

프로그래밍 수업의 플립드러닝 학습모형 설계 및 적용

최숙영†

요 약

본 연구는 대학의 프로그래밍 수업을 위해 플립드러닝에 기반한 수업 모형을 설계하고 이를 적용한 내용이다. 프로그래밍 과정은 프로그래밍 언어의 문법에 대한 충분한 이해와 함께 높은 추상적인 사고를 요구하기 때문에 많은 학생들이 학습에 어려움을 갖게 된다. 이러한 프로그래밍 학습의 특징으로 인하여 한정된 수업시간에 프로그래밍 언어의 문법적인 요소들을 강의하고 예제를 따라하는 학습 방법으로는 프로그래밍 수업의 학습 목표를 효과적으로 달성할 수 없다. 이러한 문제점을 극복하고 효과적인 프로그래밍 수업이 이루어질 수 있도록 본 연구에서는 플립드러닝에 기반한 수업 모형을 설계하고 그에 따라 수업을 진행하였다. 특히, 본 연구에서는 프로그래밍 학습에서 발생하는 학습자들의 어려움과 실제 프로그래밍 과정에서 발생하는 오류들을 분석한 후 이를 기반으로 하여 수업 모형을 설계하였다. 이 모델을 실제 수업에 적용한 결과, 학생들은 교실 수업에서 다른 학생들과 많은 소통의 기회를 가졌고 학습의 참여도가 높아진 것에 대해 긍정적으로 생각하며 수업에 대해 대체로 만족하는 것으로 나타났다.

주제어 : 플립드러닝, 프로그래밍 학습, 교수학습모형

Design and Application of an Instructional Model for Flipped learning of Programming Class

Sook Young Choi†

ABSTRACT

The purpose of this study is to design and implement an instructional model for flipped learning of programming class. Because a programming process requires a high level of abstract thinking with a good understanding of a programming language, many students have difficulty in programming. For this characteristic of programming, it is not easy to effectively achieve learning goals of programming lesson by lecturing grammatical elements of a programming language and the following examples in a limited class time. In order to overcome these problems and to make effective programming lessons, we designed an instructional model based on flipped learning. Especially, in this study, we analyzed learners' difficulties in programming learning and errors that occurred in actual programming process for designing the instructional model. As a result of applying this model to the class, it was found that the students were generally satisfied with the lesson by having positive communication with other students in the classroom and actively participating in the learning.

Keyword : Flipped Learning, Programming Learning, Instructional Model

† 정 회 원: 우석대학교 정보보호안학과 교수
논문접수: 2017년 5월 31일, 심사완료: 2017년 7월 11일, 게재확정: 2017년 7월 28일

1. 서론

미래의 모든 산업이 컴퓨팅의 영향력 아래 놓이기 때문에 세계 각국에서는 프로그래밍의 중요성을 인지하고 국가 차원의 교육과정을 마련하여 프로그래밍 교육을 강화하고 있다[1][2]. 초중고뿐만 아니라 대학에서도 전공에 불문하고 신입생들에게 프로그래밍 수업을 듣도록 하고 있는 추세이다. 이와 같이 프로그래밍 교육이 강조되고 상황이지만 실제로 많은 학생들은 프로그래밍 학습을 어렵게 느끼고 있는 것이 현실이다.

프로그래밍 과정은 주어진 문제를 정확히 이해하여 문제 해결을 위한 전략을 세우고 이에 따라 알고리즘을 기술하고 기술된 알고리즘을 토대로 특정 언어의 문법에 맞게 번역하는 과정이 필요하다. 이에 따라, 프로그래밍 언어에 대한 충분한 문법적인 지식과 활용 능력이 요구되고, 문제를 해석하고 문제 해결 전략을 세우는 과정 등 고등사고력을 필요로 한다. 이러한 이유로 인하여 학생들이 프로그래밍 과정을 어렵게 느끼고 중도에 포기하는 결과가 발생한다[3].

기존의 프로그래밍 교육은 강의자 중심의 프로그래밍 언어의 문법적 요소의 강의나 그것을 사용하는 예제 프로그램 실행 등으로 이루어졌다. 이러한 수업은 학습자의 흥미를 떨어뜨릴 뿐 아니라 실제적인 프로그래밍 능력을 키울 수 없게 되는 문제점이 발생되었다. 더욱이 한정된 수업시간에 프로그래밍 언어의 문법적인 지식을 익히고 그것을 활용하여 다양한 응용 프로그램을 충분히 작성하는 것이 쉽지 않기 때문에 이를 학생들에게 과제로 제공하게 된다. 하지만 학습자 입장에서 학습내용에 대한 충분한 이해와 활용을 해보지 않은 상태에서 심화 과제를 해결하기란 결코 쉽지 않기 때문에 프로그래밍 학습에 대한 흥미를 점점 잃어가는 일이 발생하고 중도에 포기하는 학생들이 생기기 된다. 이러한 기존의 수업 방식의 문제점을 해결하기 위한 한 방안으로 고려될 수 있는 것이 플립드러닝(Flipped learning)이다. 플립드러닝은 학습자 의사소통의 증가와 배운 지식의 활성화 활동을 통해 학습자 중심의 수업을 구현하는 방법으로 기존 학습 방식을 뒤집는 학습을 의미한다[4][5]. 전통적인 학습방법에서 학

생들이 집에서 과제로 하던 응용 연습이나 심화 학습 등을 학교에서 하고 학교에서 하던 교사의 강의 듣는 것을 집에서 하는 방식이다[6]. 플립드러닝은 미국 고등학교 과학교사인 Bermamm과 Sams에 의해 처음 그 용어와 학습방식이 사용되어 이후 초등학교에서 대학에 이르기까지 다양한 교과와 과제에 확대 적용되고 있다[7]. 최근 몇 년 동안 국내에서도 플립드러닝에 대한 관심을 높아짐에 따라 이를 일선 학교에서 도입하여 수업을 진행하고 그 결과들을 많이 발표하고 있다. 플립드러닝이 효과적으로 이루어지기 위해서는 플립드러닝 수업이 이루어지는 교육환경을 고려하여 플립드러닝 수업의 구체적인 절차와 활동 내용을 포함하는 수업 모형을 개발하여 제공해야 한다[8].

