

# 프로그래밍 교육에서 동료 에이전트가 학업성취도와 자기효능감에 미치는 영향

한건우<sup>†</sup> · 이은경<sup>†</sup> · 이영준<sup>†\*</sup>

## 요 약

본 연구의 목적은 프로그래밍 교육을 지원하기 위한 동료 에이전트를 개발하고, 실제 프로그래밍 교육에 적용하여 동료 에이전트의 교육적 효과를 검증하는 것이다. 학습자와 에이전트 사이의 역할을 동료 프로그래밍 관점에서 Tutor와 Tutee로 정의하고, 에이전트가 학습자의 수준을 진단하기 위한 베이저안 네트워크 학습자 모델을 개발하였다. 이를 기반으로 동료 에이전트는 학습자의 수준에 따라 적절한 피드백이나 학습내용을 제공한다. 개발된 동료 에이전트 시스템을 실제 프로그래밍 교육에 적용한 결과, 학습자의 학업성취도와 자기효능감 증진에 효과가 있는 것으로 나타났다. 이는 동료 에이전트 학습 시스템이 학습자의 인지적 영역과 더불어 정의적 영역에 있어서도 긍정적인 영향을 주는 것을 의미한다.

키워드 : 프로그래밍 교육, 동료 프로그래밍, 동료 에이전트

## The Effects of a Peer Agent on Achievement and Self-Efficacy in Programming Education

Keun-Woo Han<sup>†</sup> · Eun-Kyoung Lee<sup>†</sup> · Young-Jun Lee<sup>†\*</sup>

### ABSTRACT

We have developed a peer agent to support programming learning and analyzed its educational effects in a programming course. The agent acts as a tutor or a tutee. The role of a tutor/tutee is like the role of a navigator/driver in pair programming. While students are learning with the peer agent, the students' programming abilities are modeled. Based on the student's model, the peer agent provides appropriate feedbacks and contents to the learner. The peer agent gives positive effects on learners' achievement and self-efficacy in a programming course. It means that the peer agent system helps the learner in an affective domain as well as a cognitive domain.

Keywords : Programming Education, Pair Programming, Peer Agent

## 1. 서 론

프로그래밍 교육은 컴퓨터 과학 분야에 있어서 매우 중요한 영역이며, 대학 교육뿐만 아니라 IT 산업 발전을 위한 인력 양성 측면에서도 중요한 가치를 지닌다. 이러한 국가적, 시대적 중요성을 반영하여 실업계 고등학교에서는 프로그래밍 관

<sup>†</sup> 종신회원: 한국교원대학교 컴퓨터교육과 박사과정  
<sup>†\*</sup> 종신회원: 한국교원대학교 컴퓨터교육과 교수(교신저자)  
 논문접수: 2007년 4월 22일, 심사완료: 2007년 6월 27일  
 이 논문은 2006년 정부(교육인적자원부)의 재원으로  
 한국학술진흥재단의 지원을 받아 수행된 연구임  
 (KRF-2006-332-B00454)

런 교육과정이 운영되고 있으며, 2007년 2월 28일 새롭게 개정된 교육과정을 통해 컴퓨터 과학의 원리를 기반으로 한 프로그래밍 교육이 크게 부각되고 있다[1].

그러나 프로그래밍 과목은 학생들에게 배우기 어려운 과목이며, 교사들에게는 가르치기 어려운 과목으로 인식되고 있다. 그 이유는 학생들에게 복잡한 인지적 능력을 요구하기 때문이다[12]. 이러한 문제점을 해결하기 위해 프로그래밍 교육 연구 분야에서는 교수적인 방법과 기술적인 방법으로 접근하고 있다. 교수 방법적인 측면의 연구들은 프로젝트 기반 학습[2]이나 동료 교수법[3] 등의 교수·학습 전략을 통해 학업성취도를 향상시키고자 하였으며, 정의적 측면에서 자기효능감(self-efficacy)이나 학습 동기화 같은 요인들이 프로그래밍 능력 향상에 긍정적인 영향을 끼치는 것으로 나타났다[17]. 반면, 기술적인 측면의 연구들은 학습자의 정보를 통해 추론하여 학습자의 프로그래밍 수준을 결정하고 학습자에게 적합한 학습내용을 제시해주는 지능형 교수 시스템을 중심으로 연구되고 있다[9].

본 연구에서는 프로그래밍 교육의 효과성을 높이기 위해 동료 프로그래밍(Pair Programming)의 관점에서 동료의 역할을 하는 동료 에이전트를 개발하였다. 동료 프로그래밍은 프로그래밍 학습에 있어서 긍정적인 효과가 있는 것으로 나타났다[6].

