

코딩수업을 위한 학습자 중심의 플립드 러닝 적용 사례 연구

A Study on a Case Applying Learner-Centered Flipped Learning for Coding Classes

이 애 리*

가톨릭관동대학교 VERUM교양대학 교양과

Ae-ri Lee*

Department of Liberal Arts, Catholic Kwandong University, Gangneung 25601, Korea

[요약]

본 연구는 대학의 교양수업인 코딩수업에 플립드 러닝을 적용한 사례 연구이다. 컴퓨터를 전공하지 않는 학생들이 배우는 교양 코딩 수업은 전문가 양성을 위한 코딩 교육과는 차별화된 교육방법이 필요하다. 본 연구에서는 교양 수업으로의 코딩교육을 위한 플립드 러닝 수업 모형을 제안하고 그 효과 및 가능성을 살펴보았다. 학습자는 언제 어디서나 원하는 시간, 장소에서 온라인 콘텐츠로 학습을 하고, 수업시간에는 학습된 내용에 근거한 실습교육에 집중하는 플립드 러닝을 통해 교수 학습의 효율을 극대화하는 방안을 모색하였다. 실제 수업에 적용 후 사전·사후 설문을 실시하여 플립드 러닝 방식의 수업을 진행한 학생들이 전통적인 방식으로 수업을 진행한 학생들보다 학습만족도에 대해 긍정적으로 평가하였고 학습 성취도에 대한 결과도 더 나은 효과가 있음을 확인하였다.

[Abstract]

This is a study on a case applying flipped learning to coding classes that is a college liberal arts course. A required coding class for the students who do not major in computers needs a teaching method differentiated from a coding education for training experts. The present study presented a flipped learning teaching model for the coding education of non-major students, and observed its effect and possibility. Flipped learning enables learners to learn with on-line contents anywhere and anytime they want and concentrate on practice education based on what they learned during class. Accordingly, the study sought for the solution to maximize the efficiency of teaching and learning through flipped learning. A pre and post surveys after applying a flipped learning to a practical class confirmed that the students taught using flipped learning were more positively assessed in learning satisfaction than those taught using a traditional method, and that in academic achievement as well, flipped learning was more effective.

Key Words: Coding education, Flipped learning, Instructional design model, Programming

<http://dx.doi.org/10.14702/JPEE.2017.023>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 24 April 2017; Revised 23 May 2017

Accepted 24 May 2017

*Corresponding Author

E-mail: allee@cku.ac.kr

I. 서론

현재 디지털 기술이 다양한 영역과 융합되면서 4차 산업 혁명으로 불리는 급격한 기술 변화가 진행되고 있다. 4차 산업 혁명 시대를 대변하는 모든 것이 ITC기술을 바탕으로 한 소프트웨어를 통해 구현되기 때문에 코딩 교육이 관심의 대상이 되고 있다[1,2].

코딩 교육은 컴퓨터의 활용법을 배우는 것이 목적이 아니라 프로그래밍 언어를 통해 사물이나 현상을 논리적으로 분석해 문제점을 발견하고 해결하는 과정에서 창의적 사고력, 논리적 사고력, 문제해결력 등을 키우는 것을 목적으로 한다 [3]. 또한 코딩 교육은 과학기술에 대한 인문·사회학적 접근이나 IT기술을 활용하여 인문 지식과 아이디어를 구현하는 하는 등 학문 간의 융합을 할 수 있는 바탕이 되므로 대학에서의 코딩 교육은 매우 중요하다고 할 수 있다.

그러나 지금까지의 코딩 교육은 전문가 양성을 목적으로 한 경우가 많았기 때문에 보편적 교육으로서의 코딩 교육에 대한 면밀한 연구가 필요한 시점이다. 코딩 수업을 진행해보면 코딩을 처음 접하는 학생들은 비교적 쉬운 교육용 언어를 사용하는 수업에서도 중도에 포기하거나 수업의 어려움을 호소하는 경우가 많이 나타난다[4].

또한 기존의 강의 식 수업 방식에서는 프로그래밍 언어의 문법을 설명하는데 많은 시간을 할애해야 하고 실습에 필요한 지식도 설명해야 하므로 배경 지식을 모두 배우고 나면 학생들이 실제 직접 코딩을 해 볼 시간이 턱없이 부족하다는 문제점이 있다. 수업에서 설명해야 하는 내용은 사전에 제공 되는 동영상으로 학생 스스로 예습하고 수업시간은 실습에 활용하는 플립드 러닝방식을 적용하면 좀 더 체계적인 수업으로 운영되고 학생들의 능동적인 수업 참여를 도와 이러한 문제점을 해결 할 수 있다.