본 연구에서는 프로그래밍 수업에 플립드러닝을 적용한 내용을 기술한다. 특히 플립드러닝을 위해 프로그래밍 학습에서 발생하는 학습자들의 어려움과 실제 프로그래밍 과정에서 발생하는 오류들을 분석한 후 이를 기반으로 하여 수업 모형을 설계하였다. 이 수업모형을 토대로 실제 수업을 설계하여 진행하고 학습자들의 반응을 분석하였다.

2. 관련연구

2.1 플립드러닝

플립드러닝은 교실안에서 전형적인 교사주도의 강의와 수업 후 가정학습의 형태인 과제를 수행하던 방식이 역으로 이루어지는 수업을 의미한다. 즉, 교실 수업 전에는 강의 동영상이나 수업자료를 온라인으로 학습자들에게 제공하여 학습하도록 하고, 교실 수업에서는 사전에 학습한 내용을 적용해보는 다양한 활동에 참여함으로써 학습자가 중심이 되어 과제나 문제를 해결하도록 하는 방식이다.

플립드러닝의 구성요소로 Hamdan과 그의 동료들[7]은 융통성 있는 학습환경, 학습자 중심 문화, 의도된 학습내용, 전문성을 갖춘 교육자를 제안하였다. 융통성 있는 학습 환경은 플립드러닝에서 다양한 학습방식이 가능하다는 것이며, 학습자 중심 문화는 기존의 수업에서는 교사가 중심이 된

수업이었지만 플립드러닝에서는 학습자가 중심이 되어 학습주제에 대한 깊이 있는 탐색을 가능하게 해준다. 의도된 학습내용은 교수자가 교실 전 수업에서 가르쳐야 할 내용을 계획적으로 선정하고 교실안 수업에서 학습자가 학습내용을 적용·심화 학습할 수 있도록 어떤 내용을 어떤 교수방법으로 진행할지를 고려하는 작업이다. 전문성을 갖춘 교육자는 플립드러닝에서 교수 역할이 사라지는 것이 아니라 전 수업과정에서 잘 훈련되고 전문성을 갖춘 교수자로서의 역할이 더 필요하다는 것이다. Chen과 그의 동료들[9]은 위와 같은 플립드러닝에 대한 구성요소에 진보적 네트워킹 활동, 참여적이고 효과적 학습 경험, 다각적이고 원활한 학습 플랫폼을 추가하여 플립드러닝 모형을 정의하였다. 이러한 3가지 요소들은 대학교육의 상황에서 플립드러닝을 실행하는데 적절한 요소로 고려된다[10]. 진보적 네트워킹 학습활동은 학습도구로서 소셜미디어와 온라인 네트워킹을 활용하는 것을 의미하고 참여적·효과적 학습 경험은 학습자들이 교수자와 동료학습자와 함께 수업활동을 참여하고 긍정적 인식을 유지할 수 있도록 다각적인 수업전략과 기술을 이용하여 효과적으로 학습이 이루어질 수 있도록 하는 것이다. 다각적이고 원활한 학습 플랫폼은 다양한 학습 환경에서 학습자의 학습이 원활하며 지속적으로 이루어질 수 있도록 하는 것이다.

이러한 플립드러닝에 대한 특징은 다음과 같이 정리될 수 있다. 먼저, 학습자의 책임과 주도성의 증가로 교수자중심의 수업에서 학습자 중심의 수업으로 전환이다. 교수자의 역할이 지식의 전달자가 아니라 학습의 안내자이자 조력자로서 강조된다. 또한 응용·심화 학습이 가능한 개별화된 수업의 성격을 갖는다. 수업전 학습을 위해 다양한 테크놀로지를 활용하며 교실 수업에서는 학습내용의 적용을 위한 다양한 수업 방법이 적용된다. 특히 교실 안에서 학습자들 간의 활발한 상호작용과 학습참여를 통해 고차원적인 사고를 키우고 문제해결력을 향상시키는 학습 방법이다.

2.2 프로그래밍 학습

최근 소프트웨어 교육에 대한 관심이 높아지면

서 프로그래밍 교육의 중요성에 대한 인식이 확산됨에 따라 대학에서도 신입생의 전공에 관련 없이 프로그래밍 교육을 실시하고 있는 추세이다. 하지만 학생들이 갖는 프로그래밍에 대한 일반적 인식은 어렵다고 생각하는 것이다. 실제로 프로그래밍 강좌들의 학습 탈락률이 타 과목에 비해 높은 편이다 [11].

프로그래밍의 과정은 하나의 문제해결과정으로 간주될 수 있다. Mayer는 프로그래밍을 어떤 특정 작업에 대한 자연언어의 표현을 컴퓨터 구조적 특성과 처리적 특성에 제한을 받는 형식 언어인 프로그래밍 언어로 번역하는 하나의 문제 해결 과정이라고 정의하고 있다[12]. 프로그래밍을 하나의 문제해결 과정으로서 고려할 때, 이를 해결하기 위해서는 문제해결의 각 단계에서 요구되는 인지적인 측면이 무엇인지 살펴보고, 그에 따른 적절한 수업 전략을 연구하는 것이 필요할 것이다. 프로그래밍 과정은 주어진 특정한 과제에 대한 자연언어 차원의 문제해석 단계, 컴퓨터의 구조적·처리적 특성에 대한 지식에 기반하여 문제해결 전략을 세우고, 알고리즘을 구현한 후 특정 프로그래밍 언어로 프로그램을 작성하는 단계, 작성된 프로그램을 실행시켜서 오류를 점검하고 수정하는 단계로 구성된다[13]. 따라서, 첫 번째 단계에서는 문제에 대한 정확한 이해와 사고가 중요하며, 두 번째 단계에서는 컴퓨터 구조와 처리 특성을 이해하여 프로그래밍 논리로 변환시키는 능력, 알고리즘을 표현하기 위한 논리적 사고, 특정언어에 대한 문법적 지식 등이 필요하다.

