밍 과정 이해 프로그래밍 활용 방법 이해 프로그래밍 필요성 이해	7
교수법지식	흥미 유발 적절한 예시 교수방법	6
교육과정 지식	프로그래밍 교육과정 이해 프로그래밍 교육의 단계 이해 교육과정 내에서 프로그래밍과 다른 과목과의 관계	5
평가지식	프로그래밍 개념 평가 프로그래밍 실습 평가	2

개발된 문항의 영역은 크게 네 가지로 이루어진다. 먼저 프로그래밍 내용지식에 관련된 문항이 있다. 여기에는 프로그래밍 개념 이해, 프로그래밍 과정 이해, 프로그래밍 언어 활용 방법에 대한 이해, 프로그래밍의 필요성에 대한 이해에 관련된 문항이 포함되어 있다. 교수법에 관련된 문항은 실제 교실에서 학습자를 교육하는 상황에서 적절하게 대처할 수 있는지를 묻는 문항이다. 여기에는 프로그래밍 교육을 위한 흥미 유발 활동, 프로그래밍 개념의 적절한 예시, 설명한 내용을 프로그램 코드로 표현하는 능력, 적절한 교수방법의 활용 등의 문항이 포함되어 있다. 교육과정지식은 국가교육과정 내에 프로그래밍 관련 내용을 이해

하고 있는지를 묻는 질문이다. 이 영역에는 프로그래밍 교육과정과 학교급별 교육 단계 이해, 프로그래밍 내용과 다른 과목과의 연계성 등에 관련된 문항들이 포함되어 있다. 마지막으로 평가에 관련된 지식을 묻는 영역이 있다. 여기에서는 프로그래밍 개념을 이해했는지를 평가하는 방법과 프로그래밍의 결과물을 평가하는 방법을 알고 있는지를 묻는 문항이 포함되어 있다. 일반적으로 교수내용지식을 평가하기 위한 문항에는 학생의 학습과 변화에 관련된 문항이 포함되어 있으나 이 교육프로그램에서는 학생에게 프로그래밍을 교육하는 실습은 포함되어 있지 않아 학생지식에 관련된 문항은 포함하지 않았다.

3.3 교육 내용 설계 및 적용

본 연구에서 설계한 교육 프로그램은 총 40시간 분량이다. 먼저 프로그래밍을 교육하기 위한 도구를 선정하였다. 학습자들은 프로그래밍 언어에 대한 지식, 프로그래밍 개념에 대한 지식, 프로그래밍 절차에 대한 지식을 발달시키는 동시에 프로그래밍을 학습자에게 교육하는 방법 또한 익혀야 한다. 이러한 과정에서 일어날 수 있는 인지적인 부하를 줄이고 다양한 형태의 프로그래밍 언어를 활용하는 방법을 익히기 위하여 본 교육프로그램에서는 교육용 프로그래밍 언어인 앱 인벤터를 사용하였다. 앱 인벤터는 블록 기반의 프로그래밍 방식을 가지고 있는 교육용 프로그래밍 언어로써 익히기 쉽고 실제 모바일 기기에서 동작을 확인할 수 있는 장점이 있어 프로그래밍 교육에서 다양하게 활용되고 있다[10][11]. 앱 인벤터를 활용하는 또 다른 장점은 실제 학습자들이 학습하는 프로그래밍 언어에 대한 지식을 쌓을 수 있다는 점이다. 인지적인 능력이 충분히 발달하지 않은 초중등 학생들에게는 블록 기반의 그래픽 프로그래밍 언어가 많이 활용된다. 예비정보교사가 실제로 활용하는 프로그래밍 언어를 학습하는 과정을 통해 프로그래밍 과정에서 학습자가 겪을 수 있는 다양한 문제 상황에 대한 지식을 쌓을 수 있게 되기 때문에 교육 지식의 폭을 넓힐 수 있다.