교양에서의 코딩 교육은 프로그래밍 언어의 기능을 습득 하는 것에 중점을 두지 않고 자신의 생각이나 다른 수업에서 획득한 지식을 정리하고 구체화 시키는 방법을 연습하고 이를 프로그래밍 언어라는 도구를 통해 실체화 시키고, 학생들의 특성을 고려하여 창의력, 문제해결력 신장을 위한 코딩 교육을 진행하도록 교과목을 설계가 이루어져야만 하는데 플립드 러닝 수업방식을 적용한다면 이것이 더욱 가능해질 것이다. 본 연구는 대학 1학년을 대상으로 한 교양 코딩 과목에 대한 플립드 러닝 모형을 제안하고 그 효과에 대해 논의 하고자 한다. 본 연구로 컴퓨터에 대한 지식이 없는 학생들에게 플립드 러닝 방식을 통한 코딩 교육서비스를 제공함으로써 학습자 중심의 효과적인 코딩 수업의 활성화에 도움이 될 것을 기대한다.

II. 이론적 배경

A. 플립드 러닝의 특징

플립드 러닝은 거꾸로 뒤집은 교육과정이란 뜻으로 전통적인 수업 방식은 강의 중심으로 수업을 한 후에 그 내용을 바탕으로 숙제를 해오는 방식이었다면 플립드 러닝은 학생들이 수업 내용을 미리 동영상을 통해 학습하고, 수업시간에는 그 내용을 토대로 토론을 하거나 과제 수행 활동 중심으로 교육하는 것을 말한다[5].

학습자들이 수업 전에 교실 밖에서 온라인을 통해 교수자가 제공하는 동영상 등의 학습 자료를 통해 자기 주도적으로 학습하고, 수업에서는 관련되는 과제를 연습하거나 심화 문제 풀이 등의 활동을 통해 지식을 적용하는 방식으로 이루어진다. 이때 수업에서 교수자는 지식 전달자가 아니라, 학습자들의 지식 적용 과정에서 문제 풀이가 잘 안 되거나 과제 연습이 어려운 이들을 돕는 조력자 역할을 한다[6].

플립드 러닝은 학습자의 능동적이고 자율적인 학습경험을 중요하게 생각하고 학습동기를 중요시 여기는 교육환경의 변화로 인해 관심이 고조되고 있다. 학습자는 각각의 개별 학습을 스스로 점검하고 자신의 능력에 맞춰 학습내용을 반복하여 학습할 수 있으며, 수업에 참여하지 않고 뒤쳐져 있거나 이미 알고 있는 내용을 반복하여 배울 필요가 없으며 그로 인해 수업에 대한 지루함을 느낄 필요가 없게 된다. 뿐만 아니라 학습자 각자 선행 학습한 내용을 바탕으로 강의실 내에서 부족한 부분에 대한 보충학습을 할 수 있으며 보다 관심이 가고 흥미를 가지는 부분에 대해서는 심화학습을 할 수도 있기 때문에 학습자의 자발적인 학습을 유발한다[7]. 이로 인해 플립드 러닝은 개인차이가 나는 학생들의 수준을 고려한 수업이 가능할 수 있다.

또한 플립드 러닝은 수업 전에 학생 스스로 학습을 해오는 자기주도적 학습 태도가 요구된다. 전통적 수업방식에서 수업 시간의 대부분이 교수자가 일방적으로 학습 내용을 전달하는데 사용되지만 플립드 러닝은 자기주도적인 학생 중심의 수업 활동을 통해서 교수자와 학습자의 상호작용을 하는데 더 많은 시간을 할애할 수 있다.

III. 플립드 러닝 수업 운영 사례

A. 플립드 러닝 적용 교과목 개요

본 연구는 C대학교 1학년 대상의 교양필수 교과목인 ‘창

의적 사고와 코딩'을 수강하는 학생들을 대상으로 진행되었다. 2016년 1학기(76명) 수업은 전통적인 수업방식으로 운영하였고, 2016년 2학기(93명)의 수업은 플립드 러닝 방식으로 운영하였다. 두 학기 모두 컴퓨터를 전공하지 않는 학생들을 대상으로 하였다.