한편, 문제해결 과정에서 프로그래밍 전문가와 초보자간에는 차이가 존재한다. 이러한 차이를 분석하여 적절한 교수 전략을 세우는 것은 의미가 있다. Mayer는 전문가와 초보자의 차이를 분석하여 4가지 측면, 즉, 첫째, 프로그래밍 언어의 문법적 지식, 둘째, 컴퓨터 처리 시스템에 대한 유용한 심성(mental) 모형, 셋째 문제를 큰 구조적인 측면에서 바라보고 계획을 세우는 도식적 지식, 넷째, 세부 절차를 계획하는 전략적 지식의 측면으로 구분하여 비교하였다[14].

이것을 기초로 하여 초보자들이 자주 저지르는 오류를 다음과 같은 5가지로 구분하였다[12]. 먼저, 문제의 핵심을 파악하지 못해서 발생하는 문

제 해석의 오류이다. 둘째는 전략적 오류로 목표를 달성하기 위한 방법의 세부 절차를 계획하는 과정에서의 오류이다. 세 번째는 번역오류로 알고리즘을 프로그래밍 언어로 번역하는 과정에서의 오류이다. 네 번째는 문법적 오류로 해당 프로그래밍 언어에 대한 지식의 부족으로 인한 것이다. 마지막으로 코딩 오류로 타이핑이나 연산자 누락 등과 같은 실수에 의한 오류이다.

최현종도 초보 프로그래머의 특징을 국외 연구들을 조사하여 기술하고 있는데, 그 특징들을 구분해보면 다음과 같이 구분할 수 있다 [11]. 먼저, 프로그래밍 언어의 문법적 요소들을 정확히 이해하지 않은 경우이다. 둘째, 프로그램을 작성하는데 큰 관점의 구조적인 측면에서 접근하기보다 코드 한줄 한줄을 해석하는데 집착한다는 점이다. 셋째, 프로그램 전체를 재구조화하기보다 지엽적인 버그를 수정하는데 집착한다는 점이다. 넷째, 프로그램을 이해하는 심성 모델이 부족하다는 점이다. 다섯째 배운 프로그래밍 지식을 프로그램을 작성하는데 정확하게 적용하지 못한다는 점이다.

2.3 선행연구 분석

대학 수업에 플립드러닝을 도입하여 그 효과를 분석한 연구로 여러 연구들이 존재한다. 이 연구들의 경우 플립드러닝을 위해 각 수업의 특징에 맞는 수업 모형을 설계한 후 모형에 따라 수업을 진행한 후 그 효과를 분석한 것이 주를 이룬다.

이러한 연구로 김보경은 교직 수업에 플립드러닝을 적용하기 위하여 수업 설계 단계를 포함한 플립드러닝 수업 모형을 제안하고 있다[15]. 권오남과 그의 동료들은 수학교육에 플립드러닝 수업을 적용하여 교실안에서는 학습자 중심의 문제해결 활동을 수행하도록 하고 있다[16]. 또한 공학분야에 적용한 연구로 최정빈과 김은경은 'PARTNER'라는 공학분야 플립드러닝 모형을 제안하였다[17]. 정동섭은 스페인어교육에 플립드러닝을 적용하여 학생들의 말하기 능력이 향상된 것을 확인할 수 있었다[18].

컴퓨터 전공의 프로그래밍 수업에 플립드러닝을 적용한 수업으로는 김경미와 이강이 파이썬 프로그래밍 수업을 위해 사전 질문을 이용한 플

립드러닝을 설계하여 적용한 연구이다[19]. 유상미는 C 프로그래밍 수업에 플립드러닝을 수행하여 학습자들의 학습동기 유발과 학습성과에 대한 기대감 면에서 긍정적인 결과를 얻었다 [20]. 외국의 경우에도 Chen과 그의 동료들도 C언어 수업에 플립드러닝을 도입하여 수업을 진행했는데, 기존의 강의에 비해 플립드러닝의 수업의 경우 중도 포기율이 줄어든 것을 확인할 수 있었다[9]. Amresh와 그의 동료들도 초급 프로그래밍 수업에 플립드러닝을 도입하였다. 이들의 경우 강의 자료를 MOOC에서 제공되는 자료를 이용하였으며, 플립드러닝을 통해 학습자들의 컴퓨팅에 대한 자기효능감이 높아진 것을 볼 수 있었다[21].

위와 같은 플립드러닝에 관한 선행 연구들의 분석 결과를 보면 플립드러닝에 관한 큰 틀 안에서 과목의 특징에 따라 세부 운영 전략들이 조금씩 다르게 적용된 것을 볼 수 있었다.

플립드러닝 수업을 설계하기 위해서는 먼저 왜 이 수업 방식을 채택하려고 하는 지에 대한 요구 분석 과정과 플립드러닝 방식을 통해 이루고자 하는 수업의 목표가 우선적으로 정립이 되어야 할 것이다.

특히, 대학의 컴퓨터 프로그래밍 과목의 경우, 초중고에서 접해보지 않은 학습자들이 대부분이기 때문에 그 과목 자체가 낯설고 어렵게 느껴지는 경향이 있다. 뿐만 아니라, 프로그래밍의 경우, 문제를 해석하고, 그 문제를 컴퓨터를 통해 처리하기 때문에 컴퓨팅 시스템의 처리 구조와 처리 과정에 대한 지식의 부족으로 인하여 어려움을 느낄 수 있다. 더구나, 프로그래밍 문법을 이해한다고 해서 프로그램을 작성할 수 있는 것은 아니기 때문에 학습한 구문을 활용하여 충분히 프로그램을 작성할 수 있는 기회를 가지는 것이 필요하다. 이러한 프로그래밍 과목의 특성에 따른 적절한 교수 전략을 고안하여 플립드러닝의 수업을 설계하는 것이 요구된다.

