

Pair Programming이 학업성취도와 학습동기전략에 미치는 영향

한건우[†] · 이은경[†] · 이영준^{†*}

요 약

프로그래밍 학습은 컴퓨터 과학에 있어 매우 중요한 과목이지만, 대부분의 학생들은 프로그래밍 배우는 것을 어려워하고 있다. 프로그래밍 교육에 대한 새로운 교수방법이나 기술적 관점의 연구가 필요한 시점이다. 따라서 본 연구에서는 Pair Programming을 통해 프로그래밍 교육의 가능성을 살펴보고, 교육적 효과를 높이기 위한 Pair Programming을 제안하였다. 제안된 교수-학습 전략을 실험한 결과, 학업성취도와 학습전략에 있어 효과가 있는 것으로 나타났다. 특히 학습전략에 있어서는 비판적 사고, 메타인지, 노력조절, 동료학습 영역들이 긍정적인 영향을 주는 것으로 나타났다.

키워드 : 프로그래밍 교육, 동료(작) 프로그래밍, 학습동기전략

The Effects of Pair Programming on Achievement and Motivated Strategies in Programming Course

Keun-Woo Han[†] · Eun-Kyoung Lee[†] · YoungJun Lee^{†*}

ABSTRACT

A programming language is very important subject in computer science. But it is difficult for many students to learn programming language. New pedagogical methods and techniques to help students to learn programming are needed. In this paper, we investigated the effects of Pair Programming, and we proposed a Pair Programming strategy to improve programming skills. This paper examined the proposed Pair Programming strategy. A empirical study showed the proposed teaching and learning strategy has positive effects on achievement and motivated strategies in programming course. Four factors - critical thinking, metacognitive reflection, effort regulation, and peer learning - have positive effects in learning strategies. The proposed Pair Programming strategy improves programming learning and motivated strategies.

Keywords : Programming Education, Pair Programming, Motivated Strategies

1. 서 론

프로그래밍 언어는 인간과 컴퓨터 간의 매개체 역할을 해주며, 인간은 프로그래밍 언어를 사용

하여 현실 세계에서 일어나는 문제들을 해결한다. 프로그래밍 언어는 컴퓨터 분야에 있어 주요 영역이며, 대학교의 관련 학과나 IT 산업 분야에서도 커다란 영역을 차지하고 있다. 이러한 사회적, 시대적 중요성에 의해 실업계 고등학교를 중심으로 프로그래밍 관련 교육과정이 운영되고 있으며, 일반계 고등학교에서도 제7차 교육과정 개

[†] 종신회원: 한국교원대학교 컴퓨터교육과 박사과정
^{**} 종신회원: 한국교원대학교 컴퓨터교육과 교수(교신저자)
논문접수: 2006년 10월 30일, 심사완료: 2006년 11월 24일

정안을 통해 컴퓨터 과학을 기반으로 한 프로그래밍 교육에 관심을 나타내고 있다[4].

하지만 대부분의 학생들은 프로그래밍 언어 배우기를 어려워하며, 교사 또한 프로그래밍 과목은 가르치기 어려운 과목이라고 한다. 왜냐하면 프로그래밍은 논리적인 사고를 바탕으로 구현되어야 하므로 학습자는 복잡한 인지적 기능을 사용해야 하기 때문이다. 또한, 프로그래밍 언어는 기본적으로 논리적인 개념에 바탕을 두고 있어 학생들에게 논리적인 사고와 추상적인 추론 능력을 요구하고 있다[8].

프로그래밍 교육 연구 분야에서는 이러한 문제점을 극복하고자 프로젝트 기반 학습, 동료 교수법, 구성주의적 접근 등의 다양한 교수-학습 전략을 도입하여 학습 효과를 높이고자 하였다[3, 5, 7, 9].

프로그래밍은 논리적인 사고를 바탕으로 이루어지기 때문에 수학이나 과학 같은 과목의 성적이 프로그래밍 능력과 관련이 있는 것으로 나타났다[11], 자기효능감(self-efficacy)이나 동기와 같은 요소가 프로그래밍에 영향을 주는 것으로 나타났다[15]. 또한, 프로그래밍을 배우는 학습자는 프로그래밍 언어에 대한 개념과 함께 프로그래밍 기술을 익혀야 하므로 이를 돕기 위한 교수-학습 전략이 요구된다.

이러한 프로그래밍 교육 관점에서 보면 Pair Programming은 좋은 대안이 될 수 있다. Pair Programming이란 두 사람이 하나의 키보드, 마우스, 모니터를 사용하여 프로그램 하는 기법으로 산업 현장이나 교육 환경에서 즐겁게 프로그래밍을 수행할 수 있으며, 프로그래밍 코드와 결과물의 질적 수준도 높여주는 방법이다[14, 16]. Pair Programming은 학습자에게 자신감을 증진시켜주며, 동료 교수 전략을 통한 학습 효과를 볼 수 있다[17].