플립드 러닝의 동영상은 집에서의 학습 후 실제 수업시간에 이루어지는 실습이나 심화 문제 등을 효과적으로 해결하는데 필요한 기초적인 개념들에 대한 내용을 체계적이고 간결하게 담아야 하며 미리 제공되는 수업 동영상은 15분 이내의 짧은 강의 내용을 담아야 한다. 사전 학습을 위해 제공되는 동영상은 프로그래밍 언어의 기능과 문법적인 부분 등의 개념을 설명하는 부분을 동영상으로 제공하였다, 선행학습을 위해 모바일 환경도 지원되는 대학의 LMS시스템에 동영상을 업로드하여 학생들에게 제공되었다. 매주 10분 분량의 동영상 2개 이내로 제공되었다.

이 과목은 비 전공 학생들의 코딩 수업이기 때문에 학생들은 프로그래밍 언어에 대한 지식이 전무한 상태이다. 이러한 비 전공 학생들을 대상으로 하는 코딩 교육은 언어 자체를 배우는 데 너무 많은 시간을 소요하거나 기술 전수에 집중되지 않아야 한다[8].

본 연구에서는 코딩 수업을 위해 명령어가 시각적으로 표현되어 문법 이해에 대한 부담이 적어 초보자도 이해하고 배우기 쉬운 교육용 프로그래밍 언어인 스크래치를 활용하여 수업을 진행하였다. 스크래치 같은 교육용 프로그래밍 언어는 코딩 교양 수업에서 프로그래밍을 처음 접하는 학생들

이 코딩 교육에 대해 느끼는 어려움을 최소화하고 코딩 교육에 대한 긍정적 감정을 형성할 수 있도록 하고 이를 통해 본인이 생각하는 바를 어려움 없이 디지털 기술을 통해 표현해 볼 수 있게 한다[9].

B. 수업 모델 설계

본 연구에서는 텍사스대학 교수학습센터의 플립드 러닝 기본구조[10]에 기초하여 수업을 설계하였다. 코딩교육의 특성에 맞게 15주 수업을 3개의 모듈로 나누어 조직하였다, 프로그래밍 언어의 기본 문법과 기능을 배우는 2주~7주까지의 기초모듈, 기초 모듈에서 학습한 기초 문법들을 통해 스스로 프로그램을 계획하고 작성해보는 9주~11주까지의 발전모듈, 팀 프로젝트가 이루어지는 12주~14주까지의 심화모듈의 순서로 15주 동안 수업이 진행된다. 이 중 기초 모듈과 발전 모듈에 플립드 러닝 수업이 적용되며 이때의 수업은 Pre-Class 와 3단계의 In-Class 그리고 After-Class 활동으로 이루어진다. 기초 모듈과 발전 모듈의 In-Class는 수업 내용의 특성에 따라 약간 다르게 적용되며 아래의 그림 1과 같이 플립드 러닝 수업 모델을 제시하였다.

