J. Internet Comput. Serv.
ISSN 1598-0170 (Print) / ISSN 2287-1136 (Online)
http://www.jics.or.kr
Copyright ⓒ 2023 KSII

강의와 팀 활동을 조합한 컴퓨터학 과목의 플립러닝 기반 설계

Design of Computer Science and Engineering Courses based on Flipped Learning through Integrating Lectures and Team Activities

이 시 형1*
Sihyung Lee

요 약

플립러닝은 전통적인 교실 안 활동과 교실 밖 활동을 바꾸는 수업 방식으로, 기존에 교실 안에서 배우던 내용을 수업 전에 학생이 스스로 학습해온 후 교실에서는 기존에 교실 밖에서 하던 과제를 수행하는 방식을 말한다. 교수자의 도움을 받고 팀 동료와 협력하며 과제를 수행할 수 있다는 장점 때문에 컴퓨터학 과목에도 점차 널리 적용되고 있다. 하지만 수업 내용을 교실 밖에서 학생 홀로 학습해야 하므로 심화 내용에 대한 이해가 떨어진다는 단점이 있다. 본 연구에서는 플립러닝의 단점을 보완하고 장점도 유지하기 위해 플립러닝과 전통적인 수업을 조합하는 방식을 제안한다. 이 방식에서는 수업 전에 기초 개념에 대한 학습을 학생 스스로 하도록 하고, 심화 내용은 교실 안에서 교수자의 강의와 도움을 통해 배우도록 한다. 이처럼 심화 내용을 익힌 후에는 팀 활동을 통해 배운 내용을 응용하는 문제를 풀어보고 해법을 공유한다. 제안한 방식은 컴퓨터학 4개 과목에 1~4개 학기에 걸쳐 적용해 보았으며, 학생들의 학업 성취도와 만족도를 개선함을 확인할 수 있었다. 제안한 방식이 더 많은 컴퓨터학 과목에 활용되어 좋은 수업을 만드는 데 도움이 되기를 기대한다.

☞ 주제어 : 플립러닝, 컴퓨터학 교육, 수업 방법, 선행학습

ABSTRACT

Flipped learning is an instructional approach that reverses the traditional order of in-class and after-class activities. It entails students studying course materials before attending class, and then utilizing class time for completing homework tasks. Due to collaborative support available from teachers and peers, flipped learning has gained extensive adoption in computer science and engineering courses, enabling students to effectively engage in homework assignments. Nevertheless, students are responsible for studying class materials independently, which can limit their understanding of advanced topics. We propose an approach that combines both flipped learning and the traditional method, allowing them to mutually enhance each other. In the proposed approach, students acquire foundational concepts prior to attending class, and subsequently delve into advanced topics during classroom sessions through lectures and guidance provided by the teacher. Afterward, students collaborate with their peers to solve problems that involve the application of the concepts they have learned, and exchange a variety of solutions and perspectives. We implemented the proposed approach in four computer science and engineering classes, spanning one to four semesters and observed an enhancement in students' comprehension and satisfaction levels. We anticipate implementation of the proposed approach across various computer science and engineering courses, while enhancing their overall quality.

☞ keyword : Flipped Learning, Computer Science and Engineering Education, Teaching Method, Prior Learning

1. 서 론

플립러닝(flipped learning)은 강의형 수업을 보완하기 위해 개발된 수업 방식으로 최근에는 컴퓨터학 과목에도 많이 적용되고 있다. 강의형 수업에서는 교수자가 강의를 진행한 후 이에 대한 과제를 학생이 수업 후에 수행하는데, 교수자의 도움 없이 응용 혹은 심화 문제를 풀어야 하는 단점이 있었다. 이에 반해 플립러닝을 적용한 과목에서는 수업에서 배울 내용을 수업 전 사전학습을 통해 미리 익혀온 후 본 수업에서는 강의형 수업에서 하던 과제를 수행한다 [1]. 특히 과제를 교수자 및 팀 동료와 함께 수행하므로 도움을 받을 수 있으며, 다양한 의견도 들을 수 있다. 이러한 장점 때문에, 점차 더 많은 수업이 플립러닝을 적용하고 있다 [2].

이러한 추세에 따라 플립러닝을 적용한 많은 컴퓨터학

1 School of Computer Science and Engineering, Kyungpook National University, Daegu 41566, Korea
* Corresponding author (sihyunglee@knu.ac.kr)
[Received 18 June 2023, Reviewed 21 June 2023, Accepted 20 July 2023]

과목에서는 플립러닝을 적용하기 전 본 수업에서 강의하던 내용 대부분을 사전학습 시간에 학생이 스스로 공부해 오도록 했다 [3,4,5,6,7]. 이어지는 본 수업에서는 학생들이 팀을 이루어 사전에 학습한 내용을 응용한 문제를 풀어보고 그 결과를 발표하도록 했다. 이러한 팀 활동 중심의 수업 운영은 학생들이 적극적으로 수업에 참여하도록 하고 배운 내용을 응용하는 능력을 개발할 수 있다는 장점이 있다. 하지만 대부분의 이론 학습을 사전학습 시간에 학생이 혼자 수행해야 하므로 적절한 도움을 받지 못해 심화 내용에 대한 이해도가 떨어지게 되는 단점이 있다 [8].

본 연구에서는 플립러닝의 장점을 활용하되 단점을 보완하기 위해 강의형 수업과 팀 활동을 적절히 혼용하는 방식을 시도했다. 특히 사전학습은 학생 혼자 수행하므로 본 수업에 필요한 기초 지식을 익히도록 하고, 이를 심화한 내용은 본 수업에서 교수자의 강의와 도움을 받아 익히도록 했다. 이렇게 심화 내용을 이해한 후에 개별 혹은 팀 단위로 응용문제를 해결해보는 시간을 갖도록 했다. 본 연구의 공헌도와 주요 결과를 정리하면 다음과 같다.

