

https://doi.org/10.7236/JIIBC.2023.23.3.215
JIIBC 2023-3-29

플립드러닝을 활용한 프로그래밍 교과목 개발 사례 연구

A Case Study on the Development of Programming Subjects Using Flipped Learning

허원희*

Won-Whoi Huh*

요약 현실 세계와 유사한 모델링이 가능한 객체지향 언어인 C++ 프로그래밍 수업을 플립드러닝 학습모형을 도입한 교육과정으로 개발한다면 학생들의 능동적인 문제해결 능력을 배양할 수 있다. 본 교과목 개발사례에서는 플립드러닝 기법을 프로그래밍 수업에 적용하여 학생들의 능동적인 문제해결 능력 향상에 효과가 있었다는 데 그 의의가 있다. 먼저 4차시의 강의를 Pre-Class, In-Class, Post-Class로 나누어 주제에 맞는 수업 목표를 제시하고 팀을 구성하여 토의하는 방식으로 수업을 진행하였다. 강의 마지막에는 사후 설문 조사를 시행하여 학습자들이 효과적으로 학습하였는지 확인하였다.

Abstract If the C++ programming class, an object-oriented language capable of modeling similar to the real world, is developed as a curriculum that introduces the flipped learning model, students' active problem-solving skills can be cultivated. In this subject development case, it is significant that the flipped learning technique was applied to the programming class and was effective in improving students' active problem-solving skills. First, the lectures in the 4th session were divided into Pre-Class, In-Class, and Post-Class, and the class was conducted in a way that suggested class goals suitable for the subject and formed a team to discuss. At the end of the lecture, a follow-up survey was conducted to check whether the learners learned effectively.

Key Words : Flipped learning, programming, subject development, case studies

1. 서론

1. 플립드러닝을 적용한 프로그래밍 교과목 개발의 필요성

플립드러닝이란 기존의 전통적인 교육 방식이 아닌 뒤집힌 교육 방식으로, 강의에 참석하기 전에 강의 영상을 먼저 시청한 후 강의 시간에는 해당 내용에 관한 이야기

를 나누며 교수에게는 피드백을 받을 수 있는 방식이다^[1]. 플립드러닝 수업에서 제공하는 사전 영상은 언제든지 학습자가 원하는 시간에 학습할 수 있어서 개인별 학습 속도를 조절할 수 있다. 또한, 반복 학습이 가능하므로 심화 보충 활동을 수행할 수 있어 학습 내용에 대한 이해가 높아진다^[2]. 따라서 현실 세계와 유사한 모델링이 가능한 객체지향 언어인 C++ 프로그래밍 수업을 플립드러

*정회원, 성결대학교 미디어소프트웨어학과
접수일자 2023년 2월 17일, 수정완료 2023년 4월 3일
게재확정일자 2023년 6월 9일

Received: 17 February, 2023 / Revised: 3 April, 2023 /
Accepted: 9 June, 2023

*Corresponding Author: wonwhoi@naver.com
Dept of Media Software, Sungkyul University, Korea

닝 학습모형을 도입한 교육과정으로 개발한다면 학생들의 능동적인 문제해결 능력을 배양할 수 있다. 이혜진(2018)은 팀 기반 학습과 플립드러닝에 관한 메타분석 결과로 전통적인 수업보다 학습자 중심의 교육방법이 교육적 효과가 있고, 의사소통 역량과 자기 주도학습, 내재 동기 및 자기효능감 등에서도 역시 효과적이라고 주장하였다^[3]. 또한, 서응교(2017)는 플립드러닝을 도입한 프로그래밍 강의에서 창의적 사고와 문제해결력이 유의미하게 상승한다고 하였다^[4]. 이처럼 플립드러닝을 활용한 학습자 중심 교육을 통해 c++ 프로그래밍 교과목에서 요구하는 공학 이해, 검증능력, 모델 능력을 충분히 향상할 수 있을 것으로 기대한다.

