

게임 요소 기반의 객체지향 개념 학습에 대한 수업 사례 연구

김용천[†] · 장윤재[†] · 윤일규[†] · 김자미^{††} · 이원규^{†††}

요 약

프로그래밍 교육은 컴퓨터 과학의 원리를 활용하여 우리 생활 속의 문제를 해결하는데 도움을 준다. 객체지향 프로그래밍의 경우 우리가 주변에서 볼 수 있는 문제 또는 객체를 중심으로 프로그램을 작성하기 때문에 생활 속의 활용도를 높일 수 있다. 따라서 본 연구는 객체지향 개념을 친숙한 형태로 학습시키는 방안을 모색하기 위한 목적이 있다. 목적 달성을 위해 6명의 학생을 대상으로 실험 연구를 진행하였다. 연구 결과, 첫째 학생들이 친숙하게 사용할 수 있는 학습 도구가 필요하다는 것, 둘째 프로그래밍 구현 전에, 충분히 사고하고 개념을 습득할 필요가 있음을 발견하였다. 본 연구는 초보 학습자가 이해하기 어려워하는 객체지향 프로그래밍 학습에 도움이 되는 학습 방법을 제시했다는 데 의의가 있다.

주제어 : 객체지향 개념, 프로그래밍, 사례 연구

A Case Study on Course Game Based Elements for Learning Object-Oriented Concepts

YongCheon Kim[†] · YunJae Jang[†] · IlKyu Yoon[†] · JaMee Kim^{††} ·
WonGyu Lee^{†††}

ABSTRACT

Programming education helps learners solve real world problems by applying the principles of computer science. In particular, in the case of object-oriented programming centered around objects or problems that are observed in the real world, learners are able to make the most use of their programming skills in real life. Therefore, in this study, we sought ways to have learners grasp object-oriented concepts in a learner-friendly manner. To this end we conducted an experiment with six students as participants. As a result we could derive two main points. First, the learning tools which can be easily used by learners are required. Second, it is necessary to think thoroughly to learn concepts before implementation of the programming. This study has significance in the sense that it provides a learning method which helps a novice learner to learn the object-oriented programming that he or she feels difficulty to understand.

Keywords : Object-Oriented Concepts, Programming, Case Study

[†] 정 회 원: 고려대학교 컴퓨터교육학과 박사수료

^{††} 종신회원: 고려대학교 컴퓨터학과

^{†††} 종신회원: 고려대학교 컴퓨터학과(교신저자)

논문접수: 2014년 3월 10일, 심사완료: 2014년 4월 8일, 게재확정: 2014년 5월 12일

* 본 논문은 2013년 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(NRF-2013R1A2A2A03016926)

1. 서론

컴퓨터 과학의 원리는 교육 분야 뿐 아니라 우리의 생활 속에서 IT를 활용한 문제 해결을 가능하게 하였다[1]. 특히, 프로그래밍 교육은 알고리즘적 사고를 가능하게 하며 학생들의 창의력을 증진시키는데도 기여하였다[2][3].

프로그래밍 교육을 통한 효율적인 알고리즘 개발 능력 향상은 궁극적으로 우리에게 많은 시간과 노력을 절약할 수 있도록 한다. 따라서 최근에도 우리가 주변에서 볼 수 있는 물체 또는 생각할 수 있는 그 어떤 것인 객체를 중심으로 프로그램을 작성하는 객체지향 프로그래밍의 중요성이 커지고 있다[4].

2011년 고등학교 개정교육과정에서도 객체지향 프로그래밍 학습의 중요성을 강조하였다[5]. 개정교육과정은 일상생활에서 발생할 수 있는 사건이나 문제들을 객체지향적인 관점에서 생각하고 분석하며 설계할 수 있도록 하고 있다.

하지만, 객체지향 프로그래밍은 개념이 많고 복잡해서 초보 학습자가 이해하기에는 어려움이 많다[6]. 즉, Java나 C++와 같은 프로그래밍 언어는 전문가를 위해 개발된 것으로 초보 학습자가 객체지향 프로그래밍 학습을 하기에 적합하지 않다[7]. 어려운 명령어의 사용과 복잡한 문법은 초보 학습자의 인지적 부담을 가중시켜서 객체지향 학습을 통해 일상생활의 문제를 해결하기 위한 본래의 목적과 달리 오히려 학습자가 프로그래밍 자체를 싫어하게 하는 부작용이 발생할 수 있다[8]. 따라서, 프로그래밍 교육의 목적을 달성하기 위해서는 학습자가 친숙하게 익힐 수 있는 방법을 모색할 필요가 있다. 왜냐하면 학생들의 동기나 자기 효능감 등의 부족은 학습자와 프로그래밍 내용의 적합도가 낮을 경우 실패율이 높기 때문이다[9]. 학생들의 흥미나 동기를 높여 프로그래밍 학습을 진행한다던 객체지향 프로그래밍의 목적을 달성하는데 도움이 될 것이다.

이에 본 논문은 학령이 낮은 학습자의 객체지향 개념 학습을 위해 게임에 대한 활동지를 개발하여 프로그래밍 학습을 진행하였다. 본 연구에서 게임 요소는 게임을 내용으로 학습자의 흥미를 높일 수 있도록 하는 것으로 스토리와 흥미, 기술

/기계적 기능 등을 의미한다[10].

개발된 활동지는 학습자의 수준을 고려하면서 객체지향 프로그래밍의 요소인 클래스와 객체를 선언하고, 객체의 속성과 메소드를 정의할 수 있도록 구성하였다.

또한, 스크래치 2.0을 사용하여 자신이 생각한 객체지향 개념을 직접 구현해볼 수 있도록 하였다. 스크래치 2.0은 최근 전 세계적으로 가장 많이 활용되고 있는 교육용 프로그래밍 언어이며, 초보자도 자신의 아이디어를 쉽게 구현할 수 있는 환경이다. 특히 2.0버전으로 업데이트 되면서 사용자 정의 블록을 만들 수 있는 기능이 추가되어, 기본적인 객체지향 개념 표현이 가능하여 본 연구에서 활용하였다.

이상의 연구 진행을 통해 본 연구는 학생들의 흥미를 고려한 게임 요소 기반의 학습이 프로그래밍에 대한 인식 변화에 기여하는지를 확인하고자 한다. 분석을 통해 본 연구는 어떤 형태로 객체지향 수업을 진행해야 하는지에 대해 제안하기 위한 목적이 있다.