- 사전학습에서 학생 스스로 익힐 내용과 본 수업에서 교수자와 함께 익힐 내용을 적절히 구분하는 기준과 이에 대한 구체적인 예를 제시했다. 또한 사전학습에 대한 온라인 퀴즈가 오답에 대한 피드백을 즉시 제공하며 재응시할 수 있도록 함으로써 학생 스스로 부족한 부분을 이해하고 본 수업에 참여할 수 있도록 했다.
- 본 수업을 심화학습과 팀 활동으로 구분하고 진행하는 방법과 구체적인 예를 제시했다. 또한 본 수업 후에 자유질문 시간을 가짐으로써 낙오 학생을 줄이는 방법도 제안했다.
- 컴퓨터학 4개 과목에 대해 플립러닝을 적용한 1~4개 학기와 (총 23개 분반, 수강생 962명) 플립러닝을 적용하기 전의 2~3개 학기에 대한 (총 15 분반, 수강생 491명) 학업 성취도와 만족도 분석을 통해 통계적으로 유의미한 결과를 얻고자 했다.

본 논문은 다음과 같이 구성된다. 2장에서는 본 연구와 관련 연구를 비교 분석한다. 3장에서는 본 연구에서 플립러닝을 적용한 방법을 예시와 함께 설명하며 4장에서 이러한 방법을 컴퓨터학 4개 과목에 적용한 결과를 분석한다. 마지막으로 5장에서는 향후 연구를 소개하며 결론을 맺는다.

2. 관련 연구

기존 연구에서는 알고리즘 [3,4], 자바프로그래밍과 자료구조 [5], 프로그래밍 언어 실습 [6], 데이터베이스 [7] 등 다양한 컴퓨터학 과목에 플립러닝을 적용하였다. 이들 연구에서는 이론 학습의 대부분을 사전학습 때 학생이 스스로 수행하도록 하고, 본 수업은 이를 활용한 팀 혹은 개별 활동 중심으로 진행하였다. 이러한 활동 중심 수업은 학생들의 참여도를 높일 수 있으나, 이론 학습을 교수자나 동료의 도움 없이 홀로 수행해야 해서 수업 내용에 대한 이해가 부족할 수 있다 [3,5,8]. 본 연구에서는 사전학습에서 스스로 익힐 수 있을 정도의 기초 지식을 익히도록 하고, 본 수업에서 교수자 및 동료와 함께 심화학습을 하도록 수업 내용을 분배했다. 특히 본 수업에서 교수자가 심화 내용을 강의 기반으로 설명한 후, 학생들이 개별로 문제를 풀어보도록 하며 교수자가 도움을 주어 어려운 내용을 익힐 시간을 충분히 주었다. 이후에 팀 활동을 통해 배운 내용을 응용해 보도록 했다. 3.3.1장에서는 수업 내용을 어떻게 사전학습과 본 수업으로 분배하는지에 대한 일반적인 기준을 예제와 함께 제시한다.

플립러닝을 적용한 많은 기존 연구에서는 또한 사전학습 자료로 외부에서 만든 영상과 문서를 사용했는데 [2,3,9], 이에 따라 본 수업 내용과 완벽히 연계되지 않아 사전학습의 효과가 떨어지는 단점이 있었다 [10,11]. 이를 보완하고자 자체 제작한 영상을 사용한 연구도 있었으나 [4,5,7], 사전학습을 올바로 이해했는지 확인하기 위한 퀴즈가 없거나 혹은 이를 본 수업에 와서 보게 되므로 사전학습을 잘 이해하고 본 수업에 참석하기에 어려움이 있었다. 본 연구에서는 수업 내용과 잘 연계되도록 자체 제작한 영상과 문서를 사전학습 자료로 활용했다. 또한 이를 학생 스스로 올바르게 이해하고 본 수업에 참여할 수 있도록 사전학습과 함께 풀어보는 퀴즈에서 오답에 대한 피드백이 즉시 제공되도록 준비하였고, 만점이 아닌 경우 부족한 부분을 공부해 다시 응시할 수 있도록 했다.

본 연구와 유사하게 본 수업에서 강의와 활동을 조합해 수행한 연구도 있었다 [12]. 이 연구는 수치해석 과목에 2개 학기에 걸쳐 플립러닝을 적용했는데, 일주일 두 번 수업 중 한 번은 사전학습과 활동 중심으로 수행하고, 다른 한 번은 전통적인 강의와 실습을 수행했다. 또한 사전학습 자료도 수업 내용과 연계해 직접 제작했는데, 전반적으로 긍정적인 효과를 얻었다. 본 연구는 매 수업에서 사전학습과 강의, 팀 활동을 적절히 조합해 수행했다.

(표 1) 플립러닝을 적용한 교과목 개요
(Table 1) Overview of flipped-learning courses

ID	과목명	수업 내용과 특성	플립러닝을 적용하지 않은 학기	플립러닝을 적용한 학기
1	이산수학 (1학년)	논리, 증명 방법, 함수, 그래프, 확률 등 컴퓨터학에 필요한 기초 수학을 익히는 이론 과목	3학기 (2011~2013년 가을) 3분반 (총 143명, 평균 48명)	1학기 (2014년 가을) 1분반 (총 39명)
2	데이터 통신 및 네트워크 (2학년)	OSI 7계층과 컴퓨터망의 동작 원리를 익히는 이론+실습 과목	3학기 (2012~2014년 봄) 3분반 (총 152명, 평균 51명)	4학기 (2015~2017, 2019년 봄) 7분반 (총 402명, 평균 57명)
3	네트워크 보안과 프로그래밍 실습 (3학년)	스니핑과 스푸핑 등 네트워크를 통한 공격의 원리를 익히는 이론+실습 과목	2학기 (2012~2013년 가을) 4분반 (총 85명, 평균 21명)	4학기 (2014~2017) 8분반 (총 251명, 평균 31명)
4	침입탐지와 차단 시스템 (3학년)	방화벽과 침입탐지 시스템 등 공격에 대한 방어 기법을 익히는 이론+실습 과목	3학기 (2012~2014년 봄) 5분반 (총 111명, 평균 22명)	4학기 (2015~2017, 2019년 봄) 7분반 (총 270명, 평균 39명)

또한 4개 과목에 대해 1~4개 학기 동안 (총 23개 분반, 수강생 962명) 플립러닝을 적용한 결과를 분석하여 통계적으로 유의미한 결과를 얻고자 했다.