2. 연구 목적 및 방법

대부분의 프로그래밍 수업은 강의를 중심으로 언어의 문법적 요소를 설명하고 예제 프로그램을 따라 하는 것에 그쳤다. 본 교과목 개발에서는 플립드러닝 기법을 활용하여 수업 전에 기초지식에 관한 동영상 강의를 시청하고 수업시간에는 사전에 학습한 내용을 적용하는 다양한 활동을 함으로써 학습자가 중심이 되어 과제를 해결할 수 있도록 기회를 제공한다. 동영상 강의를 제공하는 것은 학생들이 자신이 편하게 생각하는 장소에서 학습할 수 있고 학습자 개인의 필요에 따라서 재학습, 필요 부분의 선택 학습 등이 가능하며 접근성 또한 높아진다^[5].

수업 참여 학생의 수를 고려하여 제시된 과제를 팀별로 함께 풀고, 문제를 풀고 난 이후에는 각 팀에서 문제를 풀면서 새롭게 알게 된 내용이나 문제해결을 위해 사용했던 방법 등을 공유하는 시간을 갖는다. 이러한 점을 통해 학생들은 각자 선행 학습한 내용을 바탕으로 하여 교실 수업에서 부족한 부분은 보충학습을 진행하고, 관심이 있는 부분에 대해서는 심화 학습이 가능하다^[6].

마지막으로 수업 후 제시된 심화 과제를 개별로 풀어보는 것으로 주요 단원의 이해도를 높일 수 있다. 이를 위해 플립드러닝 수업 진행 후 설문을 실시하여 강의 후 인식에 변화가 있는지 알아본다.

II. 본 문

1. 교과목에서의 플립드러닝 구성

본 교과목은 4개의 수업 주제를 나누고 각 수업 주제는 Pre-class, In-class, Post-class (사전학습, 본 학

습, 사후학습)로 나누어서 진행하였다.

Pre-class는 미리 학습할 내용에 대한 개념 설명과 퀴즈 활동을 통해 무엇을 이해했는지, 어떤 개념을 이해하지 못했는지를 알고, In-class에서 해당 내용에 대한 질문과 토론을 통해 문제를 해결한다.

In-class에서는 사전 퀴즈와 미니 강의, 사전강의 내용을 활용하고 적용하는 활동, 요약정리 및 확인 퀴즈를 통해 학습을 진행한다^[7]. 이 과정을 통해 Pre-class에서 발생했던 문제점들을 해결하고 스스로 과제를 해결해 보며 능동적인 학습이 이루어지는 단계이다. 또한, 팀별 활동을 통해 자신이 이해한 내용과 이해하지 못한 내용, 과제 등에 대해 팀원 간 자유롭게 토의하고 다양한 해결 방법을 공유하는 것이 가능하다.

Post-class는 수업 중 마무리하지 못한 활동을 진행하거나 소과제를 진행하는 등 수업 후에 실행하는 단계로 앞서 학습했던 내용을 응용하여 심화 과제를 개인별로 해결해 볼 수 있도록 안내한다. 이 단계에서 자신이 이해한 내용을 다시 한번 확인하고 해당 주제에 대한 학습을 마무리한다.

2. 플립드러닝을 적용한 수업 내용

1) 팀 구성

플립드러닝 방식의 수업은 강의 전에 학습 영상을 미리 시청하고, 강의 시간에는 해당 내용에 대한 의견 나누기와 피드백 주고받기가 중요시된다. 따라서 학생들이 해당 강의에 대한 의견을 나누기 위해서는 팀 기반 학습이 필요하다. 팀을 구성하면 팀 안에서 각자 자신에게 부여된 임무를 수행하는 과정을 통해 자기 주도적인 학습이 가능하고, 제시된 팀별 과제를 함께 수행하며 창의적인 문제해결을 이룰 수 있다.