C. 플립드 러닝을 적용한 수업진행

전체 수업 15주 수업 중 첫 주와 중간/기말고사를 제외하고 12주의 수업을 기초, 발전, 심화의 세 모듈로 그룹 화하여

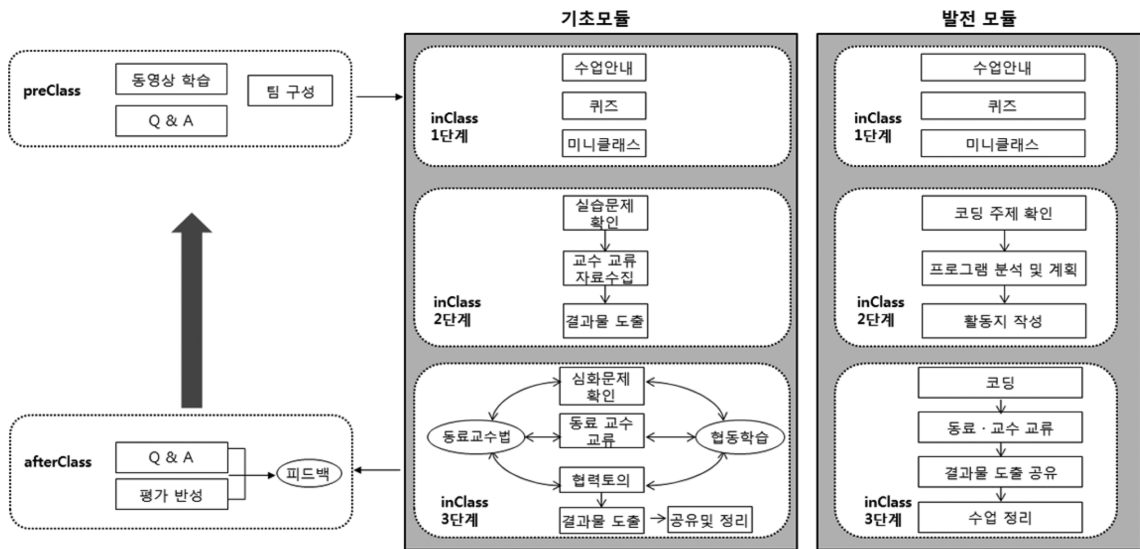


그림 1. 플립드 러닝 수업 모형의 핵심 활동
Fig. 1. Key activities of the flip-learning instructional model.

표 1. 모듈 별 수업 운영

Table 1. Classroom management by module

학습 단위	주차	플립드 러닝 적용	내용	In-class 운영 방법
기초	2~7	○	기본적인 프로그래밍 문법	시연-모방-제작
발전	9~11	○	개별프로젝트	탐구-설계-개발
심화	12~14		팀 프로젝트	팀 프로젝트

각 모듈 별 약간 상이한 방식으로 수업을 설계하여 적용하였다. 심화단계는 팀 프로젝트로 운영되기 때문에 플립드 러닝을 적용하지 않았다. 첫 주는 수업에 대한 오리엔테이션과 사전진단 평가를 실시하고, 마지막 주에는 동일한 질문으로 사후 진단평가를 실시하였다. 기초 모듈과 발전 모듈의 Pre-Class와 After-Class는 모두 동일하게 적용되며 In-Class의 경우 각 모듈 별로 다른 수업 운영 방식을 적용하며 표 1은 이를 나타낸다.

1) 기초 모듈(2주~7주의 수업 운영)

기초 모듈은 스크래치의 블록과 프로그래밍 언어적인 요소 등 기본 개념에 대해 익힌다. 교수자의 시연을 따라 코딩해 본 후 학생들은 이를 더 확장하여 새로운 결과물을 만들어 내게 된다. 플립드 러닝 수업 모델이 적용되며 수업은 Pre-Class와 3단계의 In-Class 그리고 After-Class 활동으로 구성된다.

a) Pre-Class

학습자가 학습 콘텐츠에 따라 스스로 학습을 하는 선행학습 단계이다. 수업 전 학생들은 교수자가 동영상으로 제작하여 온라인에 업로드 시켜 놓은 학습 콘텐츠를 따라 가며 새로 배울 개념에 대해 학습을 한다. 이 단계에서는 수업 동영상 예습을 하고, 팀 활동이 필요할 경우는 팀을 구성하는 단계로 이루어진다. 학습 중 모르는 내용은 LMS시스템의 질의 응답 게시판에 올려 학생들은 쌍방향으로 질의와 응답을 지속적으로 주고받을 수 있도록 하였다. 수업 동영상은 대학의 LMS시스템에 업로드하며 LMS시스템은 스마트기기에서도 재생을 할 수 있는 스마트 러닝 환경을 지원한다. 기초 모듈에서 제공되는 동영상은 프로그래밍 언어의 기본 문법과 기능에 대한 내용으로 제공된다.

b) In-Class

선행 학습에 대해 확인하고 퀴즈를 통한 확인 학습과 소규모 강의를 통해 학습자들의 과정을 안내하고 프로그램을 작성하는 교실에서의 플립드 러닝의 수업 단계이다. 기초 모듈

에서는 프로그램의 문법과 기능에 대해 익힌다. 교수자의 설명과 시범을 통해 표준모델을 제시하고, 교수자의 시연내용을 학생들이 그대로 따라 실습하고 실습의 과정에서 질문을 통해 학습자들이 교수자의 시연을 모방하고 이전의 단계에서 배운 내용을 토대로 학생들이 직접 만들어보는 시연중심 모델[11]을 적용한다. 교수자는 피드백과 소규모 강의를 통해 학습자들의 과정을 안내하고 학습자들은 배워야 할 기능 (skills)을 실습하도록 한다. In-Class는 3단계로 구성된다.