2. 관련연구

2.1 객체지향의 핵심 요소

객체의 개념이 처음 등장한 것은 1960년대 중반에 소개된 시뮬레이션 언어인 Simula를 통해서이다[11]. 객체라는 용어는 이전에도 인공지능 등 몇몇 분야에서 사용되었으나 클래스, 객체, 상속, 다형성, 캡슐화 등 객체지향의 개념은 이 시기에 소개되었다.

객체지향 개념이 소개되기 전에는 하위 프로그램들을 개발해 가면서 최종 시스템을 완성하는 방법을 사용하였다. 즉, 계획이 부족한 형태로 프로그램들간의 연결이 어렵게 되거나 모듈화가 되지 않아 중복 코딩 등의 문제가 제기되었다. 계획을 고려하고, 인간이 이해하는 방식을 시스템에 적용시키기 위해 객체지향 개념이 활용되기 시작하였다[12].

객체지향 프로그래밍은 컴퓨터 프로그램을 명령어의 목록으로 보는 것보다 문제의 대상들인 여러 개의 독립단위인 객체의 모임으로 파악하고

자 하는 것이다[9]. 객체는 객체를 설명하는 속성과 객체의 기능을 의미하는 메소드로 구성되어 있으며 각각의 객체는 메시지를 주고받으면서 데이터를 처리할 수 있다. 객체지향 개념을 이용하면 컴퓨터가 수행해야 하는 것을 하나하나 순서대로 지시할 필요가 없어 효율적이고 간단하게 문제를 해결할 수 있다[13].

객체지향 프로그래밍은 복잡하거나 방대한 양의 작업을 할 때 문제 구조를 쉽게 표현할 수 있다. 이처럼, 객체지향은 생산성과 재사용성, 유지 보수 등 효율성이 높기 때문에 다양한 산업현장에서도 각광받고 있다[9][14][15].

현실에서 중요성을 인정받은 객체지향 개념은 2007년 ‘정보’ 교과과정의 교육과정의 구성 내용이였다. 객체지향 프로그래밍 학습은 문제해결 방법과 절차영역에 포함되었으며 2011년 고등학교 개정 교육과정에서도 중요성이 강조되었다[5]. 하지만, 객체지향 개념은 추상적인 개념이 많아서 단기간에 학습하기에는 많은 어려움이 있다. 본 연구는 학령이 낮은 때부터 프로그래밍의 중요 내용인 객체지향의 기초적 이해를 바탕으로 학습을 진행할 수 있도록 하기 위하여 초등학생을 대상으로 객체지향에 대한 학습을 구상하였다.

객체지향 개념의 활용이 많아짐에 따라, 객체지향을 활용하기 위해서는 학습에서 핵심 요소를 정의할 필요가 있다. 많은 저널과 서적에서 객체지향의 필수요소에 대한 논의가 있으나 필요에 따라 사용하는 개념이 다르기 때문에 객체지향의 핵심요소를 규명하기는 어려웠다. Armstrong은 1966년부터 2005년까지 출간된 객체지향개발과 관련된 88개의 주요 저널, 잡지, 책 등을 분석하였다. 분석결과, 39개의 핵심 요소를 정의하였고 상위 10개의 객체지향 개념을 제시하면 <표 1>과 같다[16].

<표 1> 객체지향의 핵심 요소

| 개념 | 빈도수 | 비율 |
|--------|-----|-----|
| 상속 | 71 | 81% |
| 객체 | 69 | 78% |
| 클래스 | 62 | 71% |
| 캡슐화 | 55 | 63% |
| 메소드 | 50 | 57% |
| 메시지 전달 | 49 | 56% |

| | | |
|------|----|-----|
| 다형성 | 47 | 53% |
| 추상화 | 45 | 51% |
| 인스턴스 | 31 | 35% |

객체지향의 핵심요소에 대한 분석 결과, 상속(81%), 객체(78%), 클래스(71%), 캡슐화(63%), 메소드(53%) 등임을 알 수 있다.

본 논문은 학령이 낮은 학습자가 객체지향 개념 학습의 기본 개념으로 상속과 객체, 클래스를 선정하고, 기본 개념을 학습하는데 도움을 줄 수 있는 방법을 고려하였다.

2.2 객체지향 개념 학습을 위한 기존 연구

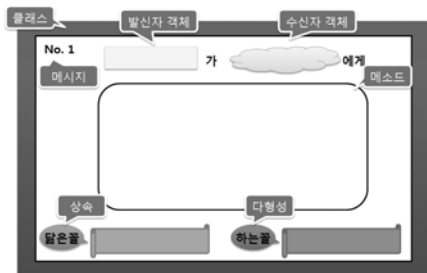
프로그래밍 교육을 성공적으로 이끌기 위해서는 프로그래밍 언어의 선택과 함께 학습자 수준에 맞는 다양한 교수학습방법이 제공될 필요가 있다[17][18]. 많은 연구자들은 학습자가 비교적 짧은 시간에 사용법을 익혀서 사용할 수 있는 교육용 프로그래밍 언어를 활용한 수업을 진행하였고, 학습자의 인지 수준에 맞는 다양한 교육 도구를 활용하였다. 프로그래밍 교육을 위한 기존의 연구는 다음과 같이 정리될 수 있다[2].

첫째, 객체지향형 교육용 프로그래밍 언어를 활용한 연구이다. 두리틀(Dolittle)은 객체지향 기반이면서 클래스나 상속 등과 같은 높은 수준의 개념에 대한 이해를 요구하지 않기 때문에 초중등 교육에서 많이 활용되고 있다[19][20]. 학습자의 인지 부담을 줄이기 위한 방법으로 클래스나 상속 등 어려운 개념이 배제된 객체지향 교육용 프로그래밍 언어인 Alice를 활용한 객체지향 개념 학습을 제안하기도 한다[21]. 학습자는 객체지향 개념을 직접 사용하지 않고도 프로그래밍 언어로 문제를 해결하는 과정에서 클래스나 상속의 개념을 자연스럽게 학습하기도 한다[7][22].