교육프로그램에 활용한 교수학습방법은 문제 기반 학습(problem-based learning)과 프로젝트

기반 학습(project-based learning)이다. 문제 기반 학습과 프로젝트 기반 학습은 실제 상황과 유사한 문제나 프로젝트를 해결하는 과정을 통해 학습자가 해당 영역의 교육내용을 스스로 학습할 수 있도록 유도하는 교육방법이다. 문제 기반 학습과 프로젝트 기반 학습은 학습자가 능동적으로 학습을 진행하고, 실제적인 해결책을 생성하고 평가하는 과정을 거치기 때문에 프로그래밍 교육에서 활발하게 활용되고 있다[16][17]. 초반의 20시간은 개인별 문제 기반 학습방식으로 구성하였다. 기존의 초중등 학생 대상의 교육내용을 대학생 수준에 맞게 재구성하여 앱 인벤터 개발환경과 앱 인벤터 기능을 학습하기 위한 다양한 앱을 학생이 제작하고, 앱의 동작에 포함된 다양한 프로그래밍 개념 및 학습 방법에 대해 학생이 이해하도록 교육하였다[10]. 후반 20시간은 집단별 프로젝트 기반 학습방식으로 구성하였다. 학습자들을 소규모 집단으로 나누고 스스로 제작하고자 하는 앱 프로젝트를 계획한 다음 이를 제작하고 평가하는 과정을 거치도록 하였다. 교육 내용의 구성은 <표 2>에 정리되어 있다.

강의를 진행하기 전에 제작한 평가지를 활용하여 사전 평가를 실시하였다. 1차시부터 4차시까지는 교육용 프로그래밍 언어의 의미와 교육용 프로그래밍 언어의 필요성에 대해 이해하고, 앱 인벤터의 사용 환경을 익히도록 하였다. 5차시부터 8차시까지 앱 인벤터에 포함된 다양한 컴포넌트를 활용하는 방법을 익히기 위한 간단한 앱을 제작하도록 하였다. 9차시부터 20차시까지는 문제 기반 학습의 형태로 수업을 구성하여, 주어진 앱을 제작하기 위해 학생들이 앱의 구동을 위한 논리적 알고리즘을 제작하고 실제 앱을 만들어 시연하는 방식으로 수업을 진행하였다. 20차시의 수업이 끝난 이후에 중간 평가를 실시하였다. 21차시부터는 프로젝트 기반 학습 형태로 수업을 진행하였다. 학습자를 일정한 모둠으로 나누고, 20차시까지 학습한 내용을 바탕으로 연구 대상자들이 제작하고자 하는 안드로이드 앱에 대한 계획을 세우게 하였다. 모둠별로 작성한 프로젝트 계획을 학습자끼리 서로 피드백한 후에 12차시에 걸쳐 모둠별 앱을 제작하도록 하였다. 이 과정이 끝난 후 각 모둠의 결과물을 발표하고 발표한 결

<표 2> 프로그래밍 교육 내용

차시	교육 내용	내용 지식
사전 평가		
1-4	교육용 프로그래밍 언어의 교육적 의미 앱 인벤터 소개 앱 인벤터 사용 환경 설정 앱 인벤터 사용 방법 익히기 메모 앱 / 메모 읽어주기 앱 제작	교육용 프로그래밍 언어의 이해 순차 실행 이벤트 핸들러 센서의 활용
5-8	그림판 앱 제작 계산기 앱 제작	컴포넌트 활용법 익히기 알고리즘
9-12	안전 관리 앱 제작 운명의 수정구슬 앱 제작	메소드 호출 데이터베이스
13-16	숫자 게임 앱 제작 야구 게임 앱 제작	알고리즘
17-20	종합 앱 제작 및 평가	알고리즘
중간 평가		
21-24	앱 개발 프로젝트 계획 세우기 프로젝트 계획 발표 및 피드백 프로젝트 진행	소프트웨어 개발 계획 소프트웨어 개발 단계
25-28	프로젝트 진행	소프트웨어 개발 단계
29-32	프로젝트 진행	소프트웨어 개발 단계
33-36	프로젝트 진행	소프트웨어 개발 단계
37-40	프로젝트 결과물 발표 학생 상호간 피드백 교육용 프로그래밍 언어의 활용에 대한 논의	소프트웨어 평가
사후 평가		

과물에 대한 평가를 진행하였다. 마지막으로 교육용 프로그래밍 언어의 프로그래밍 교육 활용에 대한 논의를 진행한 후 사후 평가를 실시하였다.