3. 플립러닝을 적용한 교과 운영

3.1 플립러닝 적용 교과목 소개

본 연구에서는 표 1과 같은 4개 과목에 플립러닝을 적용하였다. 이들은 서울 소재 중소 규모 대학교의 컴퓨터 관련 학과에서 개설하는 전공 선택 과목으로, 한 주 3시간 수업으로 구성된 3학점 과목이다. 이 중 과목 1은 컴퓨터학에 필요한 기초 수학을 학습하는 이론 과목으로, 강의를 통해 이론을 배우면서 배운 내용에 관한 문제를 교수자와 함께 풀어보는 방식으로 진행한다. 과목 2~4는 컴퓨터망과 정보보호에 대해 배우는 과목으로 강의를 통한 이론 학습과 이를 활용해 보는 실습을 병행하며 진행한다.

본 연구에서는 위 4개 과목에 대해 2014~2015년부터 플립러닝을 적용하기 시작했으며, 기존의 적용 결과에서 보고된 바와 같이 학습자가 지식을 더 잘 이해하고 활용할 수 있는 수업을 목표로 했다 [1]. 특히 플립러닝을 적용하기 전의 2~3개 학기와 플립러닝을 적용한 후의 1~4개 학기를 비교해 수업 방식이 어떻게 달라졌으며 학생들의 학업 성취도와 만족도는 어떻게 변화했는지를 분석했다.

3.2 플립러닝 적용 방법 개요

그림 1은 (a) 플립러닝을 적용하지 않은 수업 진행 방식과 (b) 적용한 진행 방식을 보여준다. 플립러닝을 적용하지 않았을 때는 교수자가 이론을 강의하며 중간중간 배운 내용에 대한 문제를 함께 풀어보거나 실습하는 방식으로 수업을 진행했다.

플립러닝을 적용한 수업에서는 먼저 ① 수업 전날까지 학생들이 수업 내용을 이해하는데 필요한 기초 지식을 예습 자료를 통해 학습하도록 했다. 사전학습 후에는 이에 대한 온라인 퀴즈를 풀도록 했는데, 잘 이해하지 못한 내용은 보완 학습할 수 있도록 오답에 대한 피드백을 즉시 제공하며 3회까지 다시 응시할 기회를 주었다.

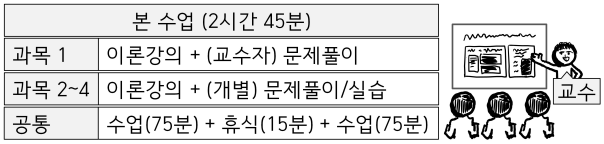
다음으로 ② 본 수업에서는 사전학습 내용을 심화한 내용과 이를 응용하는 방법을 강의하되 이를 실제 문제 풀이에 활용해 볼 수 있도록 중간중간 문제를 풀거나 실습하도록 했다. 이러한 문제 풀이와 실습은 개별 혹은 팀으로 진행했는데, 교수자가 진행 과정을 관찰하며 도움을 주되 서로 다른 해결 방법을 선정해 활동 후 모든 학생과 공유하도록 했다.

마지막으로 사전학습을 통해 40분 정도의 여유 시간을 얻을 수 있었는데, 이를 ③ 보충 시간으로 활용하여 수업 중 궁금하거나 어려운 점이 있었던 학생들이 자유롭게 교수자에게 질문하며 도움을 받도록 하였다. 특히 수업을 힘들게 따라오는 학생들이 이 시간을 편하게 활용하도록 격려했다. 이어지는 3.3~3.5장에서 단계 ①~③ 각각에 대해 자세히 기술한다.

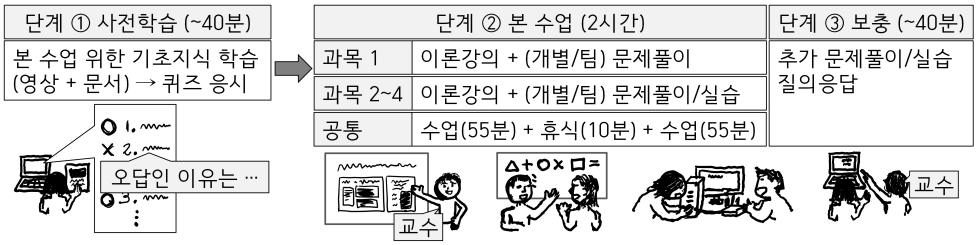
3.3 단계 ① 사전학습

3.3.1 사전학습과 본 수업 내용의 구분

사전학습 자료를 준비하기 위해서는 먼저 매주 학생들이 배워야 할 내용을 정하고 이를 사전학습과 본 수업 용



(a) 플립러닝을 적용하지 않은 수업 진행 방식



(b) 플립러닝을 적용한 수업 진행 방식

(그림 1) 플립러닝 적용 전과 후의 수업 진행 방식

(Figure 1) Before and after implementing flipped-learning

도로 나누어야 한다. 이 연구에서는 사전학습을 통해 정해진 학습 목표를 보다 잘 이해하고 활용할 수 있도록 하는 것을 지향했다. 이를 위해 본 수업에서는 더 중요하면서 심화한 내용을 배우도록 하고, 사전학습에서는 이를 배우기 위한 기초 지식을 익히도록 하였다. 이렇게 구분한 이유는 (가) 사전학습은 교수자의 도움 없이 진행하므로 스스로 이해할 수 있는 난이도여야 하며, (나) 본 수업에서는 심화한 내용에 주의를 집중함으로써 이를 더 잘 이해하도록 하기 위해서이다.