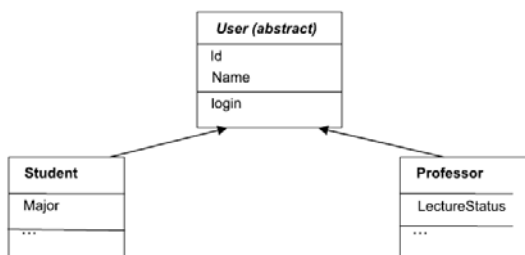
둘째, 학습자의 인지 수준에 맞는 교육 도구를 활용한 연구이다. 프로그래밍 활동은 문제해결 절차인 알고리즘을 설계하는 과정과 프로그래밍 언어를 사용하여 알고리즘을 구현하는 과정이 동시에 이루어지기 때문에 초보 학습자는 많은 인지적 부담을 갖게 된다[23][24]. 이에 학습자가 효과적으로 프로그래밍을 할 수 있도록 도움을 주기

위한 다양한 교육 도구가 활용되고 있다. Jeroen Keppen과 David Hay도 개념 지도(Concept mapping)을 사용하여 학습자가 만들고자 하는 것을 모델링할 수 있어야 하며 개념을 표현할 수 있도록 하였다[25]. 유사한 맥락으로 학령이 낮은 학생을 대상으로 자신이 만들고자 하는 내용을 우선적으로 작성하고, 프로그래밍 언어로 구현하는 과정을 연계할 수 있는 알고리즘 설계 활동지와[26], 자신이 무엇을 만들지 설계하는 성찰 활동지를 활용한 경우도 있었다[27].

특히, 객체지향 개념을 학습하는 과정에서 학습자의 흥미를 유발하기 위해 핵심 개념을 정의하고 추상화하는 경우도 있다. 학습자는 <그림 1>과 같은 객체지향 모델링 카드에 자신만의 스토리와 연관지어 객체, 메소드, 클래스를 작성함으로써 객체지향 개념을 처음 접하는 경우에도 쉽게 이해할 수 있다[28]. 또한, <그림 2>와 같이 추상화 클래스를 이용하여 클래스간의 공통된 속성과 메소드에 대한 개념을 학습하기도 한다[12].



<그림 1> 객체지향 모델링 카드



<그림 2> 추상화 클래스를 이용한 추상화

많은 연구들에서 효과적인 프로그래밍 수업을 위해 학습자의 수준에 맞는 다양한 프로그래밍 언어와 교수학습 방법을 제안하여 학습자가 프로그래밍 개념을 자연스럽게 학습할 수 있도록 하고 있다. 하지만, 비유나 활동을 통해 객체의 개

념을 학습하는 경우, 객체의 속성과 기능에 대한 개념 이해가 부족하여 개념보다 비유만 기억할 가능성이 많다[29]. 활동지를 활용하는 경우에도 프로그래밍 언어와 연계하여 구현해보지 않으면 이론 학습에 머물 수 있다.

학습자가 객체지향 프로그래밍을 하기 위해서는 우선적으로 객체의 개념에 대해 올바르게 이해하고 객체의 속성과 기능을 명확하게 정의할 수 있도록 도움을 줄 필요가 있다. 이에 본 논문은 학령이 낮은 학습자가 객체지향 개념 학습을 위해 객체에 대한 이해를 바탕으로 객체를 프로그래밍 언어로 구현하는 연구를 진행하였다.

3. 연구방법

3.1 연구 절차

본 연구는 수업에 참여하는 학습자의 내러티브를 텍스트화하여 크게 두 가지 형태로 분석하였다.

3.1.1 활동지와 대화를 통한 생각의 표현

수업 과정에서 학습자의 이해도는 학습자가 표현하는 활동과 언어, 표현된 내용을 통해 연구자가 판단할 수 있다. 따라서 연구자는 객체지향 개념 학습의 과정에서 학습자의 상태를 판단하기 위해 학습자들이 자신의 생각을 활동지와 언어를 통해 표현할 수 있도록 하였다. 그리고 해당 내용을 분석하였다.

3.1.2 활동지와 대화 내용 분석

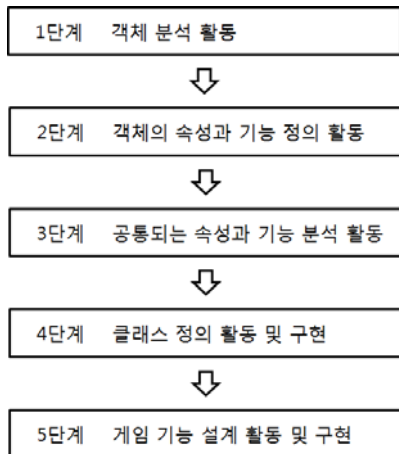
활동지는 학습 내용을 담는 그릇과 같다. 학생들이 작성한 활동지는 프로그래밍의 구현 과정에서 정확도를 판단할 수 있다. 본 연구자는 학습자가 필수적 내용을 작성하지 못한 경우는 수업 내용을 이해하지 못한 것으로, 올바르게 못하게 작성한 경우는 오개념을 갖는 것으로 판단하였다.

3.2 수업 설계

본 연구는 객체지향 개념을 학습하고 구현을

하는 과정에서 학습자에게 적합한 프로그래밍 학습을 제안하고자 하였다. 연구 목적 달성을 위해 초등학교 5-6학년 학습자 6명을 대상으로 진행되었다. 수업을 시작하기에 앞서 본 연구의 대상을 프로그래밍 경험과 객체지향 개념 학습의 경험이 전혀 없는 학생으로 한정하였다.

수업 설계는 <그림 3>과 같이 5단계로 구성되었다. 1단계는 게임에서 사용하고자 하는 객체를 분석하는 단계이다. 2단계는 객체의 세부 속성과 기능을 정의하는 단계이다. 3단계는 객체의 속성과 기능을 나열하여 공통되는 것을 표시하는 단계이다. 4단계는 객체의 공통 속성과 기능을 가진 클래스를 정의하는 단계이다. 5단계는 자신이 정의한 객체를 이용하여 어떻게 게임을 구성할지 계획하고 구현하는 단계이다. 구현은 스크래치 2.0을 활용하였다.



<그림 3> 수업 설계

수업 설계는 <표 2>와 같이 10차시로 구성하였다. 각 차시의 내용은 다음과 같다.

1차시에는 과일이나 핸드폰과 같은 주변의 객체를 찾고 객체를 특성과 기능으로 나누어 적어보는 수업을 진행하였다.

2차시에는 스크래치 웹사이트에서 제공하는 “벽돌깨기 게임”을 해보고 어떤 기능이 필요한지 스스로 분석해보는 수업을 진행하였으며 3차시에는 게임에 필요한 객체를 정의하는 수업을 진행하였다.

4차시에는 객체들의 공통되는 속성과 기능을

분석하는 수업을 진행하였다.

5차시에는 이를 바탕으로 공통되는 속성과 기능을 가진 클래스를 정의하는 수업을 진행하였다.