그러나 동료 프로그래밍은 동료가 있어야만 작업을 수행할 수 있으며, 동료에 따라 결과가 영향을 받는 것으로 나타났다. 동료의 수준이 서로 낮으면 복잡한 프로그램을 작성하기 어려우며, 수준차가 많이 나면 한 사람에 의해 작업이 이끌려 갈 수 있다[11]. 즉, 동료의 인지적 특성 및 정의적 특성에 따라 학습 결과에 영향을 끼칠 수 있으며, 동료가 자신의 역할에 따른 책임을 다하지 못하게 되면 프로그래밍 효과가 떨어지므로 역할 및 전략 연습에 시간이 많이 소요될 수 있다. 따라서 동료 프로그래밍은 올바른 동료의 선정을 전제로 해야 하며, 동료의 프로그래밍 능력, 자세에 따라 동료 프로그래밍의 효과는 달라진다. 무엇보다도 현실 세계에 동료가 있어야 한다는 큰 제약점을 가지고 있다.

본 연구에서는 이러한 문제점을 동료 에이전트를 통해 해결하고자 한다. e-러닝 환경에서 효율적이고 원활한 협력학습을 기대하기 위해서는 에이전트 기반 접근 방법이 적합하다[5]. 본 연구에서는 동료 프로그래밍의 문제점을 해결하기 위해 효과적인 프로그래밍 학습을 위한 동료 에이전트 기반 학습시스템을 개발하고 동료 에이전트의 학습 효과를 분석하였다.

## 2. 관련 연구

### 2.1 동료 프로그래밍(Pair Programming)

동료 프로그래밍은 두 명의 프로그래머가 하나의 컴퓨터를 사용하여 알고리즘 설계, 코딩, 디버깅 등을 협업하는 것을 말한다. 동료 중 한 사람(Driver)은 컴퓨터의 키보드와 마우스를 사용하여 코딩 업무를 담당하고, 다른 동료 한 사람(Navigator)은 알고리즘을 검색하거나 문법적 오류나 오타 등을 찾아주어 작업의 효율화를 가져온다. 이러한 과정을 통해 동료 간에 상호작용이 촉진되며, 다양한 의견을 제시함으로써 서로의 문제점을 보완해주거나 새로운 아이디어를 얻을 수 있다. 동료 프로그래밍의 역할에 따른 작업을 간략하게 정리하면 <표 1>과 같다[6].

<표 1> 동료 프로그래밍에서의 역할

역할	작업	비고
Driver	키보드와 마우스로 작업하기 프로그래밍 코딩 작업하기 코딩 설계하기	상호간의 의사소통하기
Navigator	필요한 알고리즘 구상하기 코딩 상의 오타 찾아내기 문법적인 오류 찾아내기	

이러한 동료 프로그래밍에 대해 제시된 교육적 효과들은 다음과 같다[16].

첫째, 프로그래밍에 대한 기초적인 개념을 이해하는데 도움이 되었으며 상호작용을 통해 서로의 지식을 공유할 수 있어 학습에 도움이 되었다.

둘째, 동료 프로그래밍의 역할을 통해 상호간의 대화와 질문에 대한 설명이 진행된다. 이러한 과정 속에 학습자는 스스로의 지식을 구조화하고 지식이 전이되는 과정을 통해 학습에 큰 영향을 주는 것으로 나타났다.

그러나 동료 프로그래밍은 프로그래머의 수준에 따라 효과가 다르게 나타날 수 있으며, 그에 따라 동료의 선정 및 역할이 중요한 문제로 대두되었다[11]. 또한 동료와 함께 짝을 지어야 하므로 오프라인상에서의 만남이 요구된다.

## 2.2 에이전트

최근 학습자를 조력하기 위한 에이전트를 학습 시스템과 연계한 교수형 에이전트에 관한 연구가 대두되고 있다. e-러닝의 발달로 인해 원격 교육이 크게 활성화 되었으며, 특히 웹 기반 교육은 전통적인 교육에 비해 보다 융통성 있고, 개별화된 환경을 제공해 줄 수 있다. 그러나 이러한 환경은 학습자에게 혼란을 가져올 수 있으며, 비인격적이고, 동기 유발이 어려운 문제점을 가지고 있다[15]. 이러한 문제점을 해결하기 위해 에이전트를 활용한 접근 방법이 연구되고 있다.

