

## 심화·보충형 수준별 학습에 의한 컴퓨터 프로그래밍 영역별 학습 효과 분석

안 유정\*, 김경아\*\*

### An Analysis of Learning Effect by Computer Programming Areas based on Leveled Intense and Supplementary Learning

You Jung Ahn\*, Kyung-Ah Kim\*\*

#### 요 약

본 연구에서는 효과적인 컴퓨터 프로그래밍 학습을 위해 정규 수업으로 진행되는 자바 프로그래밍 수업과 병행하여 심화·보충형 수준별 스터디를 운영하고, 사후 평가를 통해 프로그래밍 학습에서의 심화·보충형 수준별 학습 효과를 증명하였다. 상·중·하위 그룹으로 구성된 세 개의 스터디 팀과 스터디 미참여 그룹 학습자들을 대상으로하여 스터디 전후의 프로그래밍 학습 향상도 검사, 스터디 후 각 수준의 학습자들의 프로그래밍 영역별 이해도 차이를 분석하고 나아가 각 학습 수준별로 심화·보충형 수준별 스터디에 효과적인 학습 영역이 무엇인지를 도출하였다. 그 결과, 심화·보충형 수준별 학습에 효과적인 프로그래밍 영역으로는 개념의 심도 있는 이해를 통해 이를 적용한 프로그래밍 능력을 키우는 부분과 여러 학습 영역들의 복합적인 연계지식을 통해 프로그래밍을 하는 영역이 이에 해당되었다. 또한 심화·보충형 수준별 스터디에 참여한 학습자들이 전체 프로그래밍 학습 영역에서 학습 수준이 향상되었음은 물론이고, 하위 수준의 학습자들이 상대적으로 가장 많은 학습 향상도를 이루었다. 스터디에 참여한 하위 학습자들은 스터디 중상위 학습자들보다 개념을 활용한 프로그래밍 응용 능력에서 상대적으로 뒤쳐졌으나 스터디에 참여하지 않은 학습자들과 비교해서는 대부분의 학습 영역에서 우수한 학습 결과를 보였다.

#### Abstract

We've run leveled intense and supplementary study groups connected with JAVA regular classes for more effective learning of computer programming. We performed the test for evaluating learning understanding degrees between before and after study to analyze learning effect for three leveled study groups and a non study group, and we also analyzed the differences of learning understanding degrees among 4 learner groups by computer programming areas. As a result, we can reach the result what programming areas are effective to be operated by intense and supplementary study groups. Those areas are the area to improve the ability of programming applying concepts through in-depth understanding of concepts, and the area of programming through mixed knowledges related each other. The study group for low-leveled learners has most improved of four groups. Low-leveled group was behind the other study groups on programming application ability, but the group was more excellent than non study group in most programming areas.

▶ Keyword : 수준별 학습(levelled learning), 심화·보충형 수준별 학습(levelled intense and supplementary learning), 컴퓨터 프로그래밍 학습(computer programming learning)

· 제1저자 : 안유정    교신저자 : 김경아

· 투고일 : 2010. 08. 07, 심사일 : 2010. 08. 20, 게재확정일 : 2010. 08. 27.

\* 명지전문대학 컴퓨터정보과 교수    \*\* 명지전문대학 컴퓨터정보과 교수

## I. 서론

다양한 수준의 학습자들에게 적합한 효과적인 컴퓨터 프로그래밍 수업 유형을 찾고자 하는 연구는 꾸준히 이루어져왔다 [1][2][3][4][5][6][7][8][9]. 특히 심화·보충형 수준별 학습에 의해 학습 효과를 높이고자 하는 노력은 컴퓨터분야 뿐만 아니라 여러 분야의 학습에서 계속되어 왔으며 여러 수업 사례들이 연구되고 발표되었다[1][3][4][6][11]. 그러나 동일한 내용의 학습을 수준별 학습으로 진행하였을 때 상·중·하위 수준의 학습자들이 프로그래밍의 세세한 학습 영역 중 어느 영역에서 얼마나 학습 향상 효과를 나타내며, 서로 다른 수준의 학습자들 간에 유의미한 학습 차이를 보이는 영역은 무엇인지에 대해서는 거의 연구되어진 바가 없다.