일반적으로 Pair Programming은 학업성취도나 자기효능감 등에는 긍정적인 영향을 주는 것으로 나타났다. 그러나 Pair Programming은 학습을 위해 제작된 것이 아닌 소프트웨어 개발의 효율성을 높이기 위해 만들어진 것이다. 따라서 교육적 효과를 높이기 위해 수정이 불가피하며, 다양한 학습 효과에 대해 연구할 필요가 있다.

따라서 본 연구에서는 프로그래밍 교육에 있어서 Pair Programming이 학습자의 학업 성취도와 학습 동기 전략의 효과를 살펴보고, 이들 간의 관계를 분석하여 Pair Programming의 개선 방법에 대해 제시하였다.

2. 이론적 배경

2.1 Pair Programming

Pair Programming은 두 사람이 하나의 컴퓨터를 사용하여 협력적 분업의 형태로 프로그래밍하는 것을 말하며, Extreme Programming(XP) 개발 절차 과정 중의 한 가지이다. XP는 최근 개발방법론 중에서 급부상하고 있는 소프트웨어 개발론의 하나이다. XP에서는 19가지의 방법을 제시하고 있으며, 그 중 프로그래머의 연습 과정에 Pair Programming을 제시하고 있다.

Pair Programming은 두 명의 프로그래머가 하나의 컴퓨터에 같이 앉아 프로그램을 설계, 알고리즘, 코딩, 디버깅 등을 협업하는 형태이다[16]. 동료 중 한 사람은 Driver의 역할을 수행한다. Driver는 컴퓨터의 키보드와 마우스를 주로 사용하는 사람으로 코딩 업무를 담당하며, 설계를 작성한다. 다른 동료 한 사람은 Navigator의 역할을 수행한다. Navigator는 전략과 전술을 검색하거나 문법적인 오류, 타이핑 실수 등을 찾아주어 Driver의 작업을 보완하여 준다. Driver는 Navigator가 제시한 전략에 대해 질문하고 이에 대해 설명을 들을 수 있으며, Navigator는 프로그래밍 작업의 방향에 대해 보다 객관적인 관점으로 접근할 수 있다. Pair Programming의 역할에 따른 작업을 간략하게 정리하면 <표 1>과 같다.

<표 1> Pair Programming에서의 역할

역할	작업	비고
Driver	키보드와 마우스로 작업하기 프로그래밍 코딩 작업하기 코딩 설계하기	상호간의 의사소통하기
Navigator (Observer)	코딩 상의 오타 찾아내기 문법적인 오류 찾아내기 필요한 알고리즘 구상하기	동료 간의 역할 교환하기

동료들은 브레인스토밍 과정을 거쳐 서로의 문제점을 보완하거나 새로운 아이디어를 구안할 수 있다. Pair Programming은 프로그래밍에 대한 능력을 높여주며, 동료와의 상호작용을 통해 프로그래밍에 대한 학습 효과를 볼 수 있다.

2.2 Pair Programming에 관한 선행연구

Pair Programming의 과정을 통해 얻을 수 있는 다양한 교육적 효과를 살펴보기 위해 선행 연구물을 분석하였다. Pair Programming은 프로그래밍 결과물에 대한 질적 향상과 더불어 부가적인 효과가 나타나는 것으로 분석되었다.

Cockburn과 Williams는 Pair Programming의 비용에 대해 조사한 결과 설계의 질, 결점 감소, 기술력 증가, 팀 구성원간의 대화가 향상된 것으로 나타났다. 또한 보다 즐겁게 작업을 한 것으로 나타나 만족도에 있어서도 유의미한 차이를 보였다[6]. Williams와 Kessler는 컴퓨터 과학 수업에 Pair Programming 전략을 도입하였다. 그 결과 혼자 프로그래밍을 한 집단보다 새로운 언어를 빠르게 습득하였으며, 실습조교의 역할이 줄어들었다고 보고한다[16]. 이들 해외의 연구사례는 프로그래머나 대학생을 대상으로 실시되었으며, Pair Programming을 통한 교육적 효과보다는 소프트웨어 개발론적 입장에서의 프로그래밍 생산 능력에 중점을 두고 연구되었다.