표 2는 앞에서 기술한 기준에 따라 과목 1과 4의 한 주차 수업 내용을 사전학습과 본 수업으로 나눈 예를 보여준다. 과목 1은 이론 과목으로 논리적 추론을 배우는 것이 한 주차의 목표이다. 이를 위해 사전학습에서는 추론에 사용되는 기초 개념인 명제와 진리표의 정의를 익히고 이들의 다양한 예를 보도록 했다. 본 수업에서는 이러한 기초 개념을 활용해 복잡한 명제의 진리표를 만들고 추론하는 방법을 배우며, 이를 여러 번의 문제 풀이를 통해 체득하도록 했다. 과목 4는 이론과 실습을 병행하는 과목으로 리눅스 방화벽의 동작 원리를 이해하고 설정하는 방법을 익히는 것이 한 주차의 목표이다. 이를 위해 사전학습에서는 리눅스 방화벽의 기본 동작 원리와 설정에 필요한 명령의 문법을 익히도록 했다. 본 수업에서는 실제 요구조건이 제시되었을 때 (예: 외부망에서 내부망 A의 웹서버로 접근을 허용하고자 함) 이러한 명령을 어떻게 선정하고 조합해 원하는 규칙을 구현하는지를 배우며, 배운 방법을 여러 다른 상황에 적용해 실습해보며 체득하도록 했다.

(표 2) 사전학습과 본 수업 내용의 구분 예

(Table 2) Examples of separation between pre-class and in-class activities

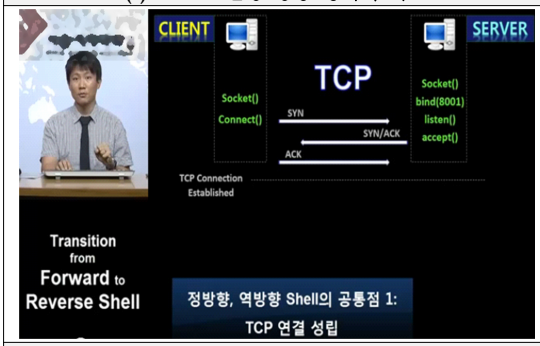
과목 1 목표	명제를 사용해 논리적인 추론하는 방법 이해
사전학습 내용	**본 수업 내용**
- 명제와 진리표의 정의 - 자주 사용하는 명제와 진리표 (예: 부정, and, or, xor, 조건, 대우, 동치)	- 복합 명제 진리표 작성법 - 진리표 사용한 추론 방법 - 추론 규칙과 사용 방법 (예: 긍정 논법, 소거)
과목 4 목표	리눅스 방화벽의 동작법과 설정 방법 이해
사전학습 내용	**본 수업 내용**
- 리눅스 방화벽을 구성하는 5개 chain과 4개 table - 방화벽 규칙 설정 명령의 문법과 의미 (예: iptables) - 방화벽 설정 시 필요한 리눅스 명령의 문법과 의미 (예: netstat, ifconfig)	- 보안 상황에 맞게 chain과 table을 선정하는 방법 - 선정한 chain과 table에 규칙을 설정하고, 순서를 정하고, 검증하는 방법 - 다양한 상황 제시하고 규칙 설정해보는 실습

3.3.2 사전학습 자료와 온라인 퀴즈 준비

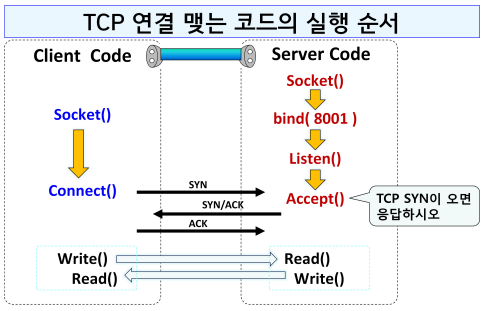
3.3.1장과 같이 사전학습 내용을 정한 후에는 사전학습 자료를 제작했다. 표 3은 과목 3에 대한 사전학습 자료의 예를 보여준다. 사전학습 자료로는 (a) PPT 형식 자료와 함께 이를 설명하는 영상을 제공하거나 혹은 (b) PPT 자

(표 3) 사전학습 자료의 예 (과목 3)
(Table 3) Examples of pre-class materials (Course 3)

(a) PPT + 설명 영상 형식의 자료



(b) 문서 형식의 자료



[네트워크 보안과 프로그래밍 실습 4주 예습자료: 역방향 커맨드 쉘]

클라이언트와 서버가 TCP를 사용해 연결을 맺고 데이터를 주고받는 단계를 정리해 보면 다음과 같다.

(1) 먼저 서버를 실행해 클라이언트로부터의 연결을 기다리고 있어야 한다. 만약 서버가 실행되지 않은 상태에서 클라이언트가 연결을 시도하면 연결받을 대상이 없으므로 연결이 이루어지지 않는다. 서버 코드에서는 연결을 기다릴 소켓을 만들고 (**socket()**), 지정한 포트로 도착한 연결을 받도록 설정하며 (**bind()**), 연결에 필요한 메모리를 준비한 후 (**listen()**), 설정한 포트로 도착한 연결 시도가 있을 때까지 대기 상태에 들어간다 (**accept()**).

(2) 다음으로 클라이언트를 실행하면 서버와 연결을 시도한다. 특히 연결을 맺을 소켓을 만들고 (**socket()**), 지정한 IP 주소와 포트로 연결을 시도하는데 (**connect()**), 특히 TCP SYN 패킷을 전송한다.

(3) 서버가 클라이언트로부터 온 SYN 패킷을 받으면 대기 상태에서 깨어나면서 클라이언트에게 SYN/ACK을 보내 응답하며, 다시 클라이언트로부터 ACK 패킷을 받으면 TCP 연결이 성립된다.

(4) 이렇게 연결이 성립한 후에는 연결의 양 끝을 의미하는 소켓에 **write()** 명령을 사용해 데이터를 전송하고 **select()**와 **read()** 명령을 사용해 받은 데이터가 있을 때 읽어오면서 양방향으로 신뢰적으로 데이터를 주고받는다.

(c) 온라인 퀴즈와 보기별 피드백

[Q] TCP 연결을 맺는 과정에 대한 설명으로 올바른 것은?