- ① 1단계: 실제 수업을 준비하기 위한 단계로 수업안내, 퀴즈, 미니 클래스로 구성하였다. 전반적인 In-Class의 각 단계에서 수행하게 될 학습 활동들에 대해 소개하고 미리 학습된 내용에 대한 확인을 위해 Lms시스템에서 간단한 퀴즈를 진행하여 학생들의 학습 정도를 확인한다. 이후 동영상 내용에 기반 한 프로그래밍 예제를 교수자가 시연하는 간단한 미니 강의를 진행 한다. 미니클래스에서 교수자의 시연내용을 학생들이 그대로 따라 실습하고 실습의 과정에서 질문을 통해 학습자들이 교사들의 시연을 모방하는 과정을 겪게 된다.
- ② 2단계: 수업시간에 진행될 코딩 예제를 확인하고 교수자의 시연을 모방하여 이전에 배운 내용을 토대로 학생들이 직접 만들어 본다. 이를 위해 필요한 자료 정보 수집하고 교수와의 교류를 통해 개별적으로 프로그램 결과물을 도출한다.
- ③ 3단계: 3단계는 이전단계의 결과물을 확장하는 심화문제를 가지고 이루어진다. 학생들은 이전 단계의 문제를 코딩 한 후 확장된 심화문제를 확인하고 동료나 교수와 교류하여 결과를 도출한다. 심화문제의 수준에 따라 팀 활동이 진행되며 이 경우 팀 별로 자료 정보를 수집하고 협력 토의하여 최종 완성된 실습 결과물을 도출한다.

c) After-Class

이 단계에서는 수업을 끝난 후의 마지막 단계의 활동이 이루어지는 단계로 평가와 반성과 정리가 이루어진다. 평가의 주체는 교수자로서 각 활동의 결과와 과정에 대해 평가한다. 학생들이 커뮤니티에 올린 결과물에 대한 동료 및 교수의 피드백이 이루어진다.

2) 발전 모듈(9주~11주의 수업 운영)

발전 모듈에서는 기초 모듈에서 학습한 기초 문법들을 통해 스스로 프로그램을 계획하고 탐구하고 작성해보는 과정을 거치게 된다. 학생 스스로 개발하고자 하는 프로그램에 대한 기본적인 탐구과정과 함께 코딩을 위한 기초 설계의 과정을 거쳐 자신만의 프로그램을 완성하게 된다.

a) Pre-Class

기초 모듈과 마찬가지로 학습자가 주어진 동영상으로 선행학습을 하는 단계이다. 이 단계에서는 수업 동영상을 예습하고, 학습 중 모르는 내용은 LMS시스템의 질의응답 게시판에 올려 학생들은 쌍방향으로 질의와 응답을 지속적으로 주고받을 수 있도록 하였다. 다양한 알고리즘의 연습을 위한 예제 프로그램을 설명하는 내용으로 동영상에 제공된다. 발전 모듈에서는 팀 활동은 이루어지지 않고 개별적으로 프로그램을 계획하고 탐구하여 코딩을 한다.

b) In-Class

학생 스스로 개발하고자 하는 프로그램에 대한 기본적인 탐구과정과 함께 개발을 위한 기초 설계의 과정을 거쳐 자신만의 프로그램을 개발하게 된다. 1단계에서는 프로그램의 분석을 하는 등의 탐구(Discovery) 과정을 거치고 2단계는 설계(Design), 3단계에는 개발(Development)의 과정을 거쳐 학습자가 코딩의 전 과정을 주도할 수 있게 된다.

- ① 1단계: 실제 수업을 준비하기 위한 단계로 수업안내, 퀴즈, 미니 클래스로 구성하였다. 전반적인 In-Class의 각 단계에서 수행하게 될 학습 활동들에 대해 소개하고 미리 학습된 내용에 대한 확인을 위해 LMS시스템에서 간단한 퀴즈를 진행하여 학생들의 학습 정도를 확인한다. 작성할 프로그램의 주요 기능과 아웃라인을 설명하는 미니 클래스를 진행 한다. 학생들은 미니 클래스를 통해 설명된 스크립트를 분석하고 탐구한다.
- ② 2단계: 수업시간에 진행될 코딩 큰 주제를 확인하고 미니클래스에서 설명한 내용을 기본으로 프로그램의 스토리, 필요한 객체, 객체의 역할과 객체 간의 상호작용 등을 프로그램 작성을 위한 계획을 하고 설계 활동지를 작성한다.
- ③ 3단계: 이전 단계의 계획에 따라 실제 프로그램으로 구현한다. 이 과정에서 동료나 교수자와 교류하여 피드백하는 작업으로 결과물을 완성한다. 학생들은 결과물을 커뮤니티에 올리고 공유한다.

c) After-Class

이 단계에서는 수업을 끝난 후의 마지막 단계의 활동이 이루어지는 단계로 평가와 반성과 정리가 이루어진다. 평가의 주제는 교수자로서 각 활동의 결과와 과정에 대해 평가한다. 학생들이 커뮤니티에 올린 결과물에 대한 동료 및 교수의 피드백이 이루어진다.