국내에서는 Pair Programming을 실업계 고등학교에 적용하여 교육적 가능성을 제시해 주고 있다. 고일제는 비주얼 베이직 언어 교육에 Pair Programming을 적용한 결과, 학업 성취도와 만족도에 있어 유의미한 결과를 보이고 있다[1]. 그러나 학습 진행과정에 있어 Pair Programming에 대한 정확한 안내 없이 단순히 Pair Programming 과정을 수업 적용하였으며, 학생들의 프로그래밍 수준을 고려하지 않고 동료를 선정하였다. 보다 체계적이고 효과적인 Pair Programming 수업이 되기 위해서는 동료선정에 대해 고려해야 한다. Schneider와 Johnston은 Pair Programming에서 프로그래머 간의 수준 차이로 인해 잘 하는 사람은 작업을 많이 하게 되

고, 못하는 사람은 학습 효과가 없는 것으로 보고한다. 따라서 동료가 잘 선정된 팀만 효과가 나타난다는 것을 지적하고 있다[13]. Pair Programming은 처음 접하는 사람에 있어 이를 이해하기 위한 사전 연습의 과정이 요구됨을 알 수 있다. 또한, 동료의 프로그래밍 수준이 비슷해야 활발한 상호작용이 이루어지며, 동료의 역할에 대한 안내가 필요함을 알 수 있다.

아울러 고일제의 연구에서는 프로그래밍 교과에 대한 교과 내용 분석을 통해 과제 선정의 개선이 필요하다고 지적하였다. 이는 기존의 선행연구가 가지고 있는 가장 큰 문제점이라고 할 수 있다. 기존의 연구들은 수업에 단순히 Pair Programming의 과정을 적용하여 프로젝트를 완성하거나 문제를 해결하여 결과를 산출하는데 중점을 두고 있다. 따라서 Pair Programming을 교육현장에서 효과적으로 활용하기 위한 수업전략이 모색되어야 하며, 이를 통해 얻을 수 있는 교육적 효과에 대해 자세히 분석할 필요가 있다.

3. Pair Programming 수업 전략 설계

3.1 수업설계 방향

선행연구를 살펴볼 때 Pair Programming은 소프트웨어 개발론적인 입장에서 개발된 것으로 프로그램 생산 능력에 중점을 두고 있다. 그러나 여러 연구에서 교육적 활용 가능성을 시사하고 있다. 따라서 현행 프로그래밍 교육과정에 Pair Programming을 적용하기 위해서는 다음과 같은 점을 고려한 수업 전략을 구성해야 한다.

첫째, Pair Programming에 대한 안내가 요구된다. Pair Programming에 대한 정확한 이해가 있어야 하며, Driver와 Navigator에 대한 역할에 대해 이해할 수 있도록 다양한 선행 연습이 필요하다.

둘째, 동료 선정에 관심을 가져야 한다. 동료 선정이 잘 되어야 좋은 결과가 나타나므로 이에 대해 고려해야 한다. 동료 간에 수준 차이가 발생하면 잘하는 학생에 의해 작업이 주도되므로 수준이 유사한 학생들을 동료로 선정해야 한다.

셋째, 프로그래밍 교육과정을 통해 프로그래밍을 효과적으로 학습하기 위한 전략이 요구된다. 프로그래밍 교육을 위해 단순히 Pair Programming을 도입하기에는 문제점이 있다. 현재 중등 교육과정에 대한 이해와 학생들의 수준, 교육 내용을 감안하여 새로운 적용 방안이 모색되어야 한다.

넷째, Pair Programming을 통해 나타나는 교육적 효과를 분석하기 위해 학업 성취도와 학습 전략 및 동기를 살펴볼 필요가 있다. 본 연구에서 제안된 수업 전략이 학업 성취도에 대해 긍정적인 영향을 주는지 살펴보아야 한다. 아울러 프로그래밍 학습은 자기효능감이나 동기와 같은 요소에 영향을 받는 것으로 나타났다. Pair Programming은 동료 및 협동 학습적 요소를 가지고 있으므로, 학습자들 간의 상호작용으로 인해 학습전략과 동기에 영향을 줄 것으로 기대된다. 따라서 이에 대한 교육적 효과를 면밀히 분석할 필요가 있다.

3.2 수업설계안

Pair Programming은 소프트웨어 공학 분야에서 프로그래밍 개발 방법론에 중점을 두고 있다. 그러므로 이를 교육적 목적으로 활용하기 위해서는 일부 내용을 수정할 필요가 있다. 또한, Pair Programming의 교육적 효과에도 불구하고 발생될 수 있는 문제점을 극복하기 위해 프로그래밍 교육 환경에 적합한 Pair Programming 수업 전략이 필요하다.

본 연구에서는 Pair Programming에서 발생하는 문제점을 해결하기 위해 교수-학습 전략을 수립하였다. 교육용 Pair Programming의 수업 전략은 크게 세 단계로 구분하였으며, 각 단계별 전략 내용은 다음과 같다.