4. 연구 결과

연구 대상의 학습 결과를 평가하기 위해서 수업의 사전, 중간, 사후에 프로그래밍 교수내용지식을 평가하였다. 사전 평가는 연구 대상자가 프로그래밍 교육에 대해 어떤 지식을 가지고 있는지를 평가하기 위한 것이고, 중간, 사후 평가는

<표 3> 프로그래밍 교수내용지식 점수 변화 (N=12)

평가 시기	내용지식	교수법 지식	교육과정 지식	평가지식	평균
사전	2.87 (.668)	2.64 (.714)	2.32 (.716)	2.5 (.826)	2.63 (.634)
중간	2.63 (.611)	2.97 (.492)	2.53 (.764)	2.83 (.718)	2.9 (.544)
사후	3.32 (.790)	3.33 (.599)	3 (.603)	3 (.707)	3.21 (.563)

<표 4> 반복측정 분산분석 결과 (N=12)

개체 내 효과검정	제 III 유형 제곱합	자유도	평균 제곱	F	유의확률
내용지식	2.952	2	1.476	7.041	.004*
교수법지식	2.895	2	1.448	15.4	.000*
교육과정 지식	2.927	2	1.463	10.385	.001*
평가지식	1.556	2	.778	3.850	.037*
평균	2.074	2	1.037	16.202	.000*

*p > .05

연구 대상자의 변화를 살펴보기 위한 것이다. 평가의 시기가 두 번 이상이기 때문에 연구 대상의 변화가 유의미한지 알아보기 위하여 반복측정 분산분석을 실시하였고, 각 과정별 변화를 알아보기 위하여 사전 - 중간, 중간 - 사후 평가 결과에 대해 각각 대응표본 t 검정을 실시하여 결과를 분석하였다.

4.1 효과성 검정

예비정보교사의 프로그래밍 교수내용지식을 반복측정 분산분석을 통해 검정한 결과는 <표 3>, <표 4>와 같다. 모든 영역의 구형성 검정에서 구형성을 만족하였으므로 프로그래밍 교수내용지식 점수의 변화는 구형성을 가정한 통계값을 사용하였다.

내용지식 영역의 변화를 살펴보면, 사전검사에서 2.87이었던 점수가 중간검사에서는 2.63, 사후검사에서는 3.32로 나타났다. 세 번의 검사에서 내용지식영역의 점수는 다른 영역들의 점수보다는 높은 편인데, 이는 기본적으로 예비교사를 대상으로 한 프로그래밍 교육이 내용지식의 향상을

목적으로 하기 때문인 것으로 판단된다. 내용지식 영역 점수 변화의 유의확률은 .004로 $p > .05$ 범위를 만족하기 때문에 예비정보교사의 프로그래밍 내용지식이 유의하게 향상되었음을 확인할 수 있었다.

사전검사에서 나타난 교수법지식 영역의 점수는 2.63이었고, 중간검사에서는 2.97, 사후검사에서는 3.33으로 나타났다. 교수법지식 영역의 변화도 유의확률이 .000로 $p > .05$ 범위를 만족하기 때문에, 예비정보교사의 교수법지식 영역에도 유의한 변화가 있는 것으로 나타났다. 사후 평가와 함께 실시한 중등 프로그래밍 교육 설계에 대한 질문에서 많은 학생들이 문제 기반 학습 방법과 프로젝트 기반 학습 방법을 활용한 초중등 교육과정을 설계하는 답변을 서술한 것으로 볼 때, 교육프로그램을 수강한 학생들이 자신이 겪은 형태의 교수법을 실제로 교육에 활용하고자 하는 경향을 알 수 있었다.

사전검사에서 나타난 교육과정영역에 대한 점수는 2.32였고, 중간검사에서는 2.53, 사후검사에서는 3.0으로 나타났다. 교육과정지식영역의 변화는 유의확률이 .001로 $p > .05$ 범위를 만족하기 때문에 유의한 변화가 있는 것으로 나타났다. 본 교육프로그램에는 프로그래밍 교육과정에 직접적으로 관련된 내용이 포함되지 않았지만, 학습자들은 단계별로 진행되는 프로그래밍 교육내용을 학습하면서 개인적으로 프로그래밍의 교육 단계에 대한 지식을 쌓은 것으로 판단된다.

평가영역지식의 사전검사 점수는 2.5였고, 중간검사에서는 2.83, 사후검사에서는 3.0으로 나타났다. 평가지식 영역의 변화는 유의확률이 .037로 $p > .05$ 범위를 만족하기 때문에 유의한 변화가 있

는 것으로 나타났다. 문제 기반 학습과 프로젝트 기반 학습의 경우 결과물을 공유하고 평가하는 과정을 거친다. 본 프로그램에서도 학습자들이 앱을 만들기 위해 작성한 계획서, 최종적인 결과물을 공개하고 학습자끼리 평가하는 과정을 거치며 실제 프로그래밍의 결과물을 평가하는 지식을 쌓을 수 있었던 것으로 판단된다.