최근 에이전트 관련 연구들은 주로 교수적 에이전트(Pedagogical Agent)나 지능형 에이전트(Intelligent Agent)의 관점에서 진행되고 있으며, 학습자의 동기를 유발하고 적응적인 힌트를 제공하여 웹 기반 학습에 쉽게 적용할 수 있도록 하고 있다[10]. 또한, 웹 기반 학습은 서로 떨어진 학습자 간의 협력학습을 지원하기에 적합하나 기존의 게시판이나 화이트보드와 같은 상호작용 도구들은 협력학습을 촉진하기에 부족하므로 이를 보완하기 위한 협력학습 지원 에이전트에 관한 연구도 진행되고 있다[4].

이러한 에이전트는 학습자에게 도움이 될 수 있는 협력적 역할을 수행하므로, 먼저 에이전트의 역할과 기능을 정의하는 것이 중요하다. 학습자와의 상호작용을 통해 수집된 정보는 학습자 모델링으로 구현하기 위한 주요한 정보로 활용될 수 있으므로 에이전트의 활용이 학습자 모델링을 위한 전략적 요소로 실현 가능함을 보여주고 있다.

## 2.3 학습자 모델링

지능형 교수 시스템(ITS: Intelligent Tutoring System)은 학습자의 문제 해결과정을 모니터링하여 유용한 정보를 수집하고 이를 근거로 학습자의 지식수준을 추론하여 수준별 학습 내용과 교수 전략을 제공하는 시스템이다. 따라서 지능형 교수 시스템은 학습자의 능력에 따라 개별화 수업이 가능하며, 학습자의 오개념을 교정해 줄 수 있어 교육적 효과가 기대되고 있다.

학습자 모델링은 학습자의 지식에 대한 현재의 상태를 표현하는 것으로 이를 위해 학습자의 정보를 수집하고 추론하는 과정을 말한다. 학습자 모델링에는 지식 추적형(Knowledge Tracing)과 모델 추적형(Model Tracing)으로 구분할 수 있다[7]. 지식 추적형 모델링은 해당 주제 영역에 대한 올바른 지식과 명확한 오개념을 근거로 학습자가 알고 있는지를 결정하는 문제를 주로 다룬다. 이 모델은 학습자의 지식을 평가한 후 다음의 학습 내용이나 문제를 결정하므로 장기간의 시간을 요구한다. 모델 추적형은 학습자가 문제를 해결해 나가는 과정을 추적하여 모델링한다. 이 모델은 문제 해결 과정 중에 학습자가 요구하는 힌트나 피드백을 통해 학습자의 지식을 추론한다.

학습자 모델링은 지식에 대한 도메인과 학습자의 수준이나 오개념을 통해 구조화할 수 있는데, 크게 오버레이 모델(Overlay Model), 차분 모델(Differential Model), 버그 모델(Bug Model), 버그 목록 모델(Bug Library Model)로 구분할 수 있다[7].

지능형 교수 시스템이 전통적인 학습 시스템과 구분되는 주요 요소 중에 하나는 학습자들이 학습하는 과정을 모델링하기 위해 학습자의 행동을 이해하려는 노력이 요구된다는 점이다. 프로그래밍 학습에서 학습자의 행동을 이해하기 위해서는 다양한 상호작용과 정보 수집이 필수적이며, 이를 기반으로 한 추론의 과정이 요구된다.

이러한 관점에서 볼 때 동료 프로그래밍 과정에서 학습자와 상호작용하는 동료 에이전트는 학습자 모델링을 위한 학습자 정보를 수집하는데

유리하다고 볼 수 있다. 동료 에이전트의 역할을 통해 각기 다른 차원에서의 상호작용이 가능하므로 보다 정확한 정보를 수집할 수 있기 때문이다. 또한 문제 해결 과정에서 발생될 수 있는 학습자의 문법적 오류를 버그 목록 라이브러리로 구축할 수 있으며, 알고리즘의 의미를 순서도 등을 통해 인식할 수 있게 함으로써 학습자의 행동에 대한 이해가 가능하다.

### 3. 프로그래밍 교육을 위한 동료 에이전트 학습 시스템 개발

#### 3.1 동료 에이전트 역할 설계

동료 프로그래밍의 역할 분배를 통해 실습 활동을 통한 학습(Learning by Doing)과 가르침을 통한 학습(Learning by Teaching)을 수행하도록 개발하였다. 이는 프로그래밍 교육을 위한 동료 에이전트의 역할을 강의자(Tutor)와 학생(Tutee)의 개념에서 접근한 것이며, 동료 에이전트와의 상호작용을 통해 프로그래밍 학습을 수행하게 된다. 본 연구에서는 동료 프로그래밍을 지원하는 에이전트를 구현하기 위해 에이전트의 역할을 Driver와 Navigator로 구분하였으며, 이는 <표 2>와 같다.