팀별 구성 인원은 C++ 프로그래밍 수강 인원수에 맞추어 4명~5명으로 구성하고, 해당 팀원은 학습자의 특성을 고려하여 구성하였다. 팀 내 회의 시간은 C++ 프로그래밍 강의의 총 강의 시간인 3시간 중, 50분~40분으로 구성되어 진행하였다. 이는 팀별 활동 후 교수자의 피드백 시간을 제외한 시간으로 팀별 회의가 중심이 될 수 있도록 추진하기 위함이다.

2) 플립드러닝 적용을 위한 수업 목표와 주제

가. 수업 목표

Pre-class에서 제공하는 사전강의 영상은 동영상을 직접 녹화하여 제공하는 방식으로 진행하였으며, 제작한

ppt 강의 교안을 바탕으로 In-class에서 적용할 이론에 대한 설명과 간단한 예제 코드의 실행 결과 및 설명을 추가하여 학생들이 코드의 구성에 대해서 이해할 수 있도록 하였다. 학습자들이 사전강의에 부담감을 느끼지 않도록 많은 양의 학습 내용이나 복잡한 구조의 내용은 자제하였고, 이해에 어려움이 예상되는 주제에서는 이해를 돕기 위한 영상 자료를 함께 활용하는 방식으로 진행하였다. 사전강의 영상의 마무리 단계에서는 간단한 예제 문제를 제시하여 학습자들이 시간의 제한을 받지 않고 실행해 볼 수 있도록 구성하여 스스로 해결 방법을 생각해 볼 수 있도록 하였다. 이후 강의의 마지막에 문제의 해답을 설명과 함께 제공하여 학습자들의 풀이 결과를 확인하고 강의 내용을 되돌아볼 수 있도록 구성하였다. Pre-class의 수업 목표는 표 1과 같다.

표 1. Pre-class의 수업 목표
 Table 1. Pre-class's objectives

수업주제	Pre-class 수업 목표
문자열 입력받는 방법	C++에서 알아야 할 요소와 그 개념을 설명할 수 있다. 토큰의 개념을 이해하고 식과 문의 예를 들 수 있다. 주석문을 사용할 수 있다.
연산자의 이해	삼항 연산자를 사용할 수 있다. if문을 조건 연산자로 바꿀 수 있다. sizeof 자료형의 크기를 구할 수 있다.
제어문의 이해	switch case 문을 이해할 수 있다. break:의 쓰임을 설명할 수 있다. 조건식(if)과 평가식(switch)의 차이를 안다.
포인터의 이해	참조 타입의 개념을 설명할 수 있다. 참조 타입의 형식을 이해한다. 참조 타입을 코드에서 사용할 수 있다.

In-class에서는 교수자가 Pre-class에서 학습한 전반적인 내용을 되돌아볼 수 있는 시간을 마련하고 팀별 활동을 위한 모임 장소(대면 수업일 때 강의실, 온라인 수업일 때 온라인 소회의실 등)를 제공해준다. 학습자들은 사전에 구성한 팀별로 모여서 수업 주제에 대한 논의를 진행하며 수업 주제에 대한 의견을 나눈다. 교수자는 이때 Pre-class 강의 내용을 기반으로 한 과제를 제시하고, 과제의 난이도는 Pre-class에서 제시하였던 기초 예제보다 높은 난도의 과제를 제시하여 학습자들이 함께 문제해결을 할 수 있도록 진행한다. 회의 시간이 종료되면 모든 학생이 모여서 각 팀에서 회의를 통해 산출된 결과를 다른 팀들과 공유하고, 자신의 팀이 해결했던 방법 및 논의 내용을 발표한다. 이때 교수자가 각 팀의 해결 내용 및 논의 내용에 대한 피드백을 제공하고, 회의 중 발생한 질문 사항에 대해 답변하는 시간을 갖는다. 이의

에 각 팀별 과제 수행 중 코드 실행 오류가 발생한 문제나 더 논의가 필요한 문제가 야기되었을 경우 결과 공유 시간에 다른 학생들에게 공유하고 이후 Post-class인 심화 학습에서 다른 학생들도 함께 해결하도록 한다. In-class의 수업 목표는 표 2과 같다.