6차시에는 스크래치 2.0을 이용하여 도형을 만들고 이동시키는 것과 같은 기본 기능을 익히는 수업을 진행하였으며 7차시에는 스크래치 2.0에서 추가된 “추가 블록” 기능을 익히는 수업을 진행하였다.

8-10차시에는 객체를 구현하는 수업을 진행하였다.

<표 2> 객체지향 개념 수업 내용

| 차시 | 수업 내용 | 단계 |
|----|----------------------|-----|
| 1 | 우리 주변에서 객체 찾아보기 | 1단계 |
| 2 | 벽돌깨기 게임 분석하기 | |
| 3 | 벽돌깨기 게임의 객체 정의하기 | 2단계 |
| 4 | 객체의 공통되는 속성과 기능 분석하기 | 3단계 |
| 5 | 벽돌깨기 게임의 클래스 정의하기 | 4단계 |
| 6 | 스크래치 2.0 기본 기능 익히기 | 5단계 |
| 7 | 스크래치 2.0 추가 기능 익히기 | |
| 8 | 공 객체 구현하기 | |
| 9 | 막대 객체 구현하기 | |
| 10 | 벽돌 객체 구현하기 | |

스크래치 2.0이 속성과 기능, 상속과 같은 객체지향 개념을 정확하게 반영하고 있는 프로그래밍 언어는 아니지만, 비주얼 인터페이스 기반의 “추가 블록” 기능을 이용하면 객체를 정의할 수 있다. 즉, 활동지에서는 클래스로부터 상속받은 객체의 개념을 반영하고 있지만, 구현에서는 상속의 개념보다 복사의 개념을 반영하고 있다. 따라서, 수업을 진행하는 과정에서 상속과 복사의 개념의 차이를 설명하여 오개념의 가능성을 최소화하였다.

3.3 활동 내용

다음은 객체지향 개념 활동지를 사용하여 각 단계별로 진행된 수업 내용에 관한 설명이다.

3.3.1 1단계 - 객체 분석 활동

1단계 객체 분석 활동은 “벽돌깨기 게임”에 필

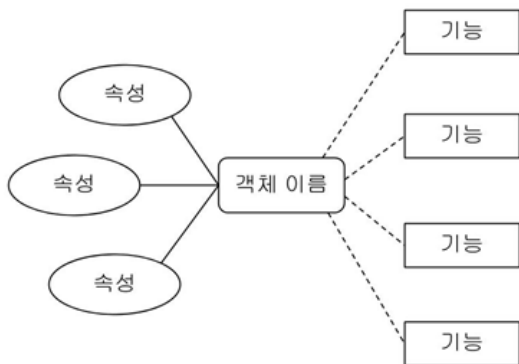
요한 객체를 정의하는 활동이다. 학습자는 객체 분석 활동지를 사용하여 게임을 만들 때 필요한 객체를 분석하고 정리한다. 객체 분석 활동지의 “객체” 부분에는 자신이 사용할 객체의 이름을 적고, “특징” 부분에는 객체가 가진 모든 특징을 적는다. <그림 4>는 객체 분석 활동지를 나타낸 것이다.

| 객체 | 특징 |
|-----|---------------------|
| 막대기 | (작성 예)막대기는 좌우로 움직인다 |
| ... | ... |
| | |

<그림 4> 객체 분석 활동지

3.3.2 2단계 - 객체의 속성과 기능 정의 활동

2단계 객체의 속성과 기능 정의 활동은 객체 분석 단계에서 정의한 객체의 세부 속성과 기능을 설계하는 단계이다. 객체 이름은 1단계에서 작성한 내용을 바탕으로 적을 수 있다. 속성과 기능은 시각적으로 명확히 구분하기 위해 속성은 동그라미와 실선, 기능은 네모와 점선을 이용해서 표현하도록 하였다. <그림 5>는 객체의 속성과 기능을 정의하는 활동지를 나타낸 것이다. 학습자는 자신이 정의할 속성과 기능에 맞게 활동지를 구성한다.



<그림 5> 객체의 속성과 기능 정의 활동지

3.3.3 3단계 - 공통되는 속성과 기능 분석 활동

3단계 공통되는 속성과 기능 분석 활동은 <그림 6>과 같이 2단계에서 작성한 객체의 속성과 기능을 나열하여 공통되는 부분을 분석하는 단계

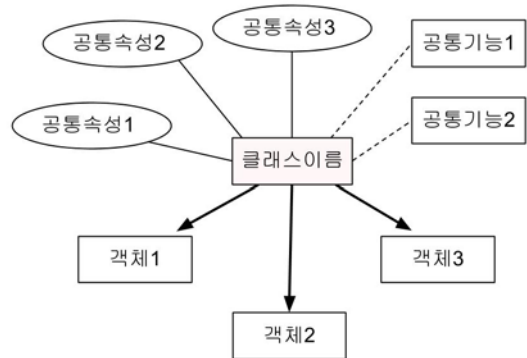
이다. 이 과정을 통해서 학습자는 모든 객체가 가지고 있는 공통 속성과 기능을 찾는다.

| | 속성1 | 속성2 | 속성3 | 속성4 | 기능1 | 기능2 |
|-----|-----|-----|-----|-----|-----|-----|
| 객체1 | ○ | ○ | ○ | | | |
| 객체2 | ○ | ○ | ○ | ○ | ○ | ○ |
| 객체3 | ○ | ○ | | | | ○ |

<그림 6> 공통되는 속성과 기능 분석 활동지

3.3.4 4단계 - 클래스 정의 활동 및 구현

4단계 클래스 정의 활동 및 구현은 3단계에서 분석한 객체의 공통 속성과 기능을 가진 클래스를 만들어내는 단계이다. 학습자는 <그림 7>과 같이 공통 속성과 공통 기능을 각각 원과 실선, 네모와 점선으로 표시하였다. 공통 속성과 기능을 가진 클래스로부터 상속받은 객체를 화살표로 표시한다.