3. 프로그래밍 수업을 위한 플립드러닝 수업 모형

3.1 수업 모형 설계를 위한 고려사항

본 연구에서는 프로그래밍 수업을 위한 플립드러닝 수업 모형의 설계를 위해 다음과 같은 사항

을 고려하였다.

첫째, 플립드러닝 수업 모형의 도입을 위한 요구 분석을 수행하였다. 플립드러닝 수업을 통해 도달하고자 하는 학습 목표를 수립하였다.

둘째, 프로그래밍 과정을 분석하여 각 단계에서 요구되는 인지적인 측면을 고려하였다.

셋째, 학습자들이 프로그래밍에 대해 어려워하는 부분 및 프로그래밍 초보자가 저지르기 쉬운 오류 형태를 분석하여 그러한 오류들을 줄일 수 있도록 수업을 설계하였다.

넷째, 학습자간의 상호작용이 활발하도록 유도하고 학습자 자신이 이해한 내용을 동료학습자에게 설명할 수 있는 기회를 많이 가질 수 있도록 수업을 설계하였다.

다섯째, 학습자들의 문제해결력을 높이기 위해 문제기반학습을 부분적으로 적용하였다.

본 연구에서는 위와 같은 고려사항에 기반하여 플립드러닝 수업 모형을 설계하였다. 특히 초보 프로그래머가 쉽게 저지르기 쉬운 오류 형태를 분석하고 이에 대한 학습 전략을 <표 1>과 같이 정리하였다. 이 오류 유형은 Mayer가 제안한 오류 유형을 수정하여 4가지 유형으로 구분한 것이다. 이 오류를 줄이기 위한 학습 전략을 고안하고 이를 포함하여 수업 설계를 하였다.

3.2 수업 설계 전략

본 연구에서 제안하는 수업 모형은 [그림 1]과 같이 교실전 학습, 교실 수업, 교실 후 수업으로 구분된다.

1) 교실전 학습

교수자는 교실전 수업에서 학습자들이 미리 학

습 내용에 대한 예습을 할 수 있도록 학습 동영상 및 학습 자료를 제공하였다. 이 단계에서 학습이 교실수업의 학습과정과 연계가 되도록 설계하고 학생들에게 이를 주지시켰다. 학습자들은 자기 주도적 학습을 수행하게 되는데 학습에 충실할 수 있도록 다양한 전략들이 고려되었다.

먼저, 교수자는 학습 동영상을 제작하는데 있어 학습자들의 학습 집중을 높이기 위해 각 동영상의 경우 최대 15분을 넘지 않도록 제작하였다. 이를 위해 학습내용을 분석하여 중요한 핵심 위주로 간추렸다. 또한, 동영상 시청이 잘 이루어지고 학습 동기를 높일 수 있도록 동영상을 시청하면 해결할 수 있는 학습 퀴즈를 제공하여 작성하여 제출하도록 하였다. 마지막으로 학습하는 과정 중에 어렵거나 의문사항이 있으면 메모하여 제출하도록 하였다.

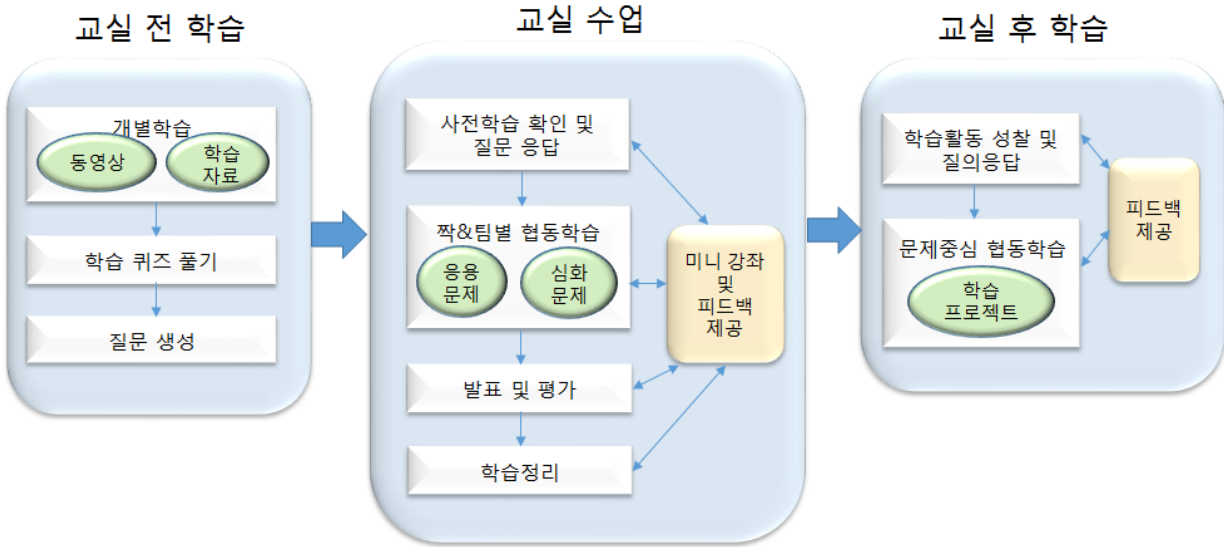
특히, 첫 번째 동영상에서는 프로그래밍 과정에서 많이 발생하는 오류의 유형과 이를 해결하는 과정을 보여줌으로써 실행상의 어려움을 최소화하도록 하였다. 뿐만 아니라, 교실전 학습 단계에서는 프로그래밍 언어의 문법적인 부분과 그 문법에 관련된 간략한 프로그램의 실행을 설명하고 있다. 특히, 학습자들의 심성모형을 형성할 수 있도록 프로그램 명령의 실행시 내부 메모리상의 변화 등을 이미지로 나타낸 자료를 사용하여 설명하였다.