표 2. In-class의 수업 목표
 Table 2. In-class's objectives

수업주제	In-class 수업 목표
문자열 입력받는 방법	소그룹 회의를 통해 Pre Class에서의 내용에 대하여 함께 토의하고 토큰, 식과 문, 주석문 등을 활용한 예제를 다양하게 만든다.
연산자의 이해	소그룹 회의를 통해 Pre Class에서의 내용에 대하여 함께 토의하고 이해하기 힘들었던 부분에 대해 서로 의견을 나눈다. 삼항연산자, sizeof 등을 활용한 예제를 스스로 만들어 본다.
제어문의 이해	소그룹 회의를 통해 Pre-class 내용을 확인한다. if, switch문을 활용한 예제를 스스로 만들어본다.
포인터의 이해	소그룹 회의를 통해 Pre-class내용을 확인한다. 참조 타입의 개념에 대해 서로 이야기 나눈다. 형식에 맞춰 참조 타입을 사용하여 코드를 작성한다. 구글링을 통해 예제를 찾아본다.

Post-class에서는 교수자가 In-class 활동에서 제시한 과제보다 심화된 과제를 제시하며 학습자는 해당 과제를 개인별로 해결한다. 개인별로 해결한 과제는 LMS에 제출하도록 한다. 이 과정에서 학습자는 심화 학습을 통해 해당 주제에 대한 이해도를 향상할 수 있고, 교수자는 학습자들의 개별 이해도를 점검하여 이후 전체적인 강의의 난이도를 결정에 해당 점검 내용을 반영할 수 있다. Post-class의 수업 목표는 표 3과 같다.

표 3. Post-class의 수업 목표
 Table 3. Post-class's objectives

수업주제	Post-class 수업 목표
문자열 입력받는 방법	프로그램 수업이 어렵지 않으며 충분히 할 수 있다는 자신감을 갖는다. 조별 활동을 통해 서로 부족한 부분을 채울 수 있다는 사실을 안다.
연산자의 이해	조별 활동을 통해 서로 친분을 쌓고 수업 참여에 동기를 만든다. 스스로 문제를 만들어 보면서 코딩의 이해도를 높인다.
제어문의 이해	팀 활동을 통해 심화된 문제를 만들어 봄으로써 코딩의 이해를 확장 시킬 수 있다.
포인터의 이해	팀 활동을 통해 난이도 있는 문제를 함께 고민해보고 풀어볼 수 있다. 좀 더 심화된 문제를 찾아보고 토의를 통해 이해할 수 있다.

나. 수업 주제

수업 주제는 4개의 주제로 구분하였으며 '문자열 입력 받는 방법', '연산자의 이해', '제어문의 이해', '포인터의 이해'로 나누어 실시하였다 각 Class의 세부 주제는 표 4에 작성하였다.

표 4. 수업 주제와 세부내용

Table 4. The subject and details of the class.

수업주제	Pre-class	In-class	Post-class
문자열 입력받는 방법	cin과 >> 연산자를 이용한 키 입력 (Enter) 키를 칠 때 변수에 값 전달 cin으로부터 키 입력받는 과정 이해	cin을 이용한 문자열 입력 cin.getline()으로 공백이 낀 문자열 입력	string 클래스를 이용한 문자열 입력 및 다루기
연산자의 이해	연산자의 개요 연산자의 종류	다양한 연산 실습하기	연산자의 우선순위 예제
제어문의 이해	선택문, 반복문, 다중선택문, 점프문의 이해	다양한 조건문 실습하기	제어문을 활용한 예제
포인터의 이해	포인터 원리 이해 포인터를 통해 변수 값 확인 포인터로 변수 값 변환	포인터 사용 실습하기	이중 포인터 예제

3) 교과목 수행 내용

가. Pre-Class

1차 Pre-class는 앞으로 학습해야 할 전반적인 C++의 구성 요소와 개념에 대해 설명하였다. 토큰의 개념과 종류에 대해 설명하고 식과 문의 예제, 주석문의 사용법 등을 통해 전반적인 C++ 언어의 사용법을 숙지하도록 하였다. 모든 Pre-class 강의 영상은 학습자들이 자유로운 시간에 수강할 수 있도록 수업 당일을 기준으로 일주일 전에 영상을 올리고 강의를 올렸다는 공지를 통해 학습자들이 수강할 수 있도록 하였다.