첫 번째 단계는 Pair Programming에 대한 수업 안내이다. Pair Programming은 두 사람이 하나의 컴퓨터를 가지고 진행하며, 각자의 역할에 따라 수업이 이루어진다. 그러나 처음 접하는 학생들은 역할에 대한 이해하기 어려워 원활한 수업이 어려울 수 있다. 따라서 Pair Programming

에 대한 각자의 역할을 이해할 수 있도록 수업 과정을 예시로 보여주며, Driver와 Navigator의 역할을 실습해 보도록 구성하였다. 아울러 진단 평가를 통해 교사가 학생들의 수준을 파악하고 프로그래밍 수준이 유사한 학생들을 동료로 선정하였다.

두 번째 단계는 프로그래밍 학습을 위한 실습이다. 프로그래밍 학습에 있어서 좋은 학습 예제는 필수적인 요소라고 할 수 있다[13]. 학생들의 수준을 고려하여 단계적으로 실습예제 제시하였다. 학생들이 예제를 통해 학습한 후에는 코드를 확장하고, 수정하여 스스로 다양한 코드를 생성하고 개념을 정립하는 과정(Navigator의 역할)을 제시하였고, 주석문을 통한 문서화 작업, 코딩 간결화(Driver의 역할)의 과정을 제시하였다. 아울러 역할을 바꾸어가며 유사한 예제를 해결하도록 수행하여 Driver와 Navigator의 역할을 골고루 실습할 수 있도록 하였다.

세 번째 단계는 평가 단계이다. 지금까지의 학습 과정을 통해 배운 프로그래밍 개념을 스스로 정리하여 학습에 대한 이해도를 살펴보도록 하였다. 아울러 Pair Programming을 통해 문제를 해결하는 과정에서 각자의 역할을 충실히 수행하였는지에 대해 동료 평가를 실시하였다.

교육용 Pair Programming의 수업 전략을 정리하여 제시하면 <표 2>와 같다.

<표 2> 교육용 Pair Programming 수업 전략

절차	전략
1. 수업 안내	<ul style="list-style-type: none"> • Pair Programming 소개 - 수업 전략에 대한 소개 및 역할 안내하기 • Pair Programming 시연 - 역할에 대한 시연으로 학습자의 이해 돕기 • Pair Programming 실습 - 실습을 통한 역할 모델 체험하기 • 진단 평가하기 - 실습을 통해 학생들의 수준을 파악하여 동료 선정하기
2. 실습하기	<ul style="list-style-type: none"> • 예제 실습하기 - 좋은 예제코드를 따라서 코딩하기 - 쉽고 간결한 예제를 중심으로 제시하기 - 코드에 대한 문서화 작업하기 - 예제 코드를 간결하게 수정하기 - 다른 명령어로 수정하기 • 역할 교환하기 - Driver와 Navigator 역할을 교환하여 작업하기
3. 평가하기	<ul style="list-style-type: none"> • 학습내용 스스로 정리하기 - 분별적인 내용을 서로 의견교환을 통해 정리하여 보고서 제출하기 • 동료 평가하기 - 동료의 역할과 작업에 대해 평가하기

4. 연구방법

4.1 연구대상 및 검사도구

본 연구는 프로그래밍 교육을 실시하는 실업계 고등학교 4개 반의 학생 172명을 대상으로 하였으며, 웹 프로그래밍 과목에서 ASP(Active Server Pages)의 학습내용을 적용하였다. 이 중 임의의 2개 반 63명의 학생은 실험집단으로 선정하여 Pair Programming 전략으로 수업 하였으며, 나머지 2개 반 69명의 학생은 통제집단으로 선정하여 전통적인 강의법 및 실습·시연 수업을 실시하였다. 본 연구에 참여한 학생들은 모두 프로그래밍에 대한 경험이 있는 집단이며, 집단별로 같은 교사가 수업을 실시하였다.

학생들의 프로그래밍 능력과 학습동기·전략 수준을 검사하기 위해 수행 평가 점수와 학습동기·전략 검사지(MSLQ: Motivated Strategies for Learning Questionnaire)를 사용하였다. MSLQ는 Pintrich & De Groot[12]에 의해 개발된 것으로 본 연구에서는 권용선[2]이 동 검사지를 번안하여 전문가 2인에게 타당도를 검증받은 검사지를 사용하였다.