2) 교실 수업

교실 수업에서는 학습자들의 사전학습이 잘 이루어졌는지를 체크하기 위하여 학습자들이 제출한 퀴즈 답안 채점 결과와 학습자들의 질문을 토대로 학습자들이 잘 이해가 안 되는 부분을 다시

<표 1> 오류 유형에 따른 학습 전략

오류 유형	학습전략	학습단계
문제 해석 오류	• 토론학습, 짝학습을 통해 서로 문제를 이해한 내용을 서로 설명	교실 수업
전략적 오류	• 주어진 문제를 해결하기 위해 문제분해, 패턴인식, 추상화 하는 과정에 대해 시범을 보여주고 학생들의 활동 유도	교실 수업
번역 오류	• 심성모형을 형성할 수 있도록 프로그램 명령의 실행시 내부 메모리 상의 변화 등을 이미지로 표현하여 제공	교실전 수업
	• 프로그램 구문에 대한 프로그램으로의 적용능력을 키우기 위해 유사한 사례들을 제공	교실 수업
코딩 오류	• 프로그래밍 과정에서 많이 발생하는 오류의 유형과 이를 해결하는 과정을 보여줌	교실전 수업



[그림 1] 플립드러닝 수업 모형

설명해주고 문제를 같이 풀어보도록 하였다. 상황에 따라서 개념의 이해를 돕기 위해 미니 강의도 제공하였다.

또한 배운 내용을 토대로 프로그램 코딩을 할 수 있도록 응용문제를 제시하였다. 그 응용문제는 프로그램에 대한 알고리즘과 프로그램 코드에 대한 부분내용이 기술되고 나머지 부분을 완성하는

학습지 형태로 제공되었다. 먼저, 학습자는 스스로 문제를 해결해보는 시간을 가져 본 다음에 짝(동료)과 함께 그 문제를 같이 토의하는 시간을 갖도록 하였다. 이때 짝과 함께 각자가 문제에 대해 이해한 내용을 설명하는 시간을 갖도록 하였다. 그다음 한 팀을 선정하여 서로 설명한 내용을 전체 학생 앞에서 발표하도록 하였다. 교수자는

<표 2> 플립드러닝 수업 전략

	교실전 학습		교실 수업		교실 후 학습	
학습내용	<ul style="list-style-type: none"> 자바언어의 문법 및 간단한 예제 프로그램 		<ul style="list-style-type: none"> 교실전 수업에서 학습한 구문을 적용하는 응용프로그램 및 심화 프로그램 		<ul style="list-style-type: none"> 프로젝트 진행 	
학습방법	<ul style="list-style-type: none"> 개별학습 자기주도적 학습 		<ul style="list-style-type: none"> 시범수업 탐구수업 미니강의 짝학습과 팀별 학습 		<ul style="list-style-type: none"> 문제기반학습 반추학습 	
교수학습 활동	교수자	학습자	교수자	학습자	교수자	학습자
	<ul style="list-style-type: none"> 동영상 제공 학습자료 제공 학습지 제공 	<ul style="list-style-type: none"> 동영상 시청 및 학습자료 읽기 학습지 풀기 질문 작성 	<ul style="list-style-type: none"> 질의응답 교실전 수업에서 제공된 내용과 유사한 다른 예제 프로그램을 준비하여 프로그램 작성 시범 및 설명 학생들 관찰 및 피드백 제공 	<ul style="list-style-type: none"> 교수자가 설명한 프로그램과 유사한 프로그램을 학습자료에서 찾아 서로 비교해보기 관련 응용 프로그램 작성 작성할 프로그램에 대해 짝과 토론 및 팀 활동 프로젝트 진행 사항에 대한 팀별 발표 	<ul style="list-style-type: none"> 질문에 대한 피드백 제공 	<ul style="list-style-type: none"> 프로그램 작성과정의 에러 사례 기술 및 반추 프로젝트 진행을 위한 팀별 협동학습
학습성취 목표	<ul style="list-style-type: none"> 자바 문법 지식의 이해 		<ul style="list-style-type: none"> 문법 지식의 적용 및 프로그램 작성 능력 함양 		<ul style="list-style-type: none"> 문제해결능력 함양 	

이에 대한 피드백을 하였다. 이때 한 팀은 2쌍의 짝으로 구성되며 1쌍은 2명으로 구성되었다. 짝끼리 해결이 안 되는 경우에는 그 팀의 다른 쌍에게 도움을 받도록 하였다. 또한 문제를 보고 바로 코딩을 하기 보다는 그 문제를 분석하고 어떻게 해결할 것인지를 생각해보는 시간을 갖도록 유도하였다.

또한 단원별로 심화 프로그램 작성 단계가 있으며, 이를 위해 문제해결의 전략적 측면에서 접근할 수 있도록 지도하였다. 즉, 주어진 문제를 분석한 후 필요하다면 문제를 분해하고, 기존에 학습한 내용 중 주어진 문제와 유사한 문제가 있었는지 찾아보도록 하였다. 또한 문제의 중요한 핵심을 정리하여 UML로 표현하도록 하였다.

학습한 문법 요소들을 포함하여 응용 프로그램을 작성하는 것을 프로젝트 과제로 부여하여 학기말에 제출하도록 한다. 이에 대한 진행 과정을 수업 시간에 팀별로 발표하는 시간을 갖도록 하였다.