2차 Pre-class는 기타 연산자와 관련한 학습 목표를 설정하고 해당 내용에 대한 설명을 진행하였다. 삼항 연산자 퀴즈를 통해 학생들이 직접 해결해 볼 수 있는 시간을 할당하여 이해도를 높일 수 있도록 구성하였다. if문을 조건 연산자로 변환하는 방법과 size of를 사용하여 자료형의 크기를 구할 수 있는 예제를 첨부하여 해당 코드의 사용 방법과 결과를 확인해 볼 수 있도록 하였다.

3차 Pre-class는 제어문 중 switch case문에 대해 알아보고 함께 사용되는 break;의 쓰임에 대해 설명하였다. 이와 더불어 if문과 switch 문의 차이점에 대해 설명하고 예제 코드를 제공하여 실제 코드 실행 후의 결과에 어떠한 차이점이 있는지 알아볼 수 있도록 하여 이해도를 높이고자 하였다. Pre-class 종료 전 간단한 코드

를 작성해 볼 수 있도록 퀴즈를 제공하여 Pre-class의 내용을 다시 한번 점검해 볼 수 있도록 제공하였다.

4차 Pre-class는 참조 타입에 대한 개념과 원리를 설명하였다. 참조 타입의 형식에 대해 알아보고 예제를 통해 작동 원리를 살펴보았다. 또한, 참조 타입의 특징을 알아보고 앞서 학습해 보았던 포인터와의 차이를 인지할 수 있도록 포인터와 참조 타입의 예제를 함께 제공하여 설명하였다. 해당 주제의 경우 앞의 주제들보다 학습자들에게 어려움이 있을 것이라고 예상되어 추가 설명 자료를 함께 제공하여 학습에 도움이 될 수 있도록 하였다.

나. in-class

1차 In-class에 들어가기 전, 팀명 및 팀 구성원의 역할을 정하고, 학습자들이 해당 팀원들과의 친밀감 형성을 할 수 있는 시간을 마련하였다. 이는 4차시로 구성된 플립 러닝에서 큰 부분을 차지하는 팀 활동이 원활히 진행될 수 있도록 하기 위함으로, 간단한 자기소개 및 편안한 주제로 이야기를 나눌 수 있도록 진행하였다. 각 팀원이 결정한 팀명은 그림 1과 같다.



그림 1. 플립드러닝 팀명
Fig. 1. Flipped Learning Team Name

팀명을 정한 이후 실시한 In-class에서는 Pre-class 내용인 입력문과 출력문(cin, cout)을 사용하여 자유로운 주제로 코드를 작성할 수 있도록 과제를 제시하였다. 이후 그림2처럼 팀별 회의를 진행하고 팀에서 작성한 코드의 내용을 발표할 수 있도록 안내하였다.

2차 In-class는 지난 수행결과공유시간에 팀별 회의 시간이 길다는 의견을 수렴하여 학습자들에게 개선안을 제시하였다. 회의를 먼저 종료한 팀은 회의를 마쳤다는 것을 알리도록 하여 교수자가 전체 회의의 진행 상황을 알 수 있도록 하였다. 회의 시간은 40분 전후로 구성하여 실시하였고 삼항 연산자와 size of를 사용한 예제를 팀 별로 만들어 볼 수 있도록 과제를 제시해 주었다.