<그림 7> 클래스 추상화 활동지

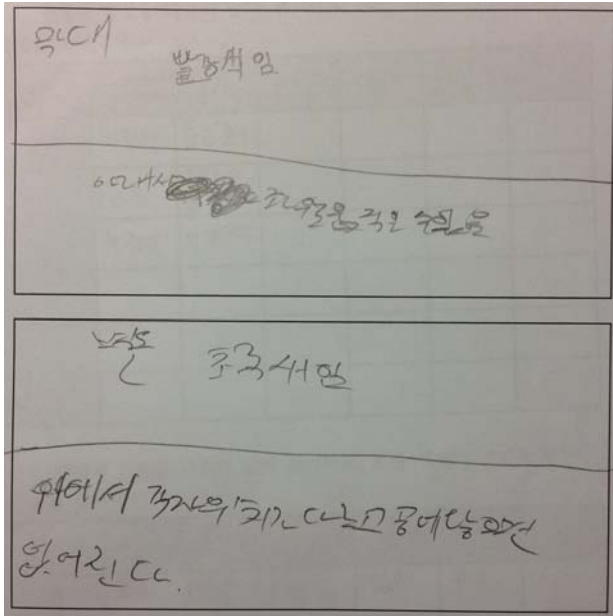
3.3.5 5단계 - 게임 기능 설계 활동 및 구현

게임 기능 설계 활동 및 구현은 자신이 만들고자 하는 “벽돌깨기 게임”을 스토리보드 형태로 작성하고 스크래치 2.0을 이용하여 구현하는 단계이다. 이를 위해 학습자는 스크래치의 기본 명령어와, 클래스 및 객체를 표현하기 위한 “추가 기능” 사용 방법을 익혔다.

4. 연구결과

본 연구는 학령이 낮은 학습자의 객체지향 개념 학습을 위한 목적으로 진행되었다. 즉, 활동

분석이 이루어지지 않은 것으로 해석할 수 있다. 셋째, 속성과 기능을 구분하지 못한 경우이다. <그림 11>과 같이 “움직이지 않는다”, “사라진다”와 같은 기능을 속성으로 표시한 경우이다.



<그림 10> 객체정의 활동의 부족한 작성 예

교사는 학습자가 객체의 속성과 기능을 구분할 수 있도록 속성과 기능에 대해 다음과 같이 설명하였다.

교사 : 속성은 다른 것과 비교되는 사물의 특징이나 고유한 성질로 명사 형태로 나타냅니다. 그렇다면 강아지의 속성은 어떤 것이 있을까요?

학습자 : 강아지의 색깔, 이름 등이 있어요.

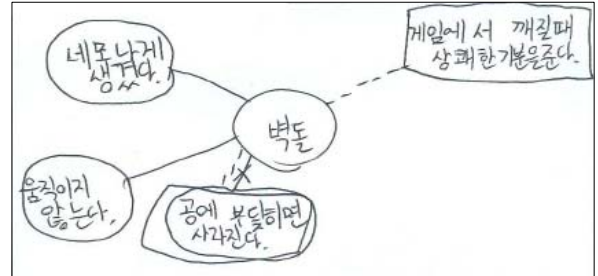
교사 : 맞아요. 기능은 사물의 구실이나 작용에 대한 것으로 동사 형태로 나타냅니다. 강아지의 기능에는 어떤 것이 있을까요?

학습자 : 달린다, 짖는다, 먹는다 등이 있어요.

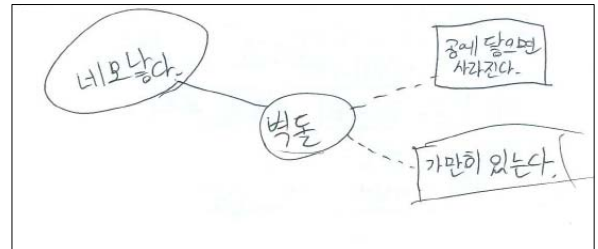
교사 : 잘했어요. 그렇다면 속성은 동그라미로, 기능은 사각형을 이용해서 표현해보세요.

교사는 학습자에게 속성과 기능의 정의에 대해 설명하고 학습자가 주변에서 속성과 기능을 찾을 수 있도록 도움을 주었다. 그 결과 학습자는 완벽하지는 않았지만, <그림 11>을 <그림 12>와 같이 객체의 속성과 기능을 구분하여 표현할 수 있게 되었다.

교사의 도움을 받기 전에 학습자는 객체의 속성과 기능에 대해 올바르게 이해하지 못했지만, 교사의 도움을 받은 후 객체를 올바르게 이해하고 구조적으로 표현한 것이다.



<그림 11> 속성과 기능을 구분하지 못한 예



<그림 12> 교사의 설명으로 수정한 예

학습자가 객체의 속성과 기능에 대해 명확하게 이해하지 못했다면 학습자가 이해하고 구조화 할 수 있도록 도움을 줄 필요가 있다. 이는 프로그래밍에서 복잡한 알고리즘을 작성하기 전에 간단한 의사코드(pseudo-code) 형태에서부터 점차 구체적으로 만들어야 한다는 연구에 기인한다[30].

4.3 3단계-공통되는 속성과 기능 분석 활동 결과

3단계는 2단계에서 정의된 객체의 공통 속성과 기능을 찾는 과정이다. 학습자는 게임에 필요한 객체를 작성하고 각 객체가 가진 속성과 기능을 나열한 후에 공통되는 속성을 찾는다.

3단계 활동 결과는 다음과 같다. 첫째, 학습자가 속성과 기능에 대해 올바르게 이해하고 분석하였다. <그림 13>과 <그림 14>는 학습자가 객체의 속성과 기능을 정확하게 구분하여 나열하고 객체가 가진 속성과 기능에 표시한 결과이다.

| 속성 | 색깔 | 위치 | 방향 | 속도 | 모양 | 크기 |
|----|----|----|----|----|----|----|
| 공 | ○ | ○ | | ○ | ○ | |
| 벽돌 | ○ | ○ | ○ | | ○ | ○ |
| 판 | ○ | ○ | | | ○ | ○ |

<그림 13> 공통되는 속성 분석 활동 결과

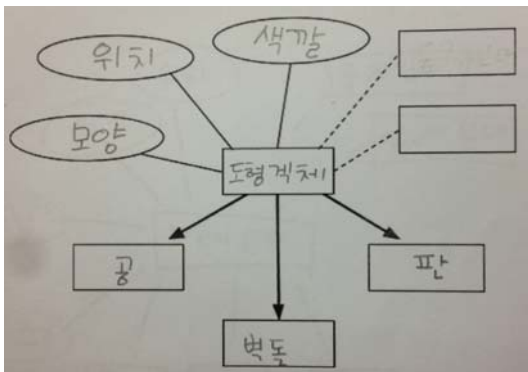
| 속성 | 입력값 | 어이값 | 뒤간다 | 움직인다 |
|----|-----|-----|-----|------|
| 공 | ○ | ○ | ○ | ○ |
| 벽돌 | | ○ | ○ | ○ |
| 판 | | | ○ | ○ |

<그림 14> 공통되는 기능 분석 활동 결과

둘째, 학습자가 공통되는 속성과 기능을 분석하는 활동의 목적에 대해 궁금해 하였다. 학습자는 2단계 “객체의 속성과 기능 정의 활동”과 3단계 “공통되는 속성과 기능 분석 활동”이 똑같은 활동을 하는 것은 아닌지 질문을 하였다. 교사는 3단계가 4단계 “클래스 정의 활동 및 구현”을 위해 객체의 속성과 기능을 정리하는 활동임을 설명하여 객체의 속성 정의와 클래스 정의 활동을 구분하도록 하였다.