<표 3> 학습동기·전략 검사지(MSLQ) 항목

구분	측정항목	문항수
동기 변인	내적목표	4
	외적목표	4
	과제인식	6
	학습신념	4
	자기효능감	8
	시험불안	5
학습전략 변인	시연	4
	정교화	6
	조직화	4
	비관적 사고	5
	메타인지	12
	학습시간관리	8
	노력조절	4
	동료학습	3
	도움구하기	4
	합계	

MSLQ는 총 81개 문항으로 구성되어 있으며, 동기 변인을 측정하기 위한 31 문항과 학습전략 변인을 측정하기 위한 50개의 문항으로 분류되어 있다. 각 문항 당 Likert 5점 척도로 81문항을 제시하여 405점 만점으로 평가하였다. 본 검사지의 세부적인 측정 항목 및 문항 수는 <표 3>과 같다. 학습성취도 평가는 중간고사(사전 검사)와 기말고사(사후 검사)의 수행평가 점수를 활용하였다. 수행평가는 ASP의 프로그래밍 요소로 구성되었으며, 40점 만점으로 평가하였다.

4.2 연구 설계

본 연구에서는 실험집단과 통제집단의 학습성취도와 자기조절학습능력을 측정하여 분석하였다. 수업처치 전에 두 집단의 수준을 비교하기 위해 사전 검사를 실시하였다. 학습성취도는 웹 프로그래밍 과목의 중간고사 수행평가 점수를 사용하였으며, 설문지를 통해 MSLQ를 측정하였다. 이후, 실험집단은 비슷한 수준의 학생들을 동료로 묶어주어 Pair Programming 수업을 진행하였으며, 통제집단은 전통적인 방식으로 강의법과 혼자 실습하는 형태로 진행하였다. 주당 3시간씩 4주간의 수업을 동일한 교사가 실험 처치하였다. 처치 후 두 집단의 학습 능력, 동기·전략의 영향을 분석하기 위해 사후 검사를 실시하였다. 학습성취도는 기말고사의 수행평가점수(40점)를 사용하였으며, 설문지를 통해 MSLQ를 측정하였다. 본 연구의 실험 설계는 <표 4>와 같다.

<표 4> 실험 설계

G ₁	O ₁	X ₁	O ₂
G ₂	O ₁	X ₂	O ₂

G₁ : 실험집단

G₂ : 통제집단

X₁ : Pair Programming 수업

X₂ : 전통적 수업

O₁ : 사전 검사(학습성취도, MSLQ)

O₂ : 사후 검사(학습성취도, MSLQ)

아울러 실험 종료 이후 성적과 설문을 분석한 결과를 토대로 실험집단 학생을 대상으로 인터뷰를 진행하였다.

5. 연구 결과

5.1 학업성취도 변인

본 연구에서는 수업 처치 전에 두 집단의 학업성취도 수준이 동질인지 검증하기 위해 사전 검사를 실시하였다. 사전검사에 대한 t-test 결과는 <표 5>와 같다.

<표 5> 학업성취도 사전검사

구분	사례수	평균	표준편차	t	p
실험집단	63	26.81	8.784	.840	.403
통제집단	69	25.49	9.187		

사전검사 결과, 실험집단과 통제집단은 통계적으로 유의미한 차이를 보이지 않았다($p < .05$). 이는 학업성취도에 있어 두 집단이 통계적으로 차이가 없으며, 동질집단임을 확인하였다.

4주간의 수업 처치 후 두 집단 간의 학업성취도 차이를 분석하기 위해 사후 검사를 실시하였다. 사후검사에 대한 t-test 결과는 <표 6>과 같다.

<표 6> 학업성취도 사후검사

구분	사례수	평균	표준편차	t	p
실험집단	63	32.90	11.535	2.120	.036
통제집단	69	28.25	13.459		

사후검사 결과, 실험집단 평균이 32.90점으로 통제집단의 평균 28.25점보다 높게 나타났으며, 통계적으로 유의미한 차이를 보이고 있다($p < .05$). 즉, Pair Programming 전략 수업이 학업성취도 향상에 영향을 주는 것으로 볼 수 있다.

5.2 학습동기·전략 변인

본 연구에서는 수업 처치 전에 두 집단의 학습동기·전략 수준이 동질인지 검증하기 위해 사전 검사를 실시하였다. 사전검사에 대한 t-test 결과는 <표 7>과 같다.

<표 7> 학습동기·전략 사전검사

구분	사례수	평균	표준편차	t	p
실험집단	63	235.83	23.605	-1.171	.864
통제집단	69	236.70	33.506		

사전검사 결과, 실험집단과 통제집단은 통계적으로 유의미한 차이를 보이지 않았다($p < .05$). 이는 학습동기·전략에 있어 두 집단이 통계적으로 차이가 없으며, 동질집단임을 확인하였다.