- 클라이언트는 socket(), bind(), listen(), accept() 순으로 실행 후 대기한다.
  [오답 피드백] 보기는 서버에 대한 설명이며, 클라이언트는 socket(), connect()를 실행
- 클라이언트와 서버 중 어느 쪽이 먼저 실행되더라도 연결 성립이 가능하다.
  [오답 피드백] 서버가 먼저 실행되어 클라이언트의 연결을 기다려야 함
- SYN, SYN/ACK, ACK 패킷을 주고받음으로써 연결을 성립한다.
  [정답 피드백] 올바른 설명이며, 이 과정을 3-way handshake라 함
- SYN을 먼저 전송한 쪽이 서버이다.
  [오답 피드백] SYN을 먼저 보낸 쪽이 클라이언트임

료 아래에 이를 설명하는 내용을 문서 형식으로 포함해 제공했다. 플립러닝을 적용한 첫 2개 학기에는 (a)와 같이 영상을 사용하였는데, 이 방법은 학기마다 새로운 내용을 반영해 영상을 수정 편집하는 데 시간이 많이 소요되었다. 이러한 부담을 줄이고자 3번째 학기부터는 (b)와 같이 문서를 사용하는 방법을 시도해 보았는데, (a) 방식보다 편집 시간은 적게 소요되면서 유사한 학습 효과를 볼 수 있었다. 4장에서는 두 방식의 학습 효과를 더 자세히 비교하고 분석한다. 사전학습 자료는 학습하는데 40분 내외의 시간이 걸리도록 준비했다. 특히 3~5개의 주제로 나누어 (a) 주제별로 5~7분 길이의 영상을 제공하거나 (b) 15장 내외의 문서를 제공했다.

사전학습을 수행해야만 본 수업을 잘 이해할 수 있으므로 학습 후에는 퀴즈에 응시하도록 하고 그 결과를 성적에 반영했다. 퀴즈는 강의지원시스템에서 온라인으로 응시하도록 했으며, 주제별로 최소 하나의 문제를 출제했다. 특히 사전학습 자료를 최대한 잘 이해하고 본 수업에 참여할 수 있도록 퀴즈 응시 후에는 표 3(c)와 같이 오답에 대한 피드백이 즉시 제공되도록 했으며, 부족한 부분을 다시 공부해 3회까지 재응시할 수 있도록 했다.

3.4 단계 ② 본 수업

본 수업에서는 사전학습을 통해 배운 기본 개념을 활용하고 심화한 주제에 대해 배우며, 또한 이를 실제 문제 풀이에 적용하는 방법을 익히는 것을 지향했다. 이를 위해 그 주에 배울 심화 주제 각각에 대해 먼저 관련된 사전학습 내용을 빠르게 복습한 후, 심화 주제에 대해 강의

하고, 이어서 문제를 풀어보거나 실습함으로써 실제 문제 풀이에 배운 내용을 바로 적용해 보도록 했다. 문제 풀이 중에는 교수자에게 사소한 질문이라도 편하게 하도록 격려했다. 또한 학생들의 문제해결 과정을 관찰해 서로 다른 해결법을 찾은 학생을 알아 두었다가 발표를 통해 다른 학생들과 공유하도록 했다. 발표할 학생에게는 발표 전에 미리 동의를 구했으며, 가능하면 기존에 발표를 안 해보았던 학생이 발표하도록 함으로써 기회가 고르게 돌아가도록 했다. 쉬운 문제는 개인별로, 어려운 문제는 팀으로 풀어보도록 했는데, 팀 활동의 경우 같은 책상을 공유한 2~5명이 한 팀이 되도록 했다.

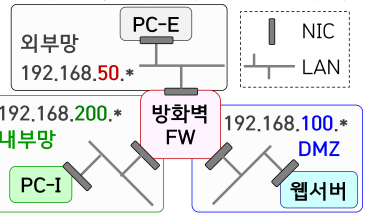
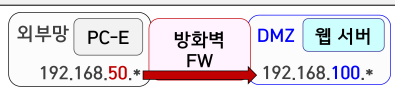
표 4는 표 2의 두 번째 예제 (과목 4)에 대한 본 수업 진행 과정을 보여준다. 이 주차의 목표는 리눅스 방화벽의 동작 원리와 설정 방법을 익히는 것이다. 이를 위해 사전학습에서 방화벽의 기본 동작 원리와 설정 명령의 형식을 익히도록 했으며, 본 수업에서는 여러 다른 상황에 대한 요구조건이 제시되었을 때 이러한 명령을 어떻게 조합해 원하는 설정을 구현하는지를 배우도록 했다. 학생들이 사전학습에서 기본 지식을 익히고 본 수업에 참여했기에 심화 내용을 익히는데 더 많은 시간을 할애할 수 있었으며, 다양한 실제 상황을 제시하고 (상황 1~4) 방화벽을 설정해 볼 수 있었다. 또한 학생들이 심화 내용을 잘 이해하여 여러 방법으로 설정을 시도하고 결과를 공유하는 모습도 볼 수 있었다. 이에 반해 플립러닝을 적용하지 않았을 때는 기본 개념을 강의하는데 40분 이상이 소요되었고, 처음 보는 내용이기에 이어지는 심화 내용을 이해하는 데도 더 오랜 시간이 걸렸다. 따라서 실습도 2가지 상황에 대해서만 수행할 수 있었다.

3.5 단계 ③ 보충 수업

사전학습이 학생들에게 추가적인 부담이 될 수 있기에 [2] 이러한 부담을 줄이고자 사전학습 시간과 (40분 이상) 본 수업 시간을 (2시간) 더한 시간이 플립러닝을 적용하지 않은 수업 시간과 (2시간 45분) 유사하도록 수업을 설계하였다. 이에 따라 본 수업을 마친 후 약 40분 정도는 강의실을 계속 사용할 수 있었는데, 이 시간을 보충 수업 시간으로 활용했다.