4.4 4단계 - 클래스 정의 활동 및 구현 결과

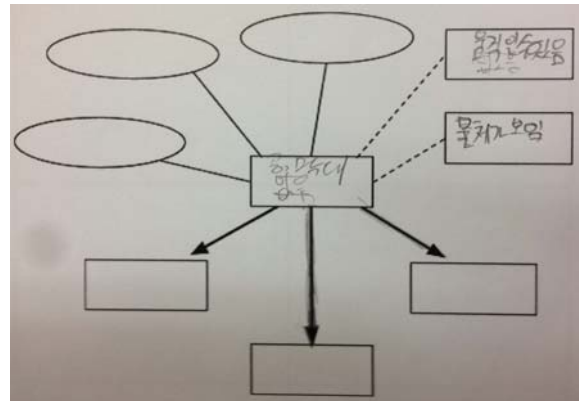
<그림 15>는 3단계에서 분석한 객체의 공통 속성과 기능을 가진 클래스를 만들고, 클래스로부터 상속받는 객체를 표시한 활동을 나타낸 것이다.



<그림 15> 클래스 정의의 올바른 예

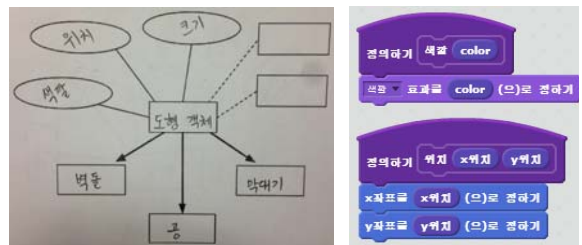
3단계 “공통되는 속성과 기능 분석 활동”에서 객체의 공통 속성과 기능에 대해 올바르게 작성하지 못한 학습자의 경우 <그림 16>과 같이 4단계 “클래스 정의 활동 및 구현”에서 클래스 정의 활동을 올바르게 수행하지 못하였다.

학습자가 작성한 클래스의 이름을 분석한 결과 클래스와 객체의 개념을 올바르게 이해한 학습자와 이해하지 못한 학습자 사이에 차이가 있음을 확인할 수 있었다. 예를 들어 클래스 개념을 올바르게 이해한 학습자는 클래스 이름을 ‘도형 객체’ 또는 ‘도형’ 과 같이 모든 객체를 포함할 수 있는 형태로 작성하였다. 반면, 이해가 부족한 경우 ‘공, 막대’, ‘도둑’, ‘가시’ 등으로 표시하였다. 초보 학습자는 전문가와 달리 클래스나 변수의 이름을 만들 때 기능을 명확하게 이해하고 있지 못하며 이름과 기능간의 구분이 명확하지 않은 것으로 해석할 수 있다.



<그림 16> 클래스 정의의 올바르지 못한 예

<그림 17>은 클래스 정의 활동을 바탕으로 “도형 객체”를 스크래치 블록으로 구현한 결과이다.



<그림 17> 클래스 정의 활동과 구현 결과

학습자는 “도형 객체”를 이용하여 “벽돌”과 “공”, “막대” 객체를 구현하였으며, 공통되는 속성인 “색깔”과 “위치”는 만들 필요가 없다는 것을

학습하였다.

다음 대화는 학습자가 “도형 객체”를 이용하여 또 다른 객체를 만드는 과정에서 공통되는 속성과 기능을 분석하여 클래스를 만드는 것이 얼마나 중요한지를 스스로 발견하는 과정을 나타낸 것이다.

교사 : 도형 객체를 이용해서 다른 객체를 만들어 보니 어떨까요?

학습자 : 모든 객체가 가진 속성이나 기능은 한번만 만들어도 되는 것 같아요

교사 : 공통되는 속성이나 기능을 가진 클래스를 사용하여 객체를 만들면 어떤 장점이 있나요?

학습자 : 똑같은 작업을 여러 번 하지 않아도 되고, 제가 만들고자 하는 것을 빠르게 만들 수 있는 것 같아요

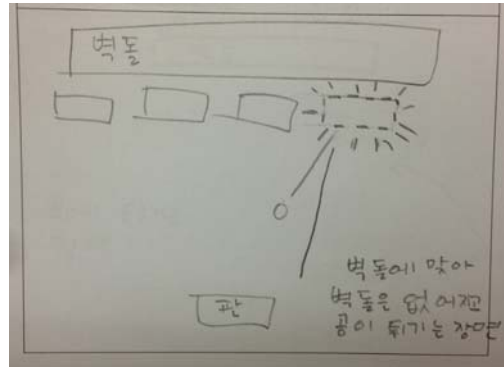
위의 대화에서 확인할 수 있듯이 학습자는 클래스를 이용하여 객체를 만들면 중복되는 기능을 여러 번 반복할 필요가 없어 빠르게 프로그래밍을 할 수 있다는 것을 알 수 있게 되었다. 객체를 분석하고 객체의 속성과 기능을 이용하여 클래스를 정의하는 활동이 학습자가 객체지향 개념 학습에 도움이 되었다는 것을 확인할 수 있다.

게임에 필요한 객체 분석, 객체의 속성과 기능 정의, 공통된 속성과 기능을 가진 클래스 정의 등의 순서를 통해 학습자는 구현 과정에서 활동지와 스크래치 블록의 일치성을 고려하였다. 따라서 활동지가 올바르게 작성되지 못할 경우 스크래치로 구현하는 과정에서 어려움이 있음을 확인하였다. 즉, 초보 학습자가 객체지향 프로그래밍을 하기 위해서는 문제 해결에 필요한 객체를 분석하고 속성과 기능을 올바르게 정의하는 활동이 우선임을 보여준다. 학습자 스스로 무엇을 만들어야 하는지 모르는 경우 알고리즘 작성에 혼란을 겪게 된다는 연구와 일치하는 결과이다[7].