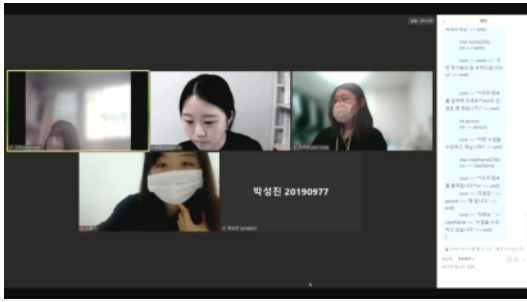


그림 2. 소회의실 팀별 회의
 Fig. 2. Meeting by team in a small conference room

회의 후, 수행결과공유시간에는 팀별 발표자가 자신의 팀에서 나누었던 회의 내용을 발표할 수 있도록 하고 팀에서 만들어 보았던 코드의 내용을 설명하여 그림3처럼 코드를 다른 학생들과 공유할 수 있도록 안내하였다.



그림 3. 팀별 수행 결과 공유
 Fig. 3. Share results by team by team

3차 In-class에서는 Pre-class에서 학습해 보았던 제어문의 종류와 switch case문을 이용하여 문제를 만들어 볼 수 있도록 안내하였다. 3차 In-class에서는 이전 회의 시간보다 회의 시간이 더 소요된 팀들을 발견할 수 있었다. 회의 리뷰를 통해 수업 주제보다 내용이 심화하였다는 점과 팀원들이 각자 다양한 방법의 코드를 작성해 보면서 시간이 소요되었다는 것을 알 수 있었다.

4차 In-class는 코로나바이러스와 관련한 사회적 거리 두기 시행 개편과 학사 운영에 대한 교육부 권고에 따라 대면 수업으로 진행하였다. 대면으로 시행한 첫 수업이기 때문에 팀원이 모일 장소를 마련하는 것에 어려움이 있을 것이라 예상하여 학습자들에게 강의실 내에서 팀별 활동을 진행할 수 있도록 자리를 안내하였고 회의 시작은 간단한 자유 주제로 이야기 나누도록 하였다. 이후 팀들의 분위기를 살피고 여유로운 분위기가 되었을 때 참조 타입과 관련한 과제를 제시해 주었다. 과제를 해결함과 동시에 학습한 내용 중 이해가 어려웠거나 질문

사항들을 서로 자유롭게 이야기 나눌 수 있도록 하여 전체적인 프로그래밍 강의 또는 플립드러닝에 대한 의견을 나누어 보도록 안내하였다. 그림4는 팀별 회의 모습이다.



그림 4. 팀별 회의 모습
 Fig. 4. A team meeting

다. Post-Class

1차 Post-class에서는 팀별로 친밀도를 높여, 앞으로의 팀별 활동이 수월하게 이루어질 수 있도록 하였다.

2차 Post-class는 팀별 단체 채팅방 등 팀원이 모일 수 있는 곳에서 자유로운 이야기를 나누며 친밀도를 쌓도록 하였고 학습자 개인이 삼항 연산자를 활용한 문제를 각자 만들어 볼 수 있도록 과제 내 학습에 어려운 부분은 자유롭게 질문할 수 있도록 하였다.

3차 Post-class는 그림 5처럼 지금까지의 강의 내용을 활용한 문제 3가지 만들기 과제를 제시하였다. 이를 통해 학습자들이 현재까지의 학습 내용을 되돌아볼 수 있도록 하고, 제출한 과제물을 바탕으로 학습자의 개별 학습 수준을 점검하고 학습에 어려움이 있는지 살펴보았다.

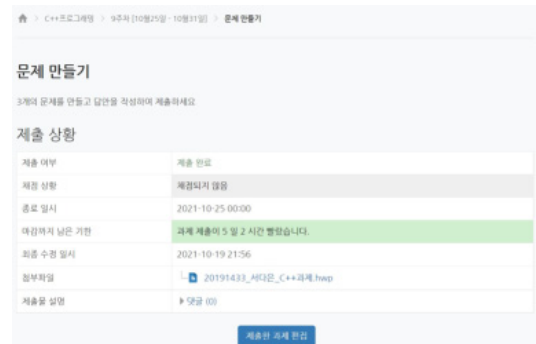


그림 5. 문제 만들기 과제 제출란
 Fig. 5. Column for submitting the 'Create a problem' task

4차 Post-class는 3차와 마찬가지로 문제를 생성하고 문제의 코드와 해답 코드를 작성하여 과제물로 제출하도록 하였다. 4차 플립드러닝 학습 주제에 어려움을 느끼는 학습자들이 있을 것을 고려하여 어려움이 발생했을 경우 자유롭게 질문할 수 있도록 하였다.