사전검사의 평균점수는 2.63, 중간검사는 2.9, 사후검사 평균점수는 3.21로 나타났다. 검사의 유의확률은 .000으로 $p > .05$ 범위를 만족하기 때문에 예비교사의 프로그래밍 교수내용지식에는 유의한 변화가 있는 것으로 나타났다. 결과적으로 설계한 교육 프로그램을 통해 예비정보교사의 프로그래밍 교수내용지식이 유의하게 향상되었음을 확인할 수 있었다. 많은 학생들이 사전 검사에서는 프로그래밍을 어떻게 교육해야 할지에 대한 자신을 충분히 가지지 못했던 것과 비교하여, 교육 프로그램을 완료한 후에 실시한 논의와 설문에서는 학습한 프로그래밍 언어를 활용하여 학습자에게 교육하는 교육 방식에 대해 구체적으로 서술하는 것을 확인할 수 있었다.

4.2 대응표본 t 검정 결과

4.2.1 사전 - 중간검사 비교

연구 대상자의 프로그래밍 교수내용지식이 문제 기반 학습 형태의 수업을 통해 어떻게 변화하였는지를 확인하기 위해 사전검사와 중간검사의 결과를 대응표본 t 검정하였다. 검정 결과는 <표 5>와 같다.

사전검사와 중간검사의 비교에서는 전체 평균 점수와 교수법지식에 대한 점수가 유의하게 변화

<표 5> 사전 - 중간 평가 대응표본 t 검정 (N=12)

영역	대응차					t	자유도	유의확률 (양쪽)
	평균	표준편차	평균의 표준오차	차이의 95% 신뢰구간				
				하한	상한			
내용지식영역	.238	.674	.194	-.190	.666	1.224	11	.246
교수법영역	-.333	.492	.142	-.646	-.020	-2.345	11	.039*
교육과정영역	-.217	.612	.177	-.605	.172	-1.227	11	.245
평가영역	-.333	.685	.198	-.769	.102	-1.685	11	.120
평균점수	-.275	.376	.109	-.514	-.036	-2.532	11	.028*

* $p > .05$

한 것으로 나타났다. 중간검사에서 나타난 특이한 점은 프로그래밍 내용지식에 대한 점수가 사전검사보다 하락한 점이다. 학생 설문에서 ‘너무 어렵다. 수준을 낮추는 것이 좋을 것 같다’, ‘블록에 대한 설명을 자세히 진행한 다음 프로그램 제작을 할 수 있으면 좋겠다’, ‘후반으로 갈수록 난이도를 따라잡기 어렵다’는 의견이 나타난 것으로 보아 내용지식 점수의 하락은 교육 프로그램의 난이도 구성에 따른 인지적 부담 때문인 것으로 판단된다. 이 외의 영역에서는 유의하지는 않지만 전반적으로 점수가 향상되는 것을 확인할 수 있다.

4.2.2 중간 - 사후검사 비교

연구 대상자의 프로그래밍 교수내용지식이 프로젝트 기반 학습을 통해 어떻게 변화하였는지 확인하기 위하여 중간검사와 사후검사의 결과를 대응표본 t 검정을 통해 비교하였다. 결과는 <표 6>과 같다.

중간검사와 사후검사의 비교에서는 평가영역의 점수를 제외하고는 모두 유의하게 변화한 것으로 나타났다. 이의 원인은 프로젝트 학습을 통해 모듈끼리 프로그램을 실제로 만들어가는 과정을 거치고, 결과물을 서로 평가하며 피드백하는 과정을 통해 내용지식과 교수법지식, 교육과정지식이 향상된 것으로 판단된다.

4.2.3 학습 관찰 및 설문 분석

중간검사와 함께 실시한 설문에서 학생들은 ‘문제 해결 수업 방식이 알고리즘을 생각하는 능력을 길러주고, 프로그래밍이라는 개념을 익숙하게 만든다’, ‘결과물이 빠르게 나타나서 피드백이 용

이하다’, ‘학생 간의 수준차이를 극복할 실습 예제가 필요하다’, ‘각자 문제에 대해 생각할 시간을 주고, 그것을 확인하는 과정에서 학습할 수 있는 것 같다’는 답변을 하였다. 이러한 답변으로 보아 학생들이 막연히 교수법 지식이 향상되었다고 느끼는 것이 아니라, 실제로 교육할 때 어떠한 장점과 단점이 있는지를 구체적으로 인식하고 있다고 유추할 수 있다.