따라서 본 연구자들은 소속 대학에서 강의하고 있는 객체지향언어 수업에 참여하고 있는 학습자들을 대상으로 수업과는 별도의 심화·보충형 수준별 스텐디를 진행하고 스텐디 전후에 각 수준별 그룹의 학습자들의 학습 능력을 평가하여 학습자들의 학습 능력이 얼마나 향상되었는가를 분석하였다. 또한 나아가 수준이 다른 그룹들 간의 상대적인 학습 평가 결과의 비교를 통해 수준별 스텐디 후에도 각 수준의 그룹들 간에 유의미한 차이를 보이는 프로그래밍 영역이 무엇인지도 분석하였다. 이러한 결과는 프로그래밍 학습을 심화·보충형 수준별 학습으로 진행하는데 있어 효과적인 수업 방법의 모델링에 좋은 근거가 될 수 있다. 또한 본 연구에서는 스텐디를 거치지 않고 정규 수업만에서 학습한 프로그래밍 영역에 대해서도 동일한 학습자들을 대상으로 평가해 봄으로써 심화·보충형 수준별 학습의 효과를 상대적으로 확인해 보기도 하였다. 본 연구의 궁극적인 목표는 이러한 결과를 토대로 효과적인 컴퓨터 프로그래밍 학습을 위한 수준별 수업 모형을 모색하는데 있다.

II장에서는 수준별 교육과정에 대한 개요 및 유형들을 소개하고 학습자들의 학습 수준 구분 시점 또한 학습 효과에 영향을 미칠 수 있다는 기존 연구를 소개한다. III장에서는 본 연구의 연구 목적 및 연구 과정 그리고 실험 대상과 방법을 소개하고, IV장에서는 본 연구에서 진행한 심화·보충형 수준별 학습의 학습 효과를 증명하기 위해 스텐디 참여 그룹과 미참여 그룹간의 학습 효과 비교 실험을 실시하고 나아가 학습 수준 상·중·하의 스텐디 그룹과 미참여 그룹간의 학습 효과를 비교 실험한 결과를 소개한다. 또한 이 네 개의 학습 그룹들을 대상으로 상대적으로 학습 효과가 두드러진 프로그래밍 영역에 대해서도 분석함으로써 향후 컴퓨터 프로그래밍 분야에 대한 효과적인 심화·보충형 수준별 수업 모형을 설계하는데 유용한 근거 자료를 제시한다. 마지막으로 V장에서는 이번 연구

의 결과와 이를 토대로 한 향후 연구 방향에 대해 제안한다.

## II. 관련연구

### 2.1 수준별 교육과정

수준별 교육과정이란 교과별로 수준을 달리하여 학생 개인의 학습 능력에 맞춰 학습할 수 있도록 한 개별화된 교수·학습 형태의 일종이다[11]. 수준별 교육 과정에서는 어떤 표준이나 정도에 의해 교육과정의 등급을 나누게 되는데, 교육개혁 위원회에서는 그 기준으로 '학습자의 능력과 적성'을 제시하였다. 그러므로 수준별 교육과정은 학생의 능력 수준에 따라 교육의 내용이나 교육 방법을 달리하는 개별화된 교육과정이나 다양화된 교육과정의 한 구체적인 전형이라 할 수 있다[6].

### 2.2 수준별 교육과정의 유형

수준별 교육과정의 유형은 학생의 학습 능력, 학습 집단에 따라 다양하게 분류될 수 있으나 크게는 세 가지 유형인 단계형, 심화·보충형, 과목 선택형으로 나눌 수 있다[6][9][10].

먼저 단계형 수준별 교육과정은 학습 내용의 위계가 분명하고 학습 집단 구성원의 능력 차이가 심하게 작용하는 교과에서 난이도나 논리적 위계를 기준으로 만든 교육과정으로, 학생의 학습 속도에 맞게 단계별로 세분화해 운영하는 교육과정이다.

심화·보충형 수준별 교육과정은 교과의 내용이 다양한 종류의 과목이나 영역으로 구성되어 있고, 학습 집단 구성원의 능력의 개인차가 그다지 심각하게 작용하지 않는 교과에서 학습 내용의 범위와 수준을 달리하여 만든 교과과정으로서, 학생의 능력 수준에 따라 기본 학습 내용을 보충 또는 심화할 수 있도록 편성·운영하는 교육과정이다.