III. 수행 결과

1. 사후 설문 조사

사후 설문 조사는 수강 학생 55명을 대상으로 진행하였다. 해당 설문은 12항목으로 구성하였으며 ‘수업 참여’, ‘수업 과정’, ‘학습 향상’에 대한 항목으로 분류하여 실시하였다. 설문에 작성한 문항은 학습자들의 수업 후의 변화에 초점을 맞추어 구성하였고, 척도를 통한 답변 외에도 설문 문항을 추가하여 구체적인 답변을 확인할 수 있도록 그림 6과 같이 작성하였다.

그림 6. 사후 설문 조사
Fig. 6. Post-survey

사후 설문 조사에서 ‘수업 과정’ 항목에 대한 문항으로 ‘수업 내용이 체계적으로 구성되고 전개되었다.’, ‘교과목의 특성을 고려하여 적합한 수업방법을 사용하였다.’에서는 ‘매우 그렇다.’에 해당하는 5번 항목에 각 58.2%, 56.4%로 가장 높았고 ‘학습 향상’항목에서는 플립드러닝의 In-class와 Post-class에 해당하는 ‘수업 전에 주요한 개념들을 학습한 것이 수업 내용을 이해하는 데 도움

이 되었다.’, ‘나는 수업 후에 심화 과제 활동에 적극적으로 참여하였다.’ 문항에서 5번(매우 그렇다.)이 각 47.3%, 49.1%로 나머지 항목 중 가장 높은 비율을 차지하였다. 또한, 다른 수업 방식보다 이런 방식이 나에게 더 효과적이다.’ 문항에서는 긍정적 답변인 4번(그렇다.)과 5번(매우 그렇다.)이 67.3%로 높은 비율을 차지하였다. 이를 통해 플립드러닝을 적용한 C++ 프로그래밍 강의가 학습자들에게 긍정적으로 작용하였다는 것을 알 수 있었다. ‘다른 동료학습자들이 수업시간에 문제를 설명하는 것이 나의 학습을 증진시켰다.’ 문항에서도 5번(매우 그렇다.)이 49.1%로 가장 높은 비율을 차지하였으며 이를 통해 학생들 간의 능동적인 토론 활동을 통해 학습 능력이 증진되었다는 것을 알 수 있었다.

서술형 답변 중 “미리 강의를 보고 조원들과 대화를 나누며 이해를 높일 수 있어서 좋았다.”, “다른 수업보다 집중이 잘 되었던 것 같다.”, “우리가 코드를 만들고 서로 바꿔보으로써 평가도 하고 조원들끼리의 얘기도 나누면서 적합하다고 생각했다.”, “플립드러닝 수업할 때 교수님이 한 팀 한팀 돌아다니면서 감시하지 않으셔서 자유롭게 얘기해 볼 수 있는 것이 좋았다.”, “새로운 수업 방식이어서 새롭고 재미있었다.” 등의 긍정적 의견이 있었다. 부정적인 의견으로는 “조별 활동시간이 너무 많다.”, “플립드러닝이 나에게 맞는지 모르겠다.”, “개념 설명에 있어 조금 더 코딩해볼 수 있는 시간을 넉넉히 가지고 자율적으로 실행해보고 싶다.” 등이 있었다.

2. 수행 결과

본 교과목에서는 플립드러닝 적용여 C++ 프로그래밍 수업을 진행하고 학습자의 사후 인식을 통해 효과적인 학습이 이루어졌는지를 파악하고자 하였다.