프로젝트 학습 과정에서 예비교사들은 프로그래밍 과정에 대한 다양한 형태의 피드백을 받는 것을 관찰하였다. 프로젝트 계획 단계와 프로젝트 결과물 단계에서는 공식적으로 학생간의 피드백이 이루어졌고, 프로젝트 제작 단계에서는 모듈 내 의견교환, 모듈 간 프로젝트 구현에 대한 피드백 등을 통해 교수자, 학생 간 다양한 형태의 상호작용이 이루어졌다. 이러한 상호작용은 프로그래밍 교육의 측면에서 내용 지식이나 교수법에 대한 이해를 높이는 데 상당한 역할을 담당한 것으로 보인다.

사후 설문과 함께 진행한 중·고등학생에 대한 프로그래밍 교육 방법 질문에서 학생들은 다양한 형태의 답변을 내놓았다. 몇몇 학생들은 본 교육 프로그램에서 학습한 문제 중심 교수법이나 프로젝트 중심 교수법을 활용하여 중학생과 고등학생에게 프로그래밍을 교육하고자 하였고, 몇몇 학생은 자신이 수업을 경험하면서 부족한 부분을 중심으로 교수방법을 재구성하는 교육계획을 작성하였다. 이러한 답변들로 미루어 보아 학생들이 실제로 프로그래밍을 교육할 때 활용할 수 있는 지식이 갖추어졌음을 파악할 수 있었다.

<표 6> 중간 - 사후 평가 대응표본 t 검정 (N=12)

영역	대응차					t	자유도	유의확률 (양쪽)
	평균	표준편차	평균의 표준오차	차이의 95% 신뢰구간				
				하한	상한			
내용지식영역	-.690	.718	.207	-1.146	-.234	-3.333	11	.007*
교수법영역	-.361	.316	.091	-.562	-.160	-3.952	11	.002*
교육과정영역	-.467	.462	.133	-.760	-.173	-3.500	11	.005*
평가영역	-.167	.537	.155	-.508	.174	-1.076	11	.305
평균점수	-.313	.268	.077	-.482	-.142	-4.038	11	.002*

*p > .05

5. 결론 및 제언

본 연구는 프로그래밍 교수내용지식 향상을 위하여 교육용 프로그래밍 언어인 앱 인벤터와 문제 기반 학습, 프로젝트 기반 학습을 결합한 프로그래밍 교육 프로그램을 설계하고 이를 예비정보교사에게 적용한 연구이다. 사전 검사에서 예비정보교사의 프로그래밍 교수내용지식은 평균적으로 낮은 것으로 나타났다. 문제 기반 학습 방식으로 5주간 20시간 분량의 강의를 진행하고 중간 평가를 실시한 결과, 평균 점수와 교수법 영역의 점수가 유의미하게 상승하였다. 프로젝트 기반 학습 방식으로 5주간 20시간 분량의 강의를 추가적으로 진행하였을 때, 평가영역의 점수를 제외한 모든 분야의 점수가 유의미하게 향상된 것을 확인할 수 있었다. 최종적으로 반복측정 분산분석을 실시한 결과 연구 대상의 프로그래밍 교수내용지식의 향상이 모든 분야에서 유의미하게 나타났다. 단순히 예비교사의 교수내용지식점수만 향상된 것이 아니라 실제로 중등학교에서 학생들에게 프로그래밍 교육을 진행하기 위한 기본적인 지식이 형성되었음을 확인할 수 있었다.

본 연구는 예비정보교사의 프로그래밍 능력을 향상시키기 위한 교육프로그램을 설계하고 적용한 연구이다. 해당 교육프로그램의 효과를 확인하기 위하여 프로그래밍 교수내용지식에 대한 평가를 진행하였지만 적절한 비교집단이 형성되지 않아 단일집단 효과만을 판별하였다. 또한 실험집단의 표본수가 적어 연구결과를 일반화하기에는 어려움이 있다. 추가적인 연구를 통해 좀 더 큰 집단에서 다양한 형태의 프로그래밍 언어를 활용하였을 때 예비교사의 교수내용지식의 변화를 비교하여 더욱 객관적으로 앱 인벤터 교육의 효과를 검증할 필요성이 있다. 또한 교수내용지식은 자기평가 형태의 평가지로 교육과정이나 교육 내용이 변화하거나, 교육의 도구가 달라지거나, 시간에 따라 변화하는 특징을 보인다. 따라서 장기적인 관점에서 프로그래밍 교수내용지식을 평가할 수 있는 객관적인 평가지의 개발 및 타당성 평가가 이루어져야 하고, 예비정보교사와 정보교사가 프로그래밍 교육에 대한 교수내용지식을 충분히 발달·유지시킬 수 있는 다양한 교사교육프로그램이 구성되고 제공되어야 할 것으로 판단된다.