3) 심화 모듈(12주~14주의 수업 운영)

심화 모듈에서는 팀 프로젝트가 이루어진다. 이때는 플립드 러닝을 위한 동영상은 제공되지 않고 수업시간 중에 학생들이 준비-기획-수행-정리의 단계를 거쳐 팀 프로젝트를 완성하게 된다.

IV. 플립드 러닝 적용 결과 분석

본 연구는 C대학교 교양필수 교과목인 ‘창의적 사고와 코딩’을 수강하는 1학년 학생들을 대상으로 선정하여 실시하였으며, 실험집단(93명)은 2016년 2학기에 플립드 러닝 방식으로 수업하였고, 통제집단(76명)은 2016년 1학기에 전통적인 수업방식으로 수업을 받도록 구성하였다. 본 연구에서 사용된 측정 도구는 학업성취도 평가를 위한 사전 진단평가, 사후 진단평가, 수업만족도 사후검사가 사용되었다. 연구대상의 플립드 러닝 적용 후의 효과를 알아보기 위해 통계프로그램인 SPSS 21을 사용하였다.

A. 학업 성취도 분석

두 집단의 학업 성취도 사전 검사를 독립표본 t검정을 이용하여 분석한 결과는 표 2와 같다. 그 결과를 살펴보면 사전 검사에서 $p < .05$ 수준에서 유의미한 차이가 있었다. 두 집단 간에 통계적으로 유의미한 차이가 있으므로 두 집단은 동질하지 않다. 실험을 진행한 학기와 대상 학과가 달라서 약간의 통계적인 차이가 있는 것으로 나타났다. 이에 사전 검사 점수를 공변인으로 통제하여 사후 검사에 대해 공분산 분석을 실시하여 두 집단은 사전 검사에서 동질적이 않은 결과를 조정하여 사후 검사를 분석하였다. 사전 검사는 통제집단($M = 21.53, SD = 7.34$)이 실험 집단($M = 21.53, SD = 7.34$)에 비해 높았으며 수업이 종료된 후 다시 측정한 사후 검사는 실험 집단($M = 39.33, SD = 4.80$)이 통제집단($M = 34.70, SD = 7.78$)보다 높은 점수를 받은 것으로 나타났다. 학업 성

표 2. 사전 학업 성취도 검사의 차이검증

Table 2. Verification of differences in prior academic achievement test

실험 집단(N = 93)		통제집단(N = 76)		t
평균	표준편차	평균	표준편차	
19.25	5.76	21.53	7.34	2.26

표 3. 사전 사후 학업 성취도 검사와 조정된 사후 검사의 평균과 표준편차

Table 3. Pre-post academic achievement test, the mean and standard deviation of adjusted post-test

	사전 학업성취도		사후 학업 성취도		조정된 사후 학업 성취도	
	평균	표준편차	평균	표준편차	평균	표준편차
실험 집단(N = 93)	19.25	5.76	39.33	4.80	39.32	0.67
통제집단(N = 76)	21.53	7.34	34.70	7.87	34.71	0.74

표 4. 학업성취도 진단평가에 대한 공분산 분석

Table 4. Analysis of covariance for assessment of academic achievement

변량원	제곱합	자유도	평균제곱	F
공변량(사전검사)	0.851	1	0.851	0.201
주 효과(집단)	861.74	1	861.74	21.10***
잔차	6777.64	166	40.83	
합계	7676.28	168		

취도의 사전 검사가 집단 간에 유의미한 차이가 있는 것으로 나타났으므로 공분산 분석을 실시하여 두 집단의 사전 검사를 이용해 조정 평균을 산출하였다. 실험 집단(M = 39.32, SD = 0.67)과 통제집단 (M = 34.71, SD = 0.74)의 평균이 조정이 되었다. 표 3은 사전 사후 학업 성취도 검사와 조정된 사후 검사의 평균과 표준편차를 나타낸다.