보충 수업 시간에는 교수자가 강의실에 남아 있으면서 학생들이 자유롭게 질문할 수 있도록 했다. 또한 본 수업 중에 해결하지 못한 문제나 실습이 있는 경우에도 계속 진행하며 교수자의 도움을 받도록 했다. 특히 문제 풀이 시간에 자주 어려움을 겪는 학생들이 이 시간을 적극적

(표 4) 본 수업 진행 과정의 예 (과목 4)
(Table 4) Examples of in-class lectures (Course 4)

<table>
<tr><th>시간</th><th>내용</th></tr>
<tr><td>10분</td><td>- 방화벽 설정에 필요한 리눅스 명령 복습하고 이를 사용해 네트워크 연결 상태 확인
- 리눅스 방화벽을 구성하는 chain과 설정 명령 복습하고 이를 사용해 현재 설정된 규칙 확인
 
</td></tr>
<tr><td>15분</td><td>- input/output chain이 사용되는 다양한 상황 학습
- accept/drop/log action이 사용되는 상황 학습
- ping과 ssh의 동작 원리 학습</td></tr>
<tr><td>15분</td><td>- (상황 1) 방화벽이 ping에 응답하게 설정하는 실습. 교수자가 시범 보이고 개별 실습하며 도움 줌</td></tr>
<tr><td>15분</td><td>- (상황 2) 내부망에서 방화벽에 ssh 접속하게 설정하는 실습. 2인 1팀으로 실습하고 결과 공유</td></tr>
<tr><td>10분</td><td>- 휴식</td></tr>
<tr><td>15분</td><td>- forward chain이 사용되는 다양한 상황 학습
- 웹서버의 동작 원리 학습</td></tr>
<tr><td>15분</td><td>- (상황 3) 외부망에서 DMZ의 웹서버에 접속하도록 설정하는 실습. 교수자 시범 후 개별 실습</td></tr>
<tr><td>20분</td><td>- (상황 4) 내부망에서 임의의 주소에 접근하도록 설정하는 실습. 2인 1팀으로 실습하고 결과 공유</td></tr>
<tr><td>5분</td><td>- 이번 시간에 배운 주요 내용 정리</td></tr>
<tr><td colspan="2">팀 활동을 통해 얻은 서로 다른 해법을 공유한 예</td></tr>
<tr><td><팀 1></td><td> 

① 외부 → DMZ 연결 시작 패킷 허용 규칙을 1번에 추가
iptables -A FORWARD -s 192.168.50.0/24 -d 192.168.100.0/24 -p tcp --dport 80 -j ACCEPT
② 기존에 허용한 연결에 속한 이후 패킷도 허용
iptables -A FORWARD -m conntrack --ctstate established -j ACCEPT</td></tr>
<tr><td><팀 2></td><td>

①에서 IP 대역 대신 NIC 기준으로도 규칙 설정 가능
iptables -A FORWARD -i eth2 -o eth0 -p tcp --dport 80 -j ACCEPT
'netstat -nr'이나 'ip route' 으로 확인해 보면 외부망에 속하는 모든 IP 대역은 eth2에, DMZ에 속하는 모든 IP 대역은 eth0에 연결되어 있기 때문</td></tr>
</table>

으로 활용하도록 격려했는데, 매시간 잘 이해하지 못한 부분에 대해 하나 이상의 질문을 하도록 독려했다.

그 결과 학생들이 점차 더 많은 질문을 편하게 하는 것을 볼 수 있었으며, 또한 최하위권 학생의 수나 중도에 포기하는 학생의 비율도 줄어드는 것을 관찰할 수 있었다. 그리고 교수자 입장에서는 학생들의 자유질문을 통해 수업에서 잘 이해하기 어려워하는 부분을 파악하고 이를 이어지는 수업에 반영해 개선하는 효과가 있었다.

4. 플립러닝 적용 결과 분석

4.1 결과 분석 방법

본 연구에서는 표 1의 4개 과목에 대해 플립러닝을 적용하기 전의 2~3개 학기와 (총 15 분반, 491명) 플립러닝을 적용한 후의 1~4개 학기를 비교해 (총 23 분반, 962명) 학생들의 학업 성취도와 수업에 대한 만족도가 어떻게 달라졌는지를 비교 분석했다. 먼저 학업 성취도를 분석하기 위해서는 중간시험과 기말시험의 성적 분포가 어떻게 변화했는지를 관찰했는데, 시험은 유사한 난이도로 출제했다. 또한 플립러닝을 적용한 학기에는 사전학습과 퀴즈를 수행한 학생의 비율 및 퀴즈 점수의 분포도 함께 분석했다. 다음으로 수업에 대한 만족도를 분석하기 위해서는 학기 말에 설문조사를 실시했다. 설문조사 문항은 표 5와 같이 5점 척도의 질문 5문항과 주관식 2문항으로 구성했다. 객관식 문항을 통해서는 플립러닝 적용 전후의 만족도를 정량적으로 비교하고자 했으며, 주관식 문항을 통해서는 이러한 만족도에 대한 이유를 파악하고자 했다.

4.2 분석 결과

4.2.1 플립러닝 적용 전후 시험 점수 분포

플립러닝을 적용하지 않은 학기와 적용한 학기 간 시험 성적의 차이가 있는지 확인하기 위해 통계적 방법인 *t*-test를 사용했다. 이를 위해 null hypothesis H_0로는 "두 집단의 성적은 유의미한 차이가 없다"를 사용했으며, alternative hypothesis H_a로는 "플립러닝을 적용한 학기에 성적이 더 향상되었다"를 사용했다. *T*-test는 *p*-value를 결과로 내는데, 이 값은 H_0를 기각하는 유의 수준 α의 최솟값을 의미한다. *P*-value가 작을수록 H_0를 기각하고 H_a를 인정할 근거가 더 확실해진다.