<표 2> 동료 에이전트의 역할

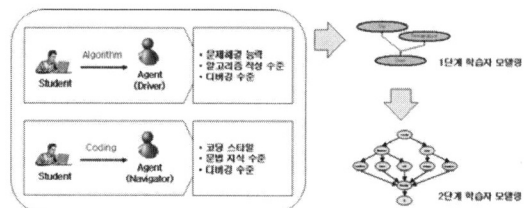
역할	내용
Driver	<ul style="list-style-type: none"> <li>- 학습자에게 코딩을 제시함</li> <li>- 학습자는 에이전트에게 문제해결을 위한 과정을 설명함</li> <li>- 에이전트는 학습자에게 학습을 받는 학생(Tutee)의 성격을 가짐</li> <li>- 가르침에 의한 학습(Learning by Teaching)을 실현함</li> </ul>
Navigator	<ul style="list-style-type: none"> <li>- 학습자에게 알고리즘을 제공함</li> <li>- 에이전트는 학습자에게 문제해결의 힌트나 학습내용을 제공함</li> <li>- 에이전트는 학습자에게 학습을 하는 강의자(Tutor)의 성격을 가짐</li> <li>- 활동(실습)에 의한 학습(Learning by Doing)을 실현함</li> </ul>

학습자는 동료 에이전트와의 역할을 번갈아가며 학습한다. 프로그래밍 언어에 대한 학습 내용을 실제 코딩하여 실습하거나 동료 에이전트에게 문제해결의 과정을 설명하고 동료 에이전트의 코딩 과정 중 발생하는 문제를 수정한다. 프로그래밍 학습을 위한 동료 에이전트의 교수 전략으로 학습자에게 제공되는 역할 모형을 제공한다.

#### 3.2 지능형 에이전트를 위한 베이지안 네트워크 학습자 모델

프로그래밍 교육을 위한 학습자 진단 모델로서 베이지안 네트워크 학습자 모델을 선정하였다. 프로그래밍 교육은 학습의 과정이 순차적이며, 학습 종속적인 성격을 지니고 있으므로 베이지안 네트워크를 통해 학습자의 수준을 진단할 수 있다. 본 연구에서는 한건우, 이은경, 이영준[13]이 개발한 CTi 모델의 학습자 진단 모듈을 사용하여 학습자를 평가하고 진단하도록 구현하였다.

학습자 정보는 동료 에이전트의 역할(Driver/Navigator)에 따라 수집 가능한 학습자의 정보를 분류하고 추론 결과를 바탕으로 학습 전략을 수립하였다. 에이전트가 Driver일 경우 학습자는 문제해결을 위한 표현과정을 거치며, 학습자의 알고리즘 구현 능력에 대한 정보를 수집할 수 있다. 에이전트가 Navigator일 경우 학습자는 문제해결 과정에 따른 코딩 스타일과 해당 언어에 대한 문법적 지식 정보를 수집할 수 있다. 특히 학습자의 디버깅 정보를 수집하여 학습자의 버그 목록을 라이브러리화 할 수 있다. 구현된 학습자 모델링은 프로그래밍 학습 시스템에 통합할 수 있도록 모듈화하였으며, 프로그래밍 교육을 위한 학습자 모델링 추론 과정은 <그림 1>과 같다.

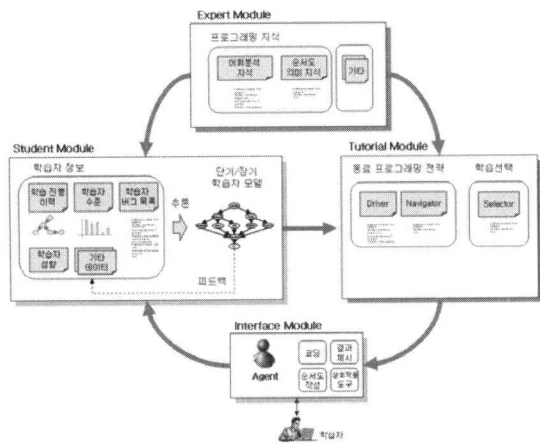


<그림 1> 학습자 추론 과정

학습자 모델링을 위한 추론은 Dynamic Bayesian Network를 활용하였으며, 학습 상태와 시간의 흐름에 따라 1단계 학습자 모델과 2단계 학습자 모델을 구현하였다. 1단계 학습자 모델링은 학습자가 문제 해결 과정이나 코딩의 과정에서 얻을 수 있는 정보를 근거로 차시별로 학습자의 수준을 진단하며, 이 결과를 차후에 2단계 학습자 모델에 적용하여 프로그래밍 전체에 대한 학습자의 수준을 진단하도록 하였다.