3) 교실 후 학습

이 단계에서 학습자들은 학습자 개인의 학습 활동에 대한 성찰을 할 수 있도록 하였다. 먼저, 자신의 수업에 대한 적극성 및 학습에서 어떤 부분이 어려웠고 힘들었는지를 돌아보는 시간을 갖는다. 특히 어떤 부분에서 오류가 발생되었는지 또한 그것을 어떻게 해결하였는지를 간단히 작성해보도록 하였다. 이에 대한 피드백을 교수자가 제공하였다. 한편 학습자는 학기말에 제출할 프로젝트에 관한 협동학습을 교실 후 학습단계에서 하도록 하였다.

단계별 협동학습 한 내용을 교실 수업에서 발표하도록 하였다. 특히, 팀별 발표자는 그날 뽑기를 해서 발표하도록 함으로써 모든 학습자가 팀의 프로젝트에 대해 이해하도록 하고 발표 준비를 할 수 있도록 유도하였다.

4. 연구 방법 및 절차

4.1 자바 수업을 위한 플립드러닝 수업 설계 후 전문가 검토

본 연구에서는 플립드러닝 수업 모형을 설계하

기 위해 3명의 전문가로부터 자문을 받았다. 또한 수업 모형을 개발한 후 그 모형의 타당성의 검증을 위해 전문가들의 검토를 받았다. 전문가들은 우선 동영상 제작시 가급적 15분을 넘지 않도록 하고 학생들의 동영상 시청을 확인할 수 있는 문제를 제시하는데 있어 너무 어려운 문제를 제시하기 보다 동영상 시청을 한 후에 해결할 수 있는 간단하고 쉬운 학습 퀴즈를 제공하도록 조언하였다. 또한 교실 수업에서는 학습자의 이해를 높이기 위한 유사한 문제들을 제시해주고 프로그램 작성을 시범으로 보여주는 것도 필요하다는 조언을 하였다. 이들 내용을 반영하여 수업 모형을 수정 보완하였다.

4.2 자바 수업을 위한 플립드러닝 적용

본 연구는 W 대학의 정보관련 학과의 자바 프로그래밍 수업을 듣는 2학년 남녀 학생 29명을 대상으로 수행되었다. 이들 학생은 선수과목으로 C언어 수업을 이수하여 프로그래밍에 대한 기본 지식을 가지고 있었다.

중간고사 이후부터 플립드러닝 수업이 진행되도록 수업을 설계하여 적용하였다. 학기초부터 바로 플립드러닝을 진행하기보다 학생들로 하여금 플립드러닝에 대한 이해를 높이고 그에 대한 마음의 준비를 할 수 있도록 시간을 두고 플립드러닝 소개 동영상 및 운영방법 등을 제시하였다.

플립드러닝을 위해 학교의 LMS 및 SNS를 이용하였다. 교실수업전 학습을 위해 동영상 및 학습자료를 LMS 및 강의 카톡방에 업로드했다. 특히 강의 카톡방에 학습자료 및 학습안내 등을 함으로써 학생들이 보다 쉽게 접근하여 학습할 수 있도록 하였다. 학습퀴즈 및 과제 제출은 LMS 시스템을 이용하였다.

강의 비디오 제작을 위해 오캠(oCam) 툴을 이용하였으며 학생들의 집중도를 높이기 위해 각 파일당 최대 15분을 넘지 않도록 하였다. 한주의 강의는 보통 3개 정도 업로드 하였다.

4.3 적용 결과

1) 플립드러닝의 학습 효과에 대한 인식

본 연구에서는 자바 프로그래밍 수업을 위해

플립드러닝을 적용하고 그 효과를 조사하였다. 플립드러닝 수업의 학습적인 면에서의 효과를 5점 척도로 조사한 결과, 학습에 도움이 된 정도는 3.94, 학습의 동기부여 정도 3.87, 프로그램 작성 능력에 도움 정도는 3.85, 수업의 집중도는 3.89로 나타났다.

<표 3> 플립드러닝 적용 결과

평가 항목		N	평균	표준 편차
학습 효과 평가	플립드러닝 방식으로 진행한 수업이 학습에 도움이 되었다.	29	3.94	.553
	플립드러닝 수업은 학습에 대한 동기부여를 제공한다.	29	3.87	.654
	플립드러닝 수업은 자바 프로그램 작성 능력을 키우는데 도움이 된다.	29	3.85	.586
	플립드러닝 수업은 학습의 집중도를 높여준다고 생각한다.	29	3.89	.593
학습 과정 평가	플립드러닝 수업방식으로 학습하는데 어려움이 없었다.	29	3.67	.686
	교실안 수업을 통해 나는 다른 학생들과 많이 소통할 수 있는 기회를 가졌다.	29	4.02	.673
	플립드러닝 수업은 일반 수업에 비해 학습준비 시간이 많다.	29	4.32	.632
	플립드러닝 수업은 학습의 참여도를 높여준다고 생각한다.	29	4.08	.592
만족도	플립드러닝 수업방식에 대해 전반적으로 만족한다.	29	4.01	.587
	플립드러닝 수업 방식을 다른 학생들에게 추천하겠다.	29	3.99	.569

2) 플립드러닝의 학습과정에 대한 인식

플립드러닝 수업과정에서 학습자들의 활동 및 그에 대한 인식을 조사한 바에 의하면 플립드러닝 수업의 학습 어려움 정도는 3.67, 수업과정안에서 학생들간의 의사소통 정도 4.02, 학습의 준비 정도 4.32, 학습의 참여도 4.08 로 나타났다. 학습자들은 학습에 대한 큰 어려움은 없지만 학습을 준비하는데 기존의 수업방식보다 미리 연습하는 시간이 필요하기 때문에 학습에 대한 부담이 어려움으로 작용하는 것으로 나타났다. 학습의 참여도와 동료학생간의 의사소통이 기존의 수업

보다 많아진 것을 확인할 수 있었다.