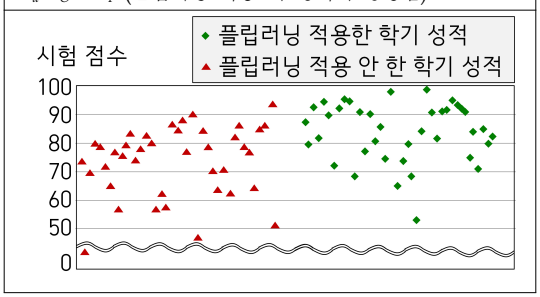
(표 5) 수업 만족도 분석을 위한 설문 문항
(Table 5) Questions for analyzing student satisfaction levels with class lectures

ID	유형	문항
1	객관식 (1~5점)	담당 교수는 수업 내용을 충실하게 구성했다.
2		수업 내용이 체계적으로 전달되었다.
3		수업 방식은 수업 내용과 잘 부합되어 학습 내용의 이해에 도움이 되었다.
4		담당 교수는 학생들의 참여와 관심을 유발하기 위해 학생들과 상호작용하며 강의를 진행했다.
5		이 수업은 관련 지식의 확충에 도움이 되었다.
객관식 1~5점 의미: 1 전혀 그렇지 않다, 2 그렇지 않다, 3 보통이다, 4 그렇다, 5 매우 그렇다		
6	주관식	이 수업에서 좋았던 점은 무엇입니까?
7		이 수업에서 개선해야 할 점은 무엇입니까?

표 6은 *t*-test에 사용한 표본, 가설 및 검정 결과를 보여준다. 네 과목 모두에서 *p*-value는 0.05보다 작게 나왔고, 따라서 유의 수준 α= 0.05에서 H_0는 기각되고 H_a가 인정된다. 이는 95% 이상의 확률로 플립러닝을 적용한 학기에 성적이 더 향상되었음을 의미한다. 표 6 아래에서는 과목 1에 대해 플립러닝을 적용하지 않은 한 학기와 플립러닝을 적용한 한 학기의 성적 분포를 scatter plot으로 보여준다. 플립러닝을 적용하지 않은 학기와 비교하면 플립러닝을 적용한 학기에는 하위권 학생의 비중이 줄어들고 중상위권 학생의 비중이 증가한 것을 볼 수 있다. 다른 3개 과목에서도 비슷한 분포를 볼 수 있었다.

(표 6) 시험 성적에 대한 T-test 결과
(Table 6) T-test results for exam scores

과목 ID	플립러닝 적용 안 한 학기 성적 s_O (μ, σ)	플립러닝 적용한 학기 성적 s_F (μ, σ)	*p*-value
1	(74.52, 12.59)	(84.34, 10.46)	$p < 0.0001$
2	(76.90, 15.52)	(79.92, 11.52)	$p = 0.0131$
3	(78.62, 13.15)	(82.36, 12.81)	$p = 0.0214$
4	(70.77, 17.10)	(75.07, 11.13)	$p = 0.0039$
H_0: $s_O = s_F$ (플립러닝 적용 전후 성적의 차이 없음) H_a: $s_O < s_F$ (플립러닝 적용 시 성적이 향상됨)			



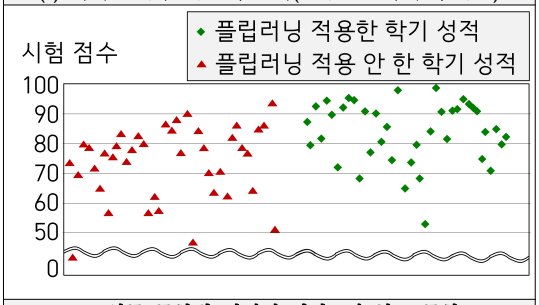
플립러닝을 적용한 학기 중에는 매주 1~2명의 학생 외에는 거의 퀴즈에 응시하였고, 즉시 제공되는 피드백 및 3번의 재응시 기회를 통해 대부분 사전학습 내용을 이해하여 퀴즈에 만점을 받고 본 수업에 참여했다. 사전학습 자료로 영상을 사용한 학기와 문서를 사용한 학기 간에도 시험 성적의 분포를 비교해 보았으나, 통계적으로 유의미한 차이는 볼 수 없었다. 즉, 문서를 사용해도 영상을 사용한 경우와 비슷한 사전학습 효과를 볼 수 있음을 알 수 있었다.

4.2.2 설문조사 결과

표 5의 설문조사 결과는 표 7과 같다. 먼저 표 7(a)는 설문 문항 1~5의 평균 점수가 학기별로 어떻게 변해가는지를 보여주는데, 4개 과목 모두에서 플립러닝을 적용하면서 4.7 이상으로 0.1~0.2점 이상 향상되었음을 알 수 있다. 표 7(b)는 이러한 향상 정도를 설문 문항별로 세분화해서 보여주는데, 수업 내용의 충실도와 (문항 1) 체계적 전달은 (문항 2) 소폭 상승하였고 수업 방법과 (문항 3) 학생 참여 및 교수자와의 상호작용 (문항 4), 그리고 지식 확충에 대해서는 (문항 5) 더 큰 폭으로 증가한 것을 볼 수 있었다. 이는 (가) 사전학습 후 심화학습을 진행하는 수업 방식 및 (나) 교수자와 함께하는 문제 풀이 후 결과를 공유하는 과정이 이해도 향상에 도움이 되었음을 뜻한다. 표 7(c)에서 보여주는 주관식 문항에 대한 답안에서도 유사한 이유를 발견할 수 있었다. 또한 자유롭게 질문을 하는 보충 수업 시간을 통해서도 중하위권 학생들이 도움을 받았음을 알 수 있었다.

(표 7) 수업 만족도에 대한 설문조사 결과
(Table 7) Questionnaire results on student satisfaction levels with class lectures

(a) 학기별 평점 평균의 변화(문항 1~5 점수의 평균)



(b) 설문 문항별 평점의 변화(5점 척도 문항)

문항	플립러닝 적용 안 한 학기 평균	플립러닝 적용한 학기 평균 (증가량)
1. 내용의 충실성	4.78	4.92 (+0.14)
2. 내용의 체계적 전달	4.69	4.90 (+0.21)
3. 수업 방식의 적절성	4.60	4.86 (+0.26)
4. 참여와 상호작용	4.61	4.88 (+0.27)
5. 지식 확충에 도움	4.61	4.86 (+0.25)

(c) 주관식 문항에 대한 대표 답안(플립러닝을 적용한 학기)