4주간의 수업 처치 후 두 집단 간의 학습동기·전략 차이를 분석하기 위해 사후 검사를 실시하였다. 사후검사에 대한 t-test 결과는 <표 8>과 같다.

<표 8> 학습동기·전략 사후검사

구분	사례수	평균	표준편차	t	p
실험집단	63	250.22	28.623	2.178	.031
통제집단	69	238.25	34.015		

사후검사 결과, 실험집단 평균이 250.22점으로 통제집단의 평균 238.25점보다 높게 나타났으며, 통계적으로 유의미한 차이를 보이고 있다($p < .05$). 즉, Pair Programming 전략 수업이 학습동기·전략 향상에 영향을 주는 것으로 볼 수 있다.

한편, MSQ 검사자의 하위 변인인 학습동기와 학습전략에 대한 총점으로 t-test를 실시하였다. 그 결과, 학습동기에는 유의미한 차이를 보이지 않았으나, 학습전략에는 유의미한 차이를 보이고 있다($p < .05$). 학습전략에 대한 사전검사와 사후검사의 t-test 결과는 <표 9>와 같다.

<표 9> 학습전략 t-test 결과

검사	구분	사례수	평균	표준편차	t	p
사전 검사	실험 집단	63	140.92	17.311	-7.32	.466
	통제 집단	69	143.43	21.688		
사후 검사	실험 집단	63	153.59	17.311	2.359	.020
	통제 집단	69	145.09	23.328		

학습동기와 학습전략 변인을 살펴본 결과, Pair Programming은 학습전략에 유의미한 영향을 주는 것으로 나타났다. 학습전략적인 측면에서 실험집단이 통제집단보다 높은 수준을 보이고 있다. 따라서 Pair Programming이 학습전략에 긍정적인 영향을 주는 것으로 볼 수 있다.

아울러 집단별로 사전 설문과 사후 설문의 학습동기·전략 평균 및 학습전략 평균에 대해 Paired Sample t-test를 추가로 분석한 결과, 실험집단은 사전, 사후 설문에 대해 유의미한 차이를 보였지만, 통제집단은 유의미한 차이를 보이지 않았다(p<.05).

5.3 학습전략 하위 항목

학습동기·전략 검사는 학습전략 변인에 대해 9가지 하위 항목(시연, 정교화, 조직화, 비판적 사고, 메타인지, 학습시간관리, 노력조절, 동료학습, 도움구하기)으로 구성되어 있다. 본 연구에서 유의미한 차이를 보인 학습전략의 하위 항목에 대해 t-test를 실시하였다.

학습전략 전체 문항에서는 유의미한 차이를 보이지 않았지만, 하위 항목을 t-test한 결과, 비판적 사고, 메타인지, 노력조절, 동료학습에 있어 유의미한 차이를 보이고 있다. 아울러 집단별로 사전 설문과 사후 설문의 학습전략 비판적 사고, 메타인지, 노력조절, 동료학습의 평균에 대해 Paired Sample t-test를 추가로 분석한 결과, 실험집단은 사전, 사후 설문에 대해 유의미한 차이를 보였지만, 통제집단은 유의미한 차이를 보이지 않았다(p<.05). 구체적인 사전, 사후 검사 결과는 다음과 같다.

비판적 사고에 대한 t-test 결과는 <표 10>과 같다.

<표 10> 학습전략의 비판적 사고 t-test 결과

검사	구분	사례수	평균	표준편차	t	p
사전 검사	실험 집단	63	13.38	3.245	-4.29	.668
	통제 집단	69	13.62	3.232		
사후 검사	실험 집단	63	15.70	2.637	2.789	.006
	통제 집단	69	14.13	3.682		

학습전략의 하위 항목인 비판적 사고 영역에서 두 집단이 유의미한 차이를 보였으며(p<.05), 실험집단이 통제집단보다 더 높은 수준을 보이고 있다. 실험집단 학생들을 대상으로 비판적 사고에 대해 의견을 인터뷰하였다. 이와 같은 결과 나타난 이유는 Pair Programming의 과정에서 발생하는 상호작용 때문인 것으로 보인다. 학생들은 서로의 역할에 따라 문제 해결의 방법이나 코드를 수행하였다. 학생들은 “어려가 왜 발생했지?”, “이 줄에 틀린 것이 뭐지?” 등의 대화를 나눈 것으로 나타났으며, 발생하는 문제점에 대해 서로 의견을 제시하고 논의하면서 문제를 해결해 나아갔다. 이러한 과정 속에서 비판적 사고 영역이 향상된 것으로 보인다.

메타인지에 대한 t-test 결과는 <표 11>과 같다.