플립드러닝 을 통해 학습자들이 자신의 학습 수준에 맞춰 학습한다면 학생들은 수동적인 수강자에서 능동적인 학습자로 바뀌고, 단순히 수업을 듣는 청취자에서 스스로 자료를 찾아 선행학습하는 능동적인 학습자로 변화할 수 있다[8].

강의 이후 실시한 사후 설문에서는 기존의 수동적인 강의방식에서 벗어나 토론을 통한 능동적인 수업 방식에 다수의 학생이 적극적으로 참여하였다는 결과를 얻을 수 있었다. 활동을 통해 사전 지식을 학습하는 것이 수업 내용을 이해하는 것에 도움이 되었으며, In-class의 토론 활동은 학습에 증진이 되었고, Post-class 활동에 적극적으로 참여하게 됨에 따라 학습자들에게 플립드러닝이 추구하는 사전-본-사후학습이 적절하게 이루어졌다는

것을 확인할 수 있었다.

개선점으로 팀별 활동의 빈도 조절하고 자율적인 문제 해결 시간 부족 등의 의견을 수렴하여 팀 활동의 규칙을 정하고 적용할 수 있도록 독려한다면 효과적인 학습이 될 수 있을 것으로 생각한다.

References

- [1] Redekopp, M. W., & Ragusa, G. "Evaluating flipped classroom strategies and tools for computer engineering", In 2013 ASEE Annual Conference & Exposition, pp. 23.548.1 - 23.548.18, 2013.
DOI:https://doi.org/10.18260/1-2--19562
- [2] Leem, Jung-Hoon. "Teaching and Learning Strategies for Flipped Learning in Higher Education: A Case Study", Journal of Educational Technology, Vol. 32, No. 1, pp. 165-199, 2016.
DOI:https://doi.org/10.17232/KSET.32.1.165
- [3] Lee, HyeJin, Ahn, Moonsang, Lee Jin Gu, "A meta-analysis of the Effects of Flipped Learning and Team-based Learning in College Classes" The Journal of Learner-Centered Curriculum and Instruction. Vol. 18, No. 21, pp. 1253-1280, 2018.
DOI:https://doi.org/10.22251/jlcci.2018.18.21.1253
- [4] Suh Eungkyo, "Development of Creative Thinking and Coding Course method on Design Thinking using Flipped Learning" The Journal of Learner-Centered Curriculum and Instruction, Vol. 17, No. 16, pp. 173-199, 2017.
DOI:https://doi.org/10.22251/jlcci.2017.17.16.173
- [5] Lee, In Sook, "Instructional Intervention Design Strategies for Flipped Learning in Higher Education". Asia-pacific Journal of Multimedia Services Convergent with Art, Humanities, and Sociology, Vol. 8, No. 1, pp. 401-408, 2018.
DOI:https://doi.org/10.35873/ajmahs.2018.8.1.041
- [6] Lee, Dong-yub, "Research on Developing Instructional Design Models for Flipped Learning", Journal of Digital Convergence, Vol. 11, No. 12, pp. 83-92, 2013.
DOI:https://doi.org/10.14400/JDPM.2013.11.12.83
- [7] Seonyoung Jang, Lee Jeongju, "Development of a Team-Based Learning Design Model in Higher Education", Asian Journal of Education, Vol. 16, No. 1, pp.271-302, 2015.
DOI:https://doi.org/10.15753/aje.2015.16.1.012
- [8] Lim, Eugene, Kim, Jong Sun, "A Study on the Effectiveness of Flipped Learning in the Design Courses of the University", Journal of Basic Design & Art, Vol. 20, No. 3, pp. 253-263, 2019.
DOI:https://doi.org/10.47294/KSBDA.20.3.20

저 자 소개

허 원 회(정회원)



- 1993년 2월 : 국민대학교 전자공학과
- 1997년 5월 : Pratt Institute Computer Graphics(MFA)
- 2012년 8월 : 서울과학기술대학교 디지털콘텐츠디자인전공 (디자인학박사)
- 2004년 3월 ~ : 성결대학교 미디어소프트웨어학과 교수
- 관심분야 : 모바일, IT, 콘텐츠디자인