문항	대표 답안
6. 좋은점	- 사전학습과 본 수업에서 개념을 두 번 반복해 보므로 더 잘 이해했으며 저절로 공부하게 되는 과목이었음. 사전학습 시 각자 이해하는 속도에 맞게 공부할 수 있어 좋았음 - 수업에서 배운 내용을 바로 이어서 활용해 보아서 지식 확장에 도움 되었고 이론과 실습이 (혹은 문제 풀이가) 잘 조화됨 - 팀별 활동과 활동 결과 공유로 여러 다른 해법 알 수 있어 좋았고 발표 통해 자신감 얻음. 수업이 활기차 재미있었음 - 교수님이 설명을 잘해주시고 질문에도 잘 답변해 주어 도움 됨 - 보충 질문 시간을 통해 어려운 내용 이해했고 따라오지 못하는 학생이 없도록 챙겨줌
7. 개선점	- 가끔 예습하는 것이 귀찮거나 수업량이 많아 힘들 때도 있었지만 잘 설명해 주셔서 이해하는 데 어려움이 없었음

5. 결 론

본 논문에서는 플립러닝과 강의형 수업을 적절히 혼용하는 방식을 제시했으며, 특히 기초 지식은 사전학습을 통해 학생 스스로 익히도록 하고 이를 심화, 응용한 지식은 본 수업에서 교수자와 팀 동료의 도움을 받아 익히도록 하는 방식을 제안했다. 제안한 방식은 컴퓨터학 4개 과목 23개 분반에 1~4개 학기 동안 적용해 보았다. 그 결과 학생들의 평균 시험 점수가 높아졌고 하위권 학생의 비율이 줄어들었으며 학생들도 수업을 통해 더 다양한 지식을 얻었다고 인지하는 것을 볼 수 있었다. 또한 플립러닝을 적용함으로써 변화한 수업 방식 및 교수자와의 상호작용 증가에 대해 대다수의 학생이 만족하는 것으로 나타났다.

하지만 수강생 수가 많아질수록 교수자가 직접 관찰을 통해 수업 내용을 이해하는 데 어려움을 겪는 학생을 찾고 도움을 주기에는 한계가 있다. 따라서 자동화된 시스템으로 수업 참여도를 분석해 이러한 학생을 판별한다면 (예: 사전학습 수행률, 퀴즈/과제 점수 등 분석 [13]) 더 많은 학생이 플립러닝의 장점을 누릴 수 있을 것으로 기대된다.

참고문헌(Reference)

[1] J. Lee, “A systematic review of flipped learning research in domestic engineering education,” Journal of Engineering Education Research, Vol. 24, No. 3, pp. 21-31, 2021. https://doi.org/10.18108/jeer.2021.24.3.21

[2] Y. Lee and S. Youn, “Reconstructing the meaning of flipped learning by analyzing learners’ experiences,” Journal of Engineering Education Research, Vol. 20, No. 1, pp. 53-62, 2017.
http://doi.org/10.18108/jeer.2017.20.1.53

[3] S. Jang, “Design of effective teaching-learning method in algorithm theory subject using flipped learning,” Journal of the Korean Institute of Information and Communication Engineering, Vol. 21, No. 5, pp. 1042-1048, 2017.
http://doi.org/10.6109/jkiice.2017.21.5.1042

[4] S. Lee, “A case study of flipped learning in algorithm class,” Proceedings of the Korea Society of Computer and Information Winter Conference, pp. 183-186, 2016. https://www.dbpia.co.kr/Journal/articleDetail?nodeId=NODE06597494

[5] O. Kim, Y. Cho, E. Kim, and Y. Lee, “A case study of flipped learning at an engineering college: focusing on thinking, creativity, and problem-solving skills,” Journal of Education and Culture, Vol. 26, No. 4, pp. 187-212, 2020.
https://doi.org/10.24159/joec.2020.26.4.187

[6] K. Kim, J. Kim, and Y. Ahn, “Design of programming course based on flipped learning in a non-face-to-face environment,” Proceedings of the Korea Society of Computer and Information Winter Conference, pp. 301-302, 2021.
https://koreascience.kr/article/CFKO202121751497312.page

[7] E. Kim, “Application of flipped learning in database course,” Journal of the Korean Institute of Information and Communication Engineering, Vol. 20, No. 4, pp. 847-856, 2016.
http://doi.org/10.6109/jkiice.2016.20.4.847

[8] S. Pi, S. Do, “The effectiveness of the flipped learning using the smart device,” Journal of Digital Convergence, Vol. 15, No. 4, pp. 65-71, 2017.
http://doi.org/10.14400/JDC.2017.15.4.65

[9] Y. Choi, “Design and application of term project model for game mathematics in flipped learning environments,” Journal of Korea Multimedia Society, Vol. 20, No. 7, pp. 1102-1112, 2017.
http://doi.org/10.9717/kmms.2017.20.7.1102

[10] J. Leem, “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. https://doi.org/10.17232/KSET.32.1.165

[11] H. Han, C. Lim, and S. Han, “Instructional strategies for integrating online and offline models of flipped learning in higher education,” Journal of Educational Technology, Vol. 31, No. 1, pp. 1-38, 2015. https://doi.org/10.17232/KSET.31.1.001

[12] H. Jang, “A study on the numerical analysis classes using the flipped learning teaching method,” Journal of Internet Electronic Commerce Research, Vol. 20, No. 1, pp. 117-130, 2020.
https://doi.org/10.37272/JIECR.2020.02.20.1.117

[13] S. Pi, “Learning effects of flipped learning based on learning analytics in SW coding education,” Journal of Digital Convergence, Vol. 18, No. 11, pp. 19-29, 2020.
http://doi.org/10.14400/JDC.2020.18.11.019

◐ 저 자 소 개 ◑

이 시 형(Sihyung Lee)

2000년 KAIST 전자전산학과 졸업(공학사)
2004년 KAIST 전자전산학과 졸업(공학석사)
2010년 Carnegie Mellon University 전자컴퓨터공학과 졸업(공학박사)
2010년~2011년 IBM TJ Watson 연구소(박사후연구원)
2011년~2019년 서울여자대학교 정보보호학과 교수
2019년~현재 경북대학교 컴퓨터학부 교수
관심분야: 프로그램 자동 생성, 인터넷 의견분석, 컴퓨터 네트워크
E-mail : sihyunglee@knu.ac.kr