참 고 문 헌

- [1] Resnick, M., Maloney, J., Monroy-Hernandez, A., Rusk, N., Eastmond, E., Brennan, K. Millner, A., Rosenbaum, E., Silver, J. Silverman, B. & Kafai, Y. (2009). Scratch: Programming for all. *Communications of the ACM*, 52(11), 60-67.
- [2] 최정원, 이영준 (2014). 프로그래밍 학습에서 학습자의 어려움 분석. *컴퓨터교육학회 논문지*, 17(5), 89-98.
- [3] Robins, A., Rountree, J., & Rountree, N. (2003). Learning and teaching programming: A review and discussion. *Computer Science Education*, 13(2), 137-172.
- [4] 소경희 (2003). '교사 전문성'의 재개념화 방향 탐색을 위한 기초연구. *교육학연구*, 21(4), 77-96.
- [5] Grossman, P. L. (1990). *The making of a teacher: Teacher knowledge and teacher education*. Teachers College Press, Teachers College, Columbia University.
- [6] Abell, S. K. (2007). *Research on science teacher knowledge*. In S. K. Abell and N. G. Lederman (Eds.), *Handbook of Research on Science Education*. Mahwah, NJ: Lawrence Erlbaum Associates Publishers.
- [7] Magnusson, S., Krajcik, J., and Borko, H. (1999). *Nature, sources, and development of pedagogical content knowledge for science teaching*. In J.Gess-Newsome and N. G. Lederman (Eds.), *Examining pedagogical content knowledge*, Boston: Kluwer Academic Publishers.
- [8] 박준기 (2011). *전문성 발달 단계를 반영한 교사자격기준 개발 및 타당성 분석*. 박사학위 논문, 단국대학교 대학원.
- [9] Jaworski, B., & Gellert, U. (2003). Educating new mathematics teachers: Integrating theory and practice, and the roles of practising teachers. *In Second*

international handbook of mathematics education, 829-875, Springer Netherlands.

- [10] 안상진, 이영준 (2014). 앱 인벤터를 활용한 초,중등 프로그래밍 교육 방안. **컴퓨터교육학회 논문지**, 17(5), 79-88.
- [11] 전성균, 이영준 (2015). LT 협동학습 기반의 앱 인벤터 프로그래밍 교육이 초등학생들의 학습 동기에 미치는 영향. **컴퓨터교육학회 논문지**, 18(2), 1-9.
- [12] Gao, M., Johnson, J., Reed, D., Sheller, C., & Turbak, F. (2015). Using App Inventor in Introductory CS Courses. *In Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, 346-347.
- [13] Soares, A., & Martin, N. L. (2015). Teaching Non-Beginner Programmers with App Inventor: Survey Results and Implications. *Information Systems Education Journal*, 13(5), 24.
- [14] 박성혜 (2003). 교사들의 과학 교과교육학 지식 측정도구 개발. **한국교원교육연구**, 20(1), 105-134.
- [15] 김자미, 윤일규, 이원규 (2010). 정보교과 교수내용지식(PCK) 수준 측정 문항 개발 및 타당화. **컴퓨터교육학회 논문지**, 13(6), 23-34.
- [16] O'Grady, M. J. (2012). Practical problem-based learning in computing education. *ACM Transactions on Computing Education (TOCE)*, 12(3), 10.
- [17] Blikstein, P. (2011). Using learning analytics to assess students' behavior in open-ended programming tasks. *In Proceedings of the 1st international conference on learning analytics and knowledge*, 110-116.



안 상 진

2002 한국교원대학교
영어교육과(교육학학사)
2012 한국교원대학교
컴퓨터교육과(교육학석사)

2013~현재 한국교원대학교 컴퓨터교육과
박사과정

관심분야: 프로그래밍 교육, 교육용 프로그래밍
언어, 정보교육과정

E-Mail: ahnsang0@nate.com



이 영 준

1988 고려대학교
전산학과(이학사)
1994 미국 미네소타대학교
(전산학 Ph.D.)

현재 한국교원대학교 컴퓨터교육과 교수

관심분야: 정보통신교육, 지능형시스템, 학습과학

E-Mail: yjlee@knue.ac.kr