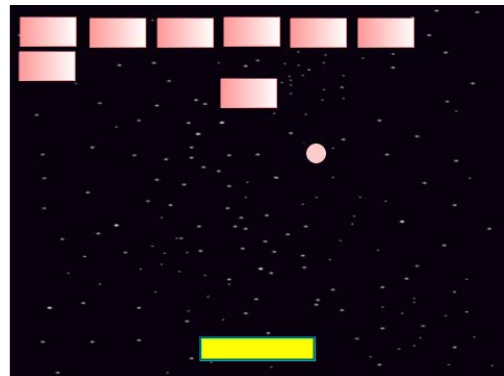
4.5 5단계 - 게임 기능 설계 활동 및 구현

<그림 18>은 자신이 만들고자 하는 게임의 화면을 설계한 활동의 결과이다. 학습자는 자신이 설계한 내용을 바탕으로 <그림 19>와 같이 게임

을 구현하였다.



<그림 18> 게임 기능 설계 활동 결과



<그림 19> 게임 기능 구현 결과

4.6 설문 결과

다음 <표 3>은 게임 기반의 객체지향 개념 학습을 하기 전후 학습자의 프로그래밍에 대한 인식 조사 결과이다.

<표 3> 설문 결과

| 설문 문항 | 구분 | M | SD |
|--------------------------------------|----|------|---------|
| 나는 물건을 볼 때 물건이 가진 특성이 무엇인지 알 수 있다. | 사전 | 4.00 | .81650 |
| | 사후 | 4.50 | .57735 |
| 나는 물건을 볼 때 물건이 가진 기능이 무엇인지 알 수 있다. | 사전 | 4.00 | .00000 |
| | 사후 | 4.50 | .57735 |
| 비슷한 종류의 물건이 주어졌을 때 공통된 특징을 찾아낼 수 있다. | 사전 | 4.00 | .81650 |
| | 사후 | 4.50 | .57735 |
| 무언가를 만들 때 계획을 하는 것은 중요하다고 생각한다. | 사전 | 3.25 | .95743 |
| | 사후 | 4.25 | .95743 |
| 나는 프로그래밍이 재미있다. | 사전 | 2.50 | 1.29099 |
| | 사후 | 4.75 | .50000 |
| 내가 계획한 것을 컴퓨터로 그대로 만드는데 자신이 있다. | 사전 | 3.00 | .81650 |
| | 사후 | 3.75 | .57735 |

설문 결과 학습자는 객체의 속성과 기능에 대한 이해도가 증가되었고 공통된 특징을 찾아내는데 자신감도 향상되었다. 프로그래밍 활동에서 계획의 중요성을 고려하는 것에도 본 활동지를 활용한 수업은 도움을 준 것으로 나타났다. 활동지를 이용하여 객체를 정의하는 활동을 한 후 컴퓨터로 구현하는 것에 대한 자신감도 향상되었다. 특히, “프로그래밍이 재미있다”라는 의견에 주목할 필요가 있다. 수업 전에는 평균 2.50, 표준편차 1.29로 낮은 평균일 뿐 아니라 다른 의견에 비해 분산도 크게 났다. 수업 후에는 1.29의 편차가 0.5로 줄어들었고 프로그램이 재미있다는 의견도 4.75이다. 게임을 소재로 했기 때문에 학습자들의 흥미를 증가시키기도 했지만, 분산이 줄어들었다는 것은 고른 의견을 보여주는 결과라 할 수 있다.

5. 결론 및 논의

본 연구는 학령이 낮은 학습자가 객체지향의 필수 요소를 학습하는데 효과적인 방법을 제안하기 위한 목적으로 진행하였다. 연구의 목적을 달성하기 위해 게임을 요소로 한 프로그래밍 수업을 진행하였다. 수업 사례는 학습자의 활동지 작성내용과 구현 과정을 분석하였다. 분석을 위해 본 연구는 초등학교 5-6학년 학습자 6명을 대상으로 활동지를 사용한 객체지향 프로그래밍 수업을 10차시에 걸쳐 진행하였다.

연구 결과, 게임에 필요한 객체를 분석하고 객체의 속성과 기능을 이용하여 클래스를 올바르게 정의한 학습자는 논리적이고 자신감 있게 프로그래밍을 수행하였다. 객체와 클래스에 대해 올바르게 이해하지 못한 학습자는 구현을 하는 과정에서 무엇을 만들어야 하는지 구체화하지 못하였다. 이는 프로그래밍 과정에서 구현하고자 하는 내용에 대해 생각하고 활동지를 통해 자신의 생각을 정리하는 성찰의 시간이 프로젝트 수행에 도움을 주었다는 연구와 일치하는 결과이다[27].

연구 결과를 토대로 할 때, 학령이 낮은 학습자를 위한 객체지향 프로그래밍 수업에서는 객체의 개념에 대해 올바르게 이해하고 객체의 속성과 기능을 명확하게 정의할 수 있는 활동이 우선되

어야 한다. 그리고 교사는 학습자가 성찰의 과정을 통해 자신이 구현하고자 하는 내용을 명확하게 이해하고 설계할 수 있도록 도움을 줄 필요가 있다.

본 연구를 통해 학령이 낮은 학습자 대상의 객체지향 프로그래밍 수업에서는 학습자가 객체지향 개념을 친숙한 형태로 표현하고 학습할 수 있는 학습 도구가 필요하다는 것을 확인할 수 있었다. 또한, 학생들이 프로그래밍 구현 전에, 충분히 사고하고 개념을 습득할 필요가 있음을 발견하였다.

본 연구는 초보학습자가 이해하기 어려워하는 객체지향 프로그래밍 학습에 도움이 되는 학습 방법을 제시했다는 데 의의가 있다.