3) 전반적 만족도

플립드러닝 수업에 대한 학생들의 전반적인 만족도는 4.01 이며, 이 과목을 다른 학생들에게 추천할 것인가에 대한 항목에서는 3.99로 나타났다. 추천하지 않는 이유로는 학업에 대한 부담감이 가장 높았다. 다른 과목에 비해 미리 시간을 내어서 학습하고 수업에 참여하는 것이 부담이 되었다고 응답하였다. 또한 동영상으로 강의를 듣기보다 직접 수업시간에 교수자로부터 강의를 듣는 것을 선호한다는 의견도 있었다. 뿐만 아니라, 수업내 짝학습과 팀활동에서도 다른 사람과 같이 학습하고 작업하는 것이 불편했다는 의견을 나타내기도 하였다.

5. 결론 및 제언

5.1 결론

본 연구에서는 자바 프로그래밍 수업을 플립드러닝기반 수업으로 설계하여 적용한 후 그 효과를 검증하였다. 기존의 프로그래밍 수업의 경우, 강의시간에 프로그래밍 문법과 관련 기초 프로그램을 배우고 그것을 활용할 수 있는 심화문제를 과제로 제공하는 방식이었다. 그런데, 많은 학생들이 수업시간에 학습하는 문법적인 요소들과 그에 관한 간단한 프로그램을 이해하는 것은 큰 무리가 없지만 그것을 적용한 보다 심화된 프로그램을 작성하는 데는 어려움이 따른다는 문제가 대두되었다. 이에 대한 하나의 해결책으로 플립드러닝을 고려하였다. 플립드러닝 수업으로 진행할 경우 미리 교수자가 준비한 동영상을 통해 문법적인 부분을 연습한 후 교실 수업에서는 그와 관련된 심화된 응용 프로그램을 작성하게 된다. 이때 프로그램 작성과정에서 발생하는 어려움을 동료들과 같이 토의할 수 있고 또한 교수자로부터 직접 피드백을 받을 수 있다는 이점이 존재한다.

특히 본 연구에서는 프로그래밍 학습과정에서 학습자들이 부딪히는 어려움과 실제 프로그래밍 과정에서 발생하는 오류들을 분석한 후 이를 기반으로 하여 플립드러닝 수업을 설계하였다.

이러한 플립드러닝 수업을 위해 먼저, 강의 동

영상을 준비해야 되는데, 학생들의 집중도를 고려하여 최대 15분을 넘지 않도록 제작하였다. 15분 이내로 동영상 만들기 위해 중요한 핵심을 간추리는 등의 학습내용을 재구성하는 작업이 간단하고 쉬운 작업은 아니었다. 또한 이러한 학습내용은 교실수업에서의 학습 활동과 관련을 두고 설계해야하기 때문에 많은 사고가 필요한 과정이라 볼 수 있다.

또한 교실수업에서는 학습자들이 온라인 수업에서 학습한 내용을 확인하고 이를 적용하고 심화하는 과정이 필요하다. 이를 위해 먼저, 온라인 학습과정에서 학습자들이 동영상을 시청하면서 이해가 안 되는 부분에 대해 질의하는 시간을 갖도록 하였다. 또한 학습한 내용의 적용과 심화 학습을 위해 학습자 중심의 학습 활동으로 짝학습과 팀학습 활동이 이루어지도록 하였다. 교수자는 학습자 중심의 활동을 지켜보며 적절한 피드백을 제공했다. 이러한 활동중심의 학습과정에서 발생하는 예상치 못한 학습자들의 질문에 대해 적절한 피드백을 제공하는 과정은 교수자로 하여금 늘 긴장감을 갖게 하였다.

플립드러닝 수업에 대한 학생들의 반응은 전반적으로 프로그래밍 학습에 도움이 되었다는 의견을 주었다. 또한 학습과정에서 학습자간의 의사소통이 활발하게 이루어지고 학습의 참여도가 높아졌다는 의견이 많이 나왔다.

플립드러닝 수업의 경우 기존의 수업 방식에 비해 학습자에게 자신의 학습에 대한 책임감을 부과하기 때문에 학습의 주도성을 가지고 학습을 하는 학생의 경우에는 많은 긍정적인 면을 볼 수 있다. 하지만 수동적인 학습 태도를 가지고 있는 학생의 경우에는 미리 학습하고 수업에 들어오는 것을 많이 부담스러워하는 면이 있었다. 그러한 학습자들의 학습 동기를 어떻게 이끌어서 수업에 적극적으로 임하게 할 것인가가 고민해야 될 문제이다. 또한 학습자의 성향에 따라 외향적이고 적극적인 학생의 경우에는 교실안 수업에서의 여러 활동에도 활발히 참여하며 즐기는 것을 볼 수 있었다. 하지만 학습자 성향이 내성적인 경우, 교실안 활동에서 짝학습과 팀활동 학습에 참여하는 것에 힘들어하고 소극적인 자세를 취하였다. 이러한 학습자들을 어떻게 이끌어갈 것인가도 앞으로

고민해야 될 문제이다.

5.2 제언

플립드러닝의 교실안 수업은 학습자가 주도적으로 참여하는 방식으로 진행되므로 학습자들간의 협업이 잘 이루어질 수 있는 물리적 환경이 지원되는 것이 필요하다. 본 연구에서 수업이 진행되었던 강의실은 교사를 바라보는 일자식 형태의 책상 배열로 이루어졌기 때문에 학습자간의 협업을 하는데 좀 제약점이 있었다.

자바 프로그래밍을 위한 플립드러닝 수업은 학생들이 프로그램 작성 능력을 향상시키고 학습 동기를 높이는데 긍정적인 영향을 미쳤다. 하지만 기존의 수업에 비해 학습자와 교수자 모두에게 더 많은 시간과 노력을 요구하기 때문에 부담이 된 면도 있었다. 특히 교수자 입장에서 기존의 수업을 준비하는 것보다 더 많은 사항들을 고려해야 되고 수업을 위한 동영상 녹화와 여러 종류의 학습지를 준비해야하기 때문에 많은 부담이 되었다. 이러한 문제들을 어떻게 해결할 것인가도 앞으로 생각해봐야 할 것이다.