### 3.3 동료 에이전트 학습 시스템

지능형 교수 시스템은 일반적으로 전문가 모듈(Expert Module), 교수 모듈(Tutorial Module), 학습자 모듈(Student Module), 인터페이스 모듈(Interface Module)로 크게 네 가지로 구성되어 있다. 동료 에이전트를 통한 프로그래밍 학습 시스템은 지능형 교수 시스템의 네 가지 구성요소에 따라 해당 모듈별로 개발하였다. 구체적인 시스템 구조도는 <그림 2>와 같다.

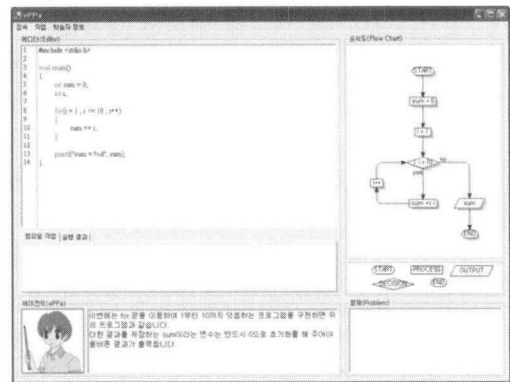


<그림 2> 학습 시스템 구조도

인터페이스 모듈은 학습자와의 상호작용을 위한 동료 에이전트 중심의 인터페이스로 개발하였으며, 학습자의 프로그래밍 지식과 코딩 등을 인식할 수 있는 전략적 도구를 제시하였다. 학습자 모듈은 학습자와 에이전트 간에 발생하는 다양한 활동 정보를 모니터링하고, 학습 이력 등을 기반

으로 학습자의 수준을 추론하도록 개발하였다. 전문가 모듈은 프로그래밍 언어에 대한 지식을 제공하며, 순서도와 코드의 오류를 진단하는 지식을 제공하도록 설계하였다. 교수 모듈은 학습자의 수준 및 상황에 따라 에이전트의 역할을 결정하고, 맞춤형 학습 내용 선택 및 피드백을 제공한다.

학습자는 로그인 과정을 통해 학습 이력에 맞는 학습 내용을 동료 프로그래밍 전략에 따라 동료 에이전트와 함께 학습하게 되며, 학습 후에 평가를 통해 학습자의 수준을 진단하게 된다. 학습 시스템 구조도에 따라 구현된 동료 에이전트 화면은 <그림 3>과 같다.



<그림 3> 동료 에이전트 구현 화면

동료 에이전트는 프로그래밍 언어 중 C 언어의 학습 내용을 학습자에게 제공한다. 동료 에이전트는 학습자가 컴파일 하는 과정을 모니터링하여 학습자가 가지고 있는 에러의 정보를 수집할 수 있다. 이러한 정보를 피드백으로 제공하여 학습자가 자신의 오류 내용을 인지할 수 있도록 구현하였다.

동료 에이전트는 학습자에게 Driver와 Navigator의 역할을 교수·학습 전략으로 제공한다. 학습자는 역할 모형을 수행하며 프로그래밍 언어의 문법과 문제해결 전략의 과정을 학습한다. 역할별로 학습내용을 진행한 후에는 관련 내용에 대해 평가를 진행하며, 평가 결과를 중심으로 페이지안 네트워크 학습자 모델링을 통해 학습자의 수준을 진단한다. 학습 주제에 대해 학습

을 마친 후에는 1단계 학습자 모델에 기반을 둔 학습자의 수준에 따라 학습자가 부족한 학습 소주제에 대해 보충학습을 제공한다.

## 4. 연구 방법

### 4.1 연구대상 및 검사도구

본 연구는 C 프로그래밍 언어를 배우는 실업계 고등학교 2개 반 68명의 학생을 대상으로 진행하였다. 이 중 임의의 1개 반 34명을 실험집단으로 선정하여 동료 에이전트 학습 시스템으로 학습하게 하고, 다른 1개 반 34명을 통제집단으로 선정하여 전통적인 강의법 및 실습·시연 수업을 실시하였다. 본 연구에 참여한 학생들은 프로그래밍 작성과 순서도에 대한 사전 지식이 있는 집단이며, 두 집단 모두 동일한 교사가 수업을 진행하였다. 수업 처치 후 실험에 참여하고 설문에 응한 학생 56명(실험집단 27명, 통제집단 29명)이 최종 연구 분석 대상으로 선정되었다.