<표 11> 학습전략의 메타인지 t-test 결과

검사	구분	사례수	평균	표준편차	t	p
사전 검사	실험 집단	63	33.29	4.524	.012	.991
	통제 집단	69	33.28	5.525		
사후 검사	실험 집단	63	36.30	5.830	2.144	.034
	통제 집단	69	34.06	6.159		

학습전략의 하위 항목인 메타인지 영역에서 두 집단이 유의미한 차이를 보였으며(p<.05), 실험집단이 통제집단보다 더 높은 수준을 보이고 있다.

실험집단 학생들을 대상으로 메타인지에 대해 의견을 인터뷰하였다. 학생들은 메타인지에 대해 정확하게 인지하고 있지는 않았지만, Pair Programming의 과정 속에서 현재 진행되고 있는 학습에 대해 서로 모니터링을 해 준 것으로 나타났다. 학생들은 “저번에 이거 배웠던가?”, “너는 이거 알아?”, “내가 한 것이 맞는 거야?” 등 학습에서 발생하는 것에 대해 서로의 지식을 평가해주고 점검해주면서 서로의 지식을 이해하려고 하였다. 이러한 과정은 서로 간의 대화를 원활하게 하기 위해 서로의 지식을 점검하는 과정이 요구되었기 때문인 것으로 나타났다. 이러한 자기평가 및 점검의 과정이나 지식의 구조화는 메타인지 영역에 있어서 매우 중요한 요소이므로 실험집단 학생이 향상된 것으로 보인다.

노력조절에 대한 t-test 결과는 <표 12>와 같다. 학습전략의 하위 항목인 노력조절 영역에서 두 집단이 유의미한 차이를 보였으며(p<.05), 실험집단이 통제집단보다 더 높은 수준을 보이고 있다. 실험집단 학생들을 대상으로 노력조절에 대해 의견을 인터뷰하였다. 학생들은 Pair Programming 과정에서 자신의 역할에 충실해야 수업이 진행되므로 이에 대해 일정부분의 책임감을 느낀 것으로 나타났다.

<표 12> 학습전략의 노력조절 t-test 결과

검사	구분	사례수	평균	표준편차	t	p
사전 검사	실험 집단	63	12.17	1.530	.546	.586
	통제 집단	69	12.03	1.534		
사후 검사	실험 집단	63	12.87	1.314	3.316	.001
	통제 집단	69	12.04	1.538		

따라서 수업의 진행과 문제를 해결하는 과정에서 자신의 역할을 해야만 했다. 그러나 일부 학생들은 자신의 역할을 포기하기도 하였으나 이 부분에 대해서는 교사가 지속적으로 지도하였으며, 아울러 동료평가를 통해 서로의 역할을 평가하도록 하였다.

동료학습에 대한 t-test 결과는 <표 13>과 같

다.

<표 13> 학습전략의 동료학습 t-test 결과

검사	구분	사례수	평균	표준편차	t	p
사전 검사	실험 집단	63	8.13	1.727	-1.756	.081
	통제 집단	69	8.74	2.221		
사후 검사	실험 집단	63	9.71	2.217	2.278	.024
	통제 집단	69	8.83	2.255		

학습전략의 하위 항목인 동료학습 영역에서 두 집단이 유의미한 차이를 보였으며(p<.05), 실험집단이 통제집단보다 더 높은 수준을 보이고 있다. 실험집단 학생들을 대상으로 동료학습에 대해 의견을 인터뷰하였다. Pair Programming은 서로에게 주어진 역할에 따라 학습을 진행하고 주어진 동료와 같이 문제를 해결해 나아가야 하므로 동료학습의 특성을 가지고 있다. 이에 대해 학생들도 유사한 의견을 제시하였다.

6. 결 론

본 연구에서는 프로그래밍 교육의 효과를 높이기 위한 방안으로 Pair Programming 기법을 활용한 수업 전략을 설계하였다. 기존의 Pair Programming 연구는 소프트웨어의 생산 능률적인 관점에서 비용, 시간 등의 효과에 집중하고 있다. 본 연구에서는 이를 통해 얻을 수 있으나 교육적 측면을 강조하여 Pair Programming 수업 전략을 개발하였다. 개발된 Pair Programming은 교육적인 측면에서 프로그래밍의 학습 효과를 높이고자 다양한 문제 해결의 과정을 제시하였으며, 평가의 과정으로 서로의 내용을 점검하도록 하였다.

연구 결과, 학업 성취도와 학습동기·전략에 있어서 긍정적인 효과가 나타났다. 학습전략의 하위 요소에서는 비판적 사고, 메타인지, 노력조절, 동료학습에 있어 유의미한 차이를 보였다. 본 연구에서 제안한 Pair Programming은 학습적인 효과 외에 자기조절의 영역인 학습전략에 있어서

도 긍정적인 영향을 주는 것으로 나타났다.