표 4에는 두 집단 간의 학업성취도 사후평가에 대한 공분산 분석을 실시한 결과를 나타낸다. 사전 검사를 공변인으로 통제한 후 사후 검사에 대해 분산분석을 실시한 결과, 주 효과는 F = 21.10으로 플립드 러닝을 수업방식을 적용한 실험 집단과 통제집단 간에 사후 검사에서 통계적으로 유의미한

차이가 있는 것으로 나타났다. 따라서 플립드 러닝을 코딩 수업에 적용함으로써 학업 성취도의 향상에 긍정적인 영향을 미친다는 것을 알 수 있다.

B. 수업 만족도 분석

플립드 러닝 수업이 진행된 학기가 끝난 후, 실험 집단 학생들의 수업에 대한 만족도를 파악하기 위해 리커트 5점 척도의 설문지를 자체 제작하였다. 해당 설문지는 이지연 외 [10]와 최유나[12]에서 사용한 수업 만족도 설문지를 본 연구의 주제에 맞게 적절히 재구성하였다. 표 5에는 수업 만족도의 설문 문항과 각각에 대한 응답 비율을 나타낸다. 표 5에서 플립드 러닝 학습에 만족한다는 질문에 매우 그렇다가 12.5% 그렇다가 69.8%로 전체적으로 그렇다는 응답률이 높음을 알 수 있다. 대부분의 만족도 문항은 긍정 답변의 비율이 높는데 비해 1번 9번 문항은 긍정 답변과 평균이 다소 떨어진다. 그 이유는 학생들 중 일부는 미리 동영상 보고 선행 학습을 하고 오는데 대한 부담을 상당히 가지고 있기 때문으로 판단된다.

표 5. 수업 만족도 설문 문항 결과

Table 5. Class satisfaction questionnaire result

문항	매우 그렇다	그렇다	보통이다	그렇지 않다	매우 그렇지 않다	평균	표준 편차
1 수업 전에 동영상을 보고 온다.	13.5	46.9	27.1	7.3	5.2	3.56	0.99
2 교수님께서 올려주시는 동영상 강의가 코딩을 공부하는 데에 많은 도움을 주었다.	10.4	62.5	19.8	7.3		3.76	0.74
3 수업의 진행 순서가 효과적이라고 생각한다. (동영상 선행학습 → 수업 중 실습)	18.8	65.6	12.5	3.1		4.00	0.66
4 수업내용에 대하여 잘 모르는 사항이 있으면 쉽게 피드백을 받을 수 있었다.	11.5	75.0	9.4	3.1	1.0	3.93	0.65
5 플립드 러닝 학습을 통해 학습에 능동적으로 참여할 수 있게 되었다.	9.4	77.1	9.4	4.2		3.92	0.59
6 수업에 관한 큰 어려움이 없었다.	5.2	80.2	10.4	4.2		3.86	0.55
7 플립드 러닝 학습을 통해 코딩에 대한 관심이 증가하였다.	6.3	74.0	16.7	3.1		3.83	0.57
8 강의 식 수업보다 플립드 러닝이 프로그래밍 개념을 이해하는데 더 많은 도움이 된다.	24.0	56.3	16.7	3.1		4.01	0.73
9 다른 과목에서도 플립드 러닝을 했으면 좋겠다.	5.2	64.6	26.0	3.1	1.0	3.70	0.67
10 나는 수업 방법에 대해 전반적으로 만족한다.	12.5	69.8	13.5	4.2		3.91	0.65

V. 결론

본 논문은 대학 교양으로서의 코딩 교육을 위한 플립드 러닝을 모델을 설계하고 실제 수업에 적용 후 적용한 후 그 효과를 진단평가와 설문을 실시해 분석 제시하였다.

코딩 교육은 과목의 특성상 학생들 스스로 사고하며 직접 해볼 수 있는 실습이 상당히 중요하데, 전통적인 수업방식에서는 강의와 실습이 수업 안에 모두 끝나야 하므로 강의를 진행하다 보면 실습에 할애할 수 있는 시간이 상당히 부족하여 학생들이 충분한 학습이 되지 않고 수업 내용에 대한 충분한 연습을 할 수가 없었다. 그에 반해 플립드 러닝 방식은 수업에서 진행해야 하는 강의내용을 사전에 동영상으로 제작하여 이를 수업 전에 예습하고 10분정도의 미니클래스를 제외한 나머지 수업 시간을 모두 학생들의 실습에 사용할 수 있었다. 수업시간에는 간단한 정리와 실습을 진행하므로 실습에 할애할 수 있는 시간을 굉장히 많이 확보할 수 있어서 따로 과제를 제출하지 않아도 충분한 학습이 되고 복습이 되어 전통적인 수업방식으로 진행한 통제집단보다 플립드 러닝을 진행한 실험집단에서 좋은 성취도 결과를 나타냈다. 이는 플립드 러닝 수업방식에서 충분한 실습과 연습을 통해 프로그램 사용법에 자신감을 갖게 된 학생들의 자기평가도 상당히 긍정적으로 나타난 것이다.