학생들의 프로그래밍 능력을 평가하기 위해 프로그래밍 평가 항목[8]을 중심으로 8개의 학업성취도 문항을 개발하였다. 본 연구에서 개발한 문항은 실제 수업 담당 교사에게 문항의 난이도를 검토 받았으며, 프로그래밍 교육 전문가 2인에게 안면 타당도를 검증 받았다. 또한 프로그래밍 교육에 있어 주요 요인 중의 하나로 제시되고 있는 자기효능감을 집단별로 측정하고 분석하였다. 본 연구에서는 자기효능감을 진단하기 위해 O'Neil과 Herl[14]이 개발한 자기조절 측정문항 중에서 자기효능감의 영역만을 번안하여 설문하였으며, Likert 4점 척도로 측정하였다.

### 4.2 연구 설계

본 연구에서는 실험집단과 통제집단이 서로 다른 반으로 구성되어 있으므로 이질집단 사전사후 검사 방법을 사용하였다. 수업처치 전에 두 집단의 프로그래밍 능력과 자기효능감 수준을 비교하기 위해 사전검사를 실시하였다. 사전검사 후에 실험집단은 동료 에이전트 학습 시스템으로 수업

을 진행하였으며, 통제집단은 전통적인 방식인 강의법과 실습 위주의 수업을 진행하였다. 주당 2시간씩 4주간에 걸쳐 실시하였으며, 수업 처치 후 두 집단의 프로그래밍 능력과 자기효능감 수준을 분석하기 위해 사후 검사를 실시하였다. 본 연구의 실험 설계는 <표 3>과 같다.

<표 3> 실험 설계

G <sub>1</sub>	O <sub>1</sub> O <sub>2</sub>	X	O <sub>3</sub> O <sub>4</sub>
G <sub>2</sub>	O <sub>5</sub> O <sub>6</sub>		O <sub>7</sub> O <sub>8</sub>

G<sub>1</sub> : 실험집단  
 G<sub>2</sub> : 통제집단  
 X : 동료 에이전트 학습 시스템 적용 수업  
 O<sub>1</sub>, O<sub>5</sub> : 학업성취도 사전검사  
 O<sub>2</sub>, O<sub>6</sub> : 자기효능감 사전검사  
 O<sub>3</sub>, O<sub>7</sub> : 학업성취도 사후검사  
 O<sub>4</sub>, O<sub>8</sub> : 자기효능감 사후검사

## 5. 연구 결과

### 5.1 학업성취도 변인

본 연구에서는 수업 처치 전에 두 집단의 학업성취도 수준이 동일한지 검증하기 위해 사전 검사를 실시하였다. 사전검사는 학생들의 프로그램 과목 중간고사 성적으로 분석하였다. 사전검사에 대한 t-test 결과는 <표 4>와 같다.

<표 4> 학업성취도 사전검사

구분	사례수	평균	표준편차	t	p
통제집단	29	58.641	14.3274	.656	.515
실험집단	27	56.105	14.6201		

학업성취도 수준에 대한 사전검사 결과, 실험 집단과 통제집단 간에는 통계적으로 유의미한 차이를 보이지 않았다(p<.05). 이는 학업성취도에 있어서 두 집단이 통계적으로 차이가 없으며, 동질집단임을 의미한다.

4주간의 수업 처치 후 두 집단 간의 학업성취

도 수준을 비교하기 위해 사후검사를 실시하였다. 사후검사는 본 연구를 위해 제작된 문항으로 평가하였다. 사후검사에 대한 t-test 결과는 <표 5>와 같다.

<표 5> 학업성취도 사후검사

구분	사례수	평균	표준편차	t	p
통제집단	29	4.62	4.321	-2.279	.027
실험집단	27	7.67	5.636		

학업성취도에 대한 사후검사 결과, 실험집단 평균이 통제집단의 평균보다 높게 나타났으며, 통계적으로 유의미한 차이를 보이고 있다( $p < .05$ ). 이는 동료 에이전트를 통한 프로그래밍 수업이 학업성취도 향상에 긍정적인 영향을 준 것으로 볼 수 있다.