참고 문헌

- [1] Denning, P. J. (2007). Computing is a natural science. *Communications of the ACM*, 50(7), 13-18.
- [2] 최해심 · 권대용 · 김형신 · 염용철 · 유승욱 · 이원규 (2005). 객체지향형 교육용프로그래밍 언어 두리틀의 다중 예약어 지원 체계. *컴퓨터교육학회논문지*, 8(2), 23-32.
- [3] 권대용 · 허경 · 이원규 (2010). 초등 교육에서의 PBL기반 라인트레이서 로봇프로그래밍 교육방법 개발. *컴퓨터교육학회논문지*, 13(3), 13-23.
- [4] 김대수(2010). *컴퓨터개론*. 경기 : 생능출판사.
- [5] 한국교육과정평가원 (2011), 2011년 개정교육과정 고등학교 교육과정, VI-7, 정보.
- [6] Wei, F., Moritz, S. H., Parvez, S. M., & Blank, G. D. (2005). A student model for object-oriented design and programming. *Journal of Computing Sciences in Colleges*, 20(5), 260-273.
- [7] Gomes, A., & Mendes, A. J. (2007). Learning to program-difficulties and solutions. *International Conference on Engineering Education - ICEE*.
- [8] Nelson, H. J., Armstrong, D. J., & Nelson, K. M. (2009). Patterns of transition: The

- shift from traditional to object-oriented development. *Applied Journal of Management Information Systems*, 25(4), 271-298.
- [9] Thota, N., & Whitfield, R. (2010). Holistic approach to learning and teaching introductory object-oriented programming. *Computer Science Education*, 20(2), 103-127.
- [10] Kosmadoudi, Z., Lim, T., Ritchie, J., Louchart, S., Liu, Y., & Sung, R. (2013). Engineering design using game-enhanced CAD: The potential to augment the user experience with game elements. *Computer-Aided Design*, 45(3), 777-795.
- [11] Rentsch, T. (1982). Object oriented programming. *ACM Sigplan Notices*, 17(9), 51-57.
- [12] Hadar, Irit. (2012). When intuition and logic clash: The case of the object-oriented paradigm. *Applied Science of Computer Programming*, 78, 1407-1426.
- [13] Wegner, P. (1990). Concepts and paradigms of object-oriented programming. *ACM SIGPLAN OOPS Messenger*, 1(1), 7-87.
- [14] 송호창 · 이영준 (2011). 로봇을 활용한 객체지향프로그램 학습 방안. **컴퓨터교육학회 동계학술발표대회논문집**, 17(1), 111-115.
- [15] 양근우 · 조혁수 (2011). 비즈니스 프로세스의 표현과 수행을 위한 객체지향적 접근법. **정보기술아키텍처연구**, 8(3), 215-228.
- [16] Armstrong, D. J. (2006). The quarks of object-oriented development. *Communications of the ACM*, 49(2), 123-128.
- [17] 이원규(1991). **컴퓨터교육론**. 서울 : 교학사
- [18] 권대용 (2013). 텐지블 프로그래밍 도구를 활용한 논리적 사고력 기반의 초등 로봇 과제 개발 및 적용. **컴퓨터교육학회논문지**, 16(4), 13-21.
- [19] 권대용 · 길혜민 · 염용철 · 유승욱 (2004). 중등 컴퓨터과학교육을 위한 객체지향형 EPL ‘두리틀’의 적용 및 평가. **컴퓨터교육학회 논문지**, 7(6), 1-12.
- [20] Kanemune, S., & Kuno, Y. (2005). Dolittle: an object-oriented language for K12 education. *EuroLogo*.
- [21] 김지수 (2010). EPL ‘Alice’를 활용한 프로그래밍 원리 학습. **정보창의교육논문지**, 4(1), 9-15.
- [22] 권대용 · 염용철 · 유승욱 · 이원규 (2005). 두리틀 로봇 프로그래밍 일원화를 위한 로봇 객체 설계. **컴퓨터교육학회 논문지**, 8(6), 1-10.
- [23] AMORIM, C. A. (2005). Beyond Algorithmic Thinking: An Old New Challenge for Science. *Philosophy & Science Teaching Conference*.
- [24] Jenkins, T. (2002). On the Difficulty of Learning to Program. *Proceedings of 3rd LTSN-ICS Conference*, 53-58.
- [25] Keppens, J., & Hay, D. (2008). Concept map assessment for teaching computer programming. *Computer Science Education*, 18(1), 31-42.
- [26] 김용천 · 최지영 · 권대용 · 이원규 (2013). 초 등학생의 프로그래밍 학습을 위한 알고리즘적 사고 문제 모델 기반의 활동지 개발 및 적용. **정보교육학회논문지**, 17(3), 233-242.
- [27] 김용천 · 김자미 · 이원규 (2012). 초 등학생의 프로그래밍 학습에서 활동지를 사용한 성찰에 대한 사례 연구. **정보교육학회논문지**, 16(1), 21-31.
- [28] 최주희 · 김현철 (2011). 객체지향 개념학습에서의 스토리텔링 활용 방법 연구. **컴퓨터교육학회 학술발표대회논문집**, 15(1), 117-122.
- [29] 이길섭, 신동완 (역) (2008). **AKIRA HIRASAWA의 성공과 실패를 결정하는 1%의 객체지향원리**. 성안당.
- [30] Karatrantou, A., Panagiotakopoulos, C., & Patras, A. (2008). Algorithm, pseudo-code and Lego Mindstorms programming. *Proceedings of International Conference on Simulation and Programming for Autonomous Robots/Teaching with Robotics: Didactic Approaches and Experiences*.



김 용 천

2010 고려대학교
컴퓨터교육학과(이학사)
2012 고려대학교
컴퓨터교육학과(이학석사)

2014~현재 고려대학교 컴퓨터교육학과
박사수료

관심분야: 컴퓨터교육, 프로그래밍 교육, 알고리즘

E-Mail: yongcheon.kim@inc.korea.ac.kr



이 원 규

1995 고려대학교
영어영문학과(문학사)
1989 筑波大學
理工學研究科(공학석사)

1993 筑波大學 工學研究科(공학박사)

1996~현재 고려대학교 컴퓨터학과 교수

관심분야: 컴퓨터교육, 정보검색, 데이터베이스

E-Mail: lee@inc.korea.ac.kr



장 윤 재

2008 고려대학교
컴퓨터교육과(이학사)
2012 고려대학교
컴퓨터교육학과(이학석사)

2014~현재 고려대학교 컴퓨터교육학과
박사수료

관심분야: 프로그래밍교육, 교육용로봇, 정보윤리

E-Mail: yunjae.jane@inc.korea.ac.kr



윤 일 규

2008 공주대학교
컴퓨터교육과(이학사)
2010 고려대학교
컴퓨터교육학과(이학석사)

2012~현재 고려대학교 컴퓨터교육학과
박사수료

관심분야: 정보교육과정, 정보교육평가

E-Mail: ilkyu.yoon@knue.ac.kr



김 자 미

1992 이화여자대학교
교육학과(문학사)
1995 이화여자대학교
교육학과(문학석사)

2011 고려대학교 컴퓨터교육학과(이학박사)

2011~현재 고려대학교 컴퓨터학과 연구교수

관심분야: 컴퓨터교육, 교육정보화평가

E-Mail: jamee.kim@inc.korea.ac.kr