또한 기존의 수업 방식과 정반대의 수업 방식이기 때문에 학습자 입장에서 수업진행 방법에 대한 충분한 숙지가 되어 있지 않은 상황에서는 많은 혼돈을 가져올 수 있다. 따라서, 학기 초나 수업이 시작되는 학기 전에 미리 플립드러닝 방식에 대한 충분한 안내가 지원되어야 하며 학습자 입장에서 무엇을 어떻게 할 것인가에 대한 세부적인 지침들을 제공하는 것이 필요하다고 볼 수 있다.

프로그래밍 과목의 경우 플립드러닝 수업을 진행한다면 학습자들의 프로그래밍 능력을 향상시킬 수 있을 것이다. 하지만 이러한 긍정적인 효과를 얻기 위해서는 프로그래밍 과목의 특징에 맞는 보다 세부적인 학습전략에 대한 연구가 앞으로 필요하다.

참 고 문 헌

- [1] International Society for Technology in Education & Computer Science Teachers Association. (2011). *CSTA K-12 Computer*

- Science Standards Revised 2011*. Retrieved from http://csta.acm.org/Curriculum/sub/urrFiles/CSTA_K-12_CSS.pdf.
- [2] Department for Education. (2013). The national curriculum in England: Framework document. Retrieved from https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/210969/NC_framework_document_-_FINAL.pdf.
- [3] 배학진, 이은경, 이영준 (2009). 문제중심 학습을 적용한 스크래치 프로그래밍 교수 학습 모형. **컴퓨터교육학회 논문지**, 12(3), 11-22
- [4] 이지연, 김영환, 김영배, 학습자 중심 플립드 러닝 수업의 적용사례. **교육공학연구**, 30(2), 2014, 163-191
- [5] 이동엽 (2013). 플립드러닝(Flipped Learning) 교수학습 설계모형 탐구. **디지털정책연구**, 11(12), 83-92.
- [6] Bergmann, J., Overmyer, J., & Wilie, B. (2013, July 9). The flipped class: what it is and what it is not. Retrieved from <http://www.thedailyriff.com/articles/the-flipped-classconversation-689.php>.
- [7] Hamdan, N., Mcknight, P., Mcknight, K. & Arfstrom, K. (2013). A REVIEW OF FLIPPED LEARNING. Retrieved from <http://www.flippedlearning.org/>
- [8] 이종연 외 (2014). Flipped learning의 의의 및 교육환경에 관한 탐색적 연구, **디지털융복합연구**, 12(9), 2014, 313-323.
- [9] Chen, Y., Wang, Y., Kinshuk, & Chen, N. S. (2014). Is FLIP enough? Or should we use the FLIPPED model instead?. *Computers & Education*, 79, 16-27.
- [10] 김연경 (2016). **대학수업을 위한 활동이론 기반 플립드 러닝 수업모형 개발**, 박사학위 논문, 중앙대학교
- [11] 최현종 (2011). 대학 프로그래밍 강좌를 위한 프로그래밍 교육 프레임워크, **컴퓨터교육학회 논문지**, 14(1).
- [12] Mayer, R. E. (1992). *Thinking, problem solving*. New York : W. H. Freeman and Company.
- [13] 이정모, 이건효, (1998). 초보자의 C언어 학습 과정에 대한 인지심리학적 분석 연구 : 프로그래밍 학습과정 동안의 은유 사용의 효과. **인지과학**, 9(4), 2-3,
- [14] Mayer, R. (Ed.)(2013). *Teaching and Learning Computer Programming: Multiple Research Perspectives*, Routledge.
- [15] 김보경(2014). 교직수업을 위한 역진행 수업 모형 개발. **교육종합연구**, 12(2), 25-56.
- [16] 권오남 외 (2013). 반전학습 (Flipped Classroom) 수업 모형 개발: 대학 미적분학 강의 사례를 중심으로. **수학교육논집**, 30, 91-111.
- [17] 최정빈, 김은경(2015). 공과대학의 Flipped Learning 교수학습 모형 개발 및 교과운영사례. **공학교육연구**, 18(2), 77-88.
- [18] 정동섭(2015). 교양 스페인어 수업의 플립드 러닝(Flipped Learning) 수업모형 제안 -말하기 중심으로 진행되는 전북대학교 교양 스페인어 수업의 사례-. **스페인라틴아메리카연구**, 8(1), 127-150.
- [19] 김경미, 이강(2016). 사전질문을 이용한 플립드 수업모델과 이 모델의 프로그래밍 초보 학습자 대상 적용 사례, **한국컴퓨터교육학회 하계 학술발표회논문집**, 20(2).
- [20] 유상미 (2015). C 프로그래밍 수업에서 플립드러닝 적용 사례 연구. **한국인터넷정보학회 추계학술발표대회 논문집**, 149-150.
- [21] Amresh, A., Carberry, A., & Femiani, J., (2013). The Effectiveness of Flipped Classrooms or Teaching CS1. Retrieved from https://interactive.asu.edu/p-content/papercite_data/pdf/amresh2013evaluating.pdf



최 속 영

1998 전북대학교 전산학과(이학사)
 1991 전북대학교 전산학과(이학석사)
 1996 충남대학교 전산학과(이학박사)
 2008 Nova Southeastern University
 교육공학 및 원격교육(교육학박사)

1996~현재 우석대학교 정보보안학과 교수
 2012.3.~2013.2 North Carolina State University 연구교수
 관심분야: 컴퓨터과학교육, 이러닝시스템, 사이버불링
 E-mail : sychoi@ws.ac.kr