사후검사에 대한 두 집단의 차이에 대한 질적 분석을 위해 실험집단 학생들을 대상으로 집단심층면접을 실시하였다. 면접에 참여한 학생들은 에이전트가 제시하는 학습내용보다는 에이전트와 같이 역할을 분담하여 진행하는 과정이 많은 도움이 되었다고 한다. 일반적인 수업과는 다르게 코딩의 역할을 하거나 순서도 작성을 통한 알고리즘 구성의 노력이 프로그래밍의 과정을 이해하는데 도움이 되었으며, 디버깅을 하는 과정에서 제시되는 피드백이 수준별로 제시되어 스스로 문제를 해결하는 데 도움이 되었다고 말하였다.

## 5.2 자기효능감 변인

본 연구에서는 수업 처치 전에 두 집단의 자기효능감 수준이 동질한 지 분석하기 위해 사전검사를 실시하였다. 사전검사에 대한 t-test 결과는 <표 6>과 같다.

자기효능감에 대한 사전검사 결과, 실험집단과 통제집단 간에는 통계적으로 유의미한 차이를 보이지 않았다( $p < .05$ ). 이는 자기효능감 수준에 있어서 두 집단이 통계적으로 차이가 없으며, 동질 집단임을 의미한다.

<표 6> 자기효능감 사전검사

구분	사례수	평균	표준편차	t	p
통제집단	29	14.66	4.073	.280	.781
실험집단	27	14.33	4.540		

4주간의 수업 처치 후 두 집단 간의 자기효능감 수준을 비교하기 위해 사후검사를 실시하였다. 사후검사에 대한 t-test 결과는 <표 7>과 같다.

<표 7> 자기효능감 사후검사

구분	사례수	평균	표준편차	t	p
통제집단	29	14.79	4.057	-2.112	.039
실험집단	27	17.07	4.018		

자기효능감에 대한 사후검사 결과, 실험집단 평균이 통제집단의 평균보다 높게 나타났으며, 통계적으로 유의미한 차이를 보이고 있다( $p < .05$ ). 이는 동료 에이전트를 통한 프로그래밍 수업이 자기효능감 향상에 긍정적인 영향을 준 것으로 보인다.

사후검사에 대한 두 집단의 차이에 대한 질적 분석을 위해 실험집단 학생들을 대상으로 집단심층면접을 실시하였다. 면접에 참여한 학생들은 디버깅의 과정을 매우 어려워했다. 디버깅의 과정에서 에이전트가 제시하는 용기와 칭찬 메시지가 자신감을 얻는데 도움이 되었다. 또한 에이전트가 학습내용을 전달하고 문제를 제시하는 것만 아니라 동료의 역할처럼 학생에게 가르침을 받으려고 하거나 도움을 구하는 과정이 문제를 풀고 해결했을 때 많은 만족감을 얻었다고 하였다.

에이전트를 통한 학습 시스템은 학습자의 인지적 영역과 더불어 자기효능감과 같은 정의적인 영역에 있어서도 긍정적인 영향을 주는 것으로 나타났다.

## 6. 결 론

최근 컴퓨터 과학 분야 및 국가, 사회적 요구에 따라 프로그래밍 교육에 대한 다양한 연구가 시도되고 있으며 이러한 연구의 일환으로 본 연구에서는 프로그래밍 교육을 위한 동료 에이전트를 개발하여 학습자가 에이전트와 함께 프로그래밍을 역할에 따라 학습할 수 있게 하였다. 에이전트는 학습자의 수준을 분석하여 가장 적합한 피드백을 제공하고, 학습 목표 수준에 미달할 경우 추가적인 보충학습을 제공하도록 하였다. 본 연구에서 구현된 에이전트는 학습자와 같은 동료로서 학습자에게 학습을 제공하거나 제공받는 역할을 수행하였다.

이러한 에이전트 기반 학습 시스템을 실제 프로그래밍 교육에 적용한 결과 학업성취도면에서 유의미한 효과를 보여주고 있으며, 자기효능감면에 있어서도 유의미한 효과를 보여주고 있다. 이는 동료 에이전트가 프로그래밍 교육에서 인지적인 영역과 정의적인 영역 모두에 있어 긍정적인 영향을 주는 것으로 볼 수 있다. 추후 동료 에이전트의 역할 모델을 통해 에이전트가 가져야 할 교수적 방법에 대해 보다 깊은 연구가 필요하며, 프로그래밍을 통한 문제해결력 신장 방안에 대해 연구하고자 한다.