본 연구에서는 Pair Programming이 프로그래밍 교수법에 있어 새로운 가능성이 있음을 확인하였다. 앞으로는 Pair Programming을 활용한 에이전트를 통해 학습 효과를 높이는 교육 모델에 대해 연구할 필요가 있다.

참 고 문 헌

- [1] 고일재(2006). 작 프로그래밍(Pair Programming) 기법을 이용한 프로그래밍 수업이 학업성취와 학습흥미에 미치는 효과. 중앙대학교 대학원 석사학위 논문.
- [2] 권용선 (2001). 지식창출을 위한 지식공유 과정과 이에 영향을 미치는 요인: 소집단 사례 연구를 중심으로. 이화여자대학교 대학원 석사학위 논문.
- [3] 김미량(2002). 컴퓨터 프로그래밍 교육에 적용 가능한 효과적 교수방법의 탐색적 대안. 한국컴퓨터교육학회 논문지, 5(3), pp. 1-9.
- [4] 한국교육학술정보원(2006). 2006년도 e-러닝 정책 포럼 자료.
- [5] Charlie McDowell, Linda Werner, Heather E. Bullock, Julian Fernald(2003). The Impact of Pair Programming on Student Performance, Perception and Persistence. 25th International Conference on Software Engineering, pp. 602-607.
- [6] Cockburn, A. & Williams, L.(2000). The Costs and Benefits of Pair Programming. eXtreme Programming and Flexible Processes in Software Engineering - XP2000.
- [7] David Davenport(2000). Experience Using a Project-Based Approach in an Introductory Programming Course, IEEE Transactions on Education, 43(4), pp. 443-448.
- [8] Garry L. White & Marcos P. Sivitanides(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.
- [9] Mark J. Van Gorp & Scott Grissom(2001). An Empirical Evaluation of Using Constructive Classroom Activities to Teach Introductory Programming. Computer Science Education, 11(3), pp. 247-260.
- [10] Nicolas Guibert, Partrick Girard (2003). Teaching and Learning Programming with a Programming by Example System. International Symposium on End User Development.
- [11] Pat Byrne & Gerry Lyons(2001). The Effect of Student Attributes on Success in Programming. Conference on Innovation and Technology in Computer Science Education, pp. 49-52.
- [12] Pintrich, P. R., & De Groot, E. V. (1990). Motivational and Self-Regulated Learning Components of Classroom Academic Performance. Journal of Educational Psychology, 82(1), pp. 33-40.
- [13] Schneider, J.G. & Johnston, L.(2003). eXtreme Programming at Universities - An Educational Perspective. Proceedings of 25th International Conference on Software Engineering, Portland, Oregon, pp. 594-599.
- [14] Shaochun Xu & Václav Rajlich(2005). Pair Programming in Graduate Software Engineering Course Projects. ASEE/IEEE Frontiers in Education Conference.
- [15] Susan Wiedenbeck(2005). Factors Affecting the Success of Non-Majors in Learning to Program. The International Computing Education Research, pp. 13-24.
- [16] Williams, Laurie & Rovert R. Kessler(2000). Experimenting with Industry's "Pair-Programming" Model in the Computer Science Classroom. Journal of Computer Science Education, 10(4).
- [17] Williams, Laurie, Kai Yang, Eric Wiebe, Miriam Ferzli, Carol Miller(2002). Pair Programming in an Introductory computer Science Course: Initial Results and Recommendations. ACM Conference on Object-Oriented Programming, Systems,

Languages, and Applications, pp. 20-27.



한 건 우

1998 한국교원대학교
컴퓨터교육과(교육학학사)

2004 한국교원대학교
컴퓨터교육과(교육학석사)

2004~현재 한국교원대학교 컴퓨터교육과
박사과정

관심분야: 프로그래밍교육, 교수 에이전트

E-Mail: flatfish@paran.com



이 은 경

1998 한국교원대학교
컴퓨터교육과(교육학학사)

2005 한국교원대학교
컴퓨터교육과(교육학석사)

2006~현재 한국교원대학교 컴퓨터교육과
박사과정

관심분야: 컴퓨터교육, 유비쿼터스, 교육용 로봇

E-Mail: soph76@hitel.net



이 영 준

1988 고려대학교 전산과학과
(이학사)

1994 미국 미네소타대학교
(전산학 Ph.D.)

현재 한국교원대학교 컴퓨터교육과 교수

관심분야: 정보통신, 지능형 시스템, 컴퓨터교육

E-Mail: yjlee@knue.ac.kr