플립드 러닝 방식은 스스로 사고하고 수업의 내용을 응용하여 결과물을 도출하는 코딩 과목의 특성에 아주 잘 맞는 수업 방식으로 학생들의 학습성과가 좋은 방식이라 할 수 있다. 그러나 플립드 러닝은 수업에 잘 참여 하는 학생들에게는 긍정적인 수업 방식이었지만 학습의욕이 떨어지는 학생들은 선행학습을 하지 않고 수업을 포기해 버리는 경우가 발생했다.

플립드 러닝방식의 수업을 진행 하면서 수업의 사전 구성 단계에서 수업을 포기하는 학생을 줄이기 위한 다양한 대책과 선행학습에 자발적으로 참여 할 수 있도록 학생들의 동기 유발을 할 수 있는 방법에 대한 연구가 필요할 것이다. 또한 향후 과제로 코딩 수업의 특성상 교양 수업에서 느낄 수 있는 코딩의 어려움과 문제점을 실증적으로 분석하고 이를 수업에 반영할 수 있는 연구가 필요할 것이다.

감사의 글

이 연구는 가톨릭관동대학교 학술연구비에 의해 지원되었음(CKURF-201604130001).

참고문헌

- [1] J. H. Lee, "Educational transformation in response to the fourth industrial revolution," *Hansun Foundation Advanced Policy Series*, pp. 158-189, 2017.
- [2] Y. R. Kim and Y. E. Moon, "Study on the use of public open data for software (SW) education," *The Journal of Internet Electronic Commerce Research*, vol. 15, no. 6, pp. 245-261, December, 2015.
- [3] D. S. Han, "University education and contents in the fourth industrial revolution," *Humanities Contents*, 42, pp. 9-24, September, 2016.
- [4] K. M. Kim and H. S. Kim, "A case study on necessity of computer programming for interdisciplinary education," *Journal of Digital Convergence*, vol. 12, no. 11, pp. 339-348, November, 2014.
- [5] J. Bergmann and A. Sams, "Flip your students' learning," *Educational Leadership*, vol. 70, no. 6, pp. 16-20, March, 2013.
- [6] B. Jonathan and S. Aaron, *Flip Your Classroom: Reach Every Student in Every Class Every Day*, USA: ISTE, 2012.
- [7] J. Bergmann, J. Overmyer, and B. Wilie, *The Flipped Class: Myths Versus Reality. The Daily Riff-Be Smarter. About Education* [Internet], Available: <http://www.thedailyriff.com/articles/the-flipped-class-conversation-689.php/>.
- [8] J. S. Sung, S. H. Kim, and H. C. Kim, "Analysis of art and humanity major learners' features in programming class," *The Journal of Korean Association of Computer Education*, vol. 18, no. 3, pp. 25-35, May, 2015.
- [9] M. Resnick, J. Maloney, A. Monroy-Hernández, N. Rusk, E. Eastmond, K. Brennan, A. Millner, E. Rosenbaum, J. Silver, B. Silverman, and Y. Kafai, "Scratch: Programming for All," *Communications of the ACM*, vol. 52, no. 11, pp. 60-67, November, 2009.
- [10] J. Y. Lee, Y. H. Kim, and Y. B. Kim, "A study on application of learner-centered flipped learning model," *Journal of Educational Technology*, vol. 30, no. 2, pp. 163-191, June, 2014.
- [11] J. S. Kim, "Development of SW education teaching and learning model," *Korean Education Development Institute, Research Report CR2015-35*, 2015.
- [12] Y. N. Choi, "The effects of flipped-learning on the self-directed learning, academic motivation, and class satisfaction of the college students," *Master Dissertation, Konkuk University, Seoul* 2016.



이 애 리 (Ae-ri Lee)

2007년 2월 : 명지대학교 컴퓨터공학과 박사

2007년 7월 ~ 2009년 8월 : CTS연구소 연구원

2011년 9월 ~ 2014년 2월 : 안동과학대학교 의료공학과 초빙교수

2014년 3월 ~ 현재 : 가톨릭관동대학교 VERUM교양대학 교양과 조교수

〈관심분야〉 네트워크 보안, IOT 보안, 코딩 교육