## 참 고 문 헌

- [1] 교육인적자원부(2007). 초·중등학교 교육과정, 교육인적자원부 고시 제 2007 - 79호.
- [2] 김미량(2002). 컴퓨터 프로그래밍 교육에 적용 가능한 효과적 교수방법의 탐색적 대안. 한국컴퓨터교육학회 논문지, 5(3), pp. 1-9.
- [3] 김은순, 박병호, 허희옥(2004). 컴퓨터 프로그래밍 교육에서 동료 지도학습이 학업성취도와 교수관계 형성에 미치는 영향, 7(4), pp. 111-120.
- [4] 서희전, 문경애(2005). 상호작용 촉진을 위한 협력학습지원 에이전트. 정보처리학회논문지, 12-A(6), pp. 547-556.
- [5] 이인숙, 임정훈(2005). e-Learning 기반 협력 학습을 위한 에이전트의 역할과 속성, 그리고 에이전트 시스템 설계 탐색. 교육정보미디어연구, 11(4), pp. 5-31.
- [6] 한건우, 이은경, 이영준(2006). Pair Programming이 학업성취도와 학습동기전략에 미치는 영향, 9(6), pp. 19-28.
- [7] 한선관, 이철환, 김영기(2003). 지능형 교수시스템의 학습자 모듈에 관한 연구. 인천교육대학교 과학교육논총, 15, pp. 369-392.
- [8] Kumar, A. N.(2005). Generation of problems, answers, grade, and feedback-case study of a fully automated tutor. Journal on Educational Resources in Computing, 5(3).
- [9] Butz, C. J., Hua, S. & Maguire, R. B.(2004). A Web-based Intelligent Tutoring System for Computer Programming. IEEE/WIC/ ACM Conference on Web Intelligence.
- [10] Conati, C. & Zhao, X.(2004). Building and Evaluating an Intelligent Pedagogical Agent to Improve the Effectiveness of an Educational Game. International Conference on Intelligent User Interfaces, pp. 6-13.
- [11] Gallis, H., Arisholm, E. & Dyba, T.(2002). A Transition from Partner Programming to Pair Programming - An Industrial Case Study. ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications, pp. 1-8.
- [12] White, G. L. & Sivitanides, M. P.(2002). A Theory of the Relationships between Cognitive Requirements of Computer Programming Languages and Programmer's Cognitive Characteristics. Journal of Information Systems Education, 13(1), pp. 59-66.
- [13] Han, K. W., Lee, E. K. & Lee, Y. J.(2006). Programming Teaching Support System using Student Model. The 14th International Conference on Computers in Education, Beijing, China, pp. 71-74.
- [14] O'Neil, H. F., & Herl, H. E.(1998). Reliability and validity of a trait measure



of self-regulation. Presented at the annual meeting of the American Educational Research Association, San Diego, CA.

- [15] Smith, T., Affleck, G., Lees, B. & Branki, C.(1999). Implementing a Generic Framework for a Web-based Pedagogical Agent. The 16th Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education.
- [16] Srikanth, H., Williams, J. G., Wiebe, E., Miller, C. & Balik, S.(2004). On Pair Rotation in the Computer Science Course. Proceedings of the 17th Conference on Software Engineering Education and Training, pp. 144-149.
- [17] Wiedenbeck, S.(2005). Factors Affecting the Success of Non-Majors in Learning to Program. The International Computing Education Research, pp. 13-24.

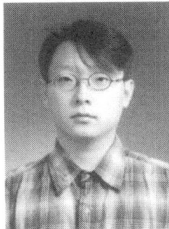
### 이 영 준



1988 고려대학교 전산과학과 (이학사)  
 1994 미국 미네소타대학교 (전산학 Ph.D.)

현재 한국교원대학교 컴퓨터교육과 교수  
 관심분야: 정보통신교육, 지능형시스템, 학습과학  
 E-Mail: yjlee@knue.ac.kr

### 한 건 우



1998 한국교원대학교 컴퓨터교육과(교육학학사)  
 2004 한국교원대학교 컴퓨터교육과(교육학석사)

2004~현재 한국교원대학교 컴퓨터교육과 박사과정

관심분야: 프로그래밍 교육, 교수 에이전트  
 E-Mail: flatfish@paran.com

### 이 은 경



1998 한국교원대학교 컴퓨터교육과(교육학학사)  
 2005 한국교원대학교 컴퓨터교육과(교육학석사)

2006~현재 한국교원대학교 컴퓨터교육과 박사과정

관심분야: u-러닝, 학습과학, 로봇 기반 학습  
 E-Mail: soph76@hitel.net