마지막으로 과목 선택형 수준별 교육과정은 학생들의 능력, 적성, 필요, 관심의 차이를 반영하는 다양한 과목을 개설하여 자신의 진로와 능력 수준에 알맞은 과목을 학생 스스로가 선택할 수 있도록 편성·운영하는 교육과정이다. 개설과목은 학습내용의 난이도에 따라 낮은 수준에서 높은 수준에 이르기까지 다양한 과목을 개설할 수 있으며 또한 교과별 학습 내용을 여러 영역으로 나누어 독립된 과목으로 개설할 수 있고 높은 수준의 과목은 학습 내용을 탐구 학습 내용 중심으로 구성하며 단위 수도 다른 교과보다 늘릴 수 있다.

### 2.3 학습수준 구분 시점에 따른 학습효과

수준별 학습에 대한 기존 연구 중의 하나로 학습자들을 학습 수준별로 나누는 시점에 따라 학업성취도가 어떻게 달라지

는지를 실험하여 분석한 연구가 있다(6). 본 학습이 이루어지기 전에 진단평가에 의해 학습 수준을 구분하는 경우와 어느 정도 학습을 진행한 후 형성평가에 의해 학습 수준을 구분하는 경우의 학업 성취도를 비교해본 결과 형성평가형 학습 모형에 참여한 학습자들의 학업 성취도가 더 높았다.

### III. 연구내용 및 방법

#### 3.1 연구 개요

본 연구에서는 정규 수업과 동일한 내용의 학습을 전공스터디를 통해 심화·보충형 수준별 학습으로 함께 진행하였을 때 상·중·하위 수준의 스터디 학습자들과 스터디에 참여하지 않은 학습자들이 프로그래밍의 세분화된 학습 영역 중 어느 영역에서 얼마나 학습 향상 효과를 나타내며, 서로 다른 수준의 학습자들 간에 유의미한 학습 차이를 보이는 영역은 무엇인지에 대해 알아보려고 한다.

이러한 연구 목적을 위해 스터디 전 평가를 통해 네 그룹 학습자들의 학습 수준을 평가 및 비교하고 스터디 종료 후에도 네 그룹 학습자들을 대상으로 정규 수업에서 다루었던 객체지향 프로그래밍의 학습 인지도에 대한 평가를 학습 영역별로 실시하였다.

#### 3.2 연구 방법

##### 3.2.1 연구 대상자

연구의 대상 학습자로는 본 연구자가 담당하고 있는 M대학 컴퓨터정보과 2학년 전공과목인 객체지향 언어 과목을 수강하는 학생들 55명을 대상으로 하였다. 학습자 구성은 표1과 같다.

표 1. 연구 대상 학습자  
Table 1. Learners for Object of This Research

대상 학습자	전공스터디 참여 여부 및 팀 유형	성별		계(명)	비율(%)
		남	여		
전공스터디 참여 그룹	1팀	14	2	16	29.09
	2팀	3	7	10	18.18
	3팀	1	4	5	9.09
전공스터디 미참여 그룹		17	7	24	43.64
계		35	20	55	100

##### 3.2.2 연구 대상 학습 영역

본 연구를 적용한 객체지향언어 과목은 자바를 통해 객체지향 프로그래밍을 학습하는 과목이다. 이 과목의 범주인 프로그래밍 언어 교육은 동일한 수준에서 출발한 학습자들 간의 단계별 학습의 차이가 점차 증대되어 학습자의 수준별 학습이

필요한 대표적인 과목이다. 단순한 지식 습득이 아니라 오랜 시간에 걸쳐 지식이 축적되고 훈련되므로 시간이 갈수록 학습 수준의 격차가 커지게 된다. 또한 학습자들의 적성이나 선행 학습의 정도에 따라서도 학습자들 간의 수준차가 현격하므로 한 강의실에서 동일한 시간에 동일한 방법으로 가르치는 고전적인 수업 방식으로는 이러한 학습자들의 학습 욕구를 만족시킬 수 없다(3)(5). 더구나 이번 연구 대상이 되는 과목인 객체지향언어는 기존의 절차적 프로그래밍에서 다루었던 변수의 개념이나 연산자, 제어문 및 함수의 개념들은 그대로 포함하면서 객체지향프로그래밍이라는 새로운 영역을 학습하는 과목이다. 따라서 절차적 프로그래밍에 익숙한 학습자들이 적응하는데 시간이 걸릴 수 있으므로 수업의 학습 효과를 증대시키기 위해서 본 연구에서는 수업이외에 전공스터디를 통해 학습자들에게 수준별 학습을 운영하였다.

##### 3.2.3 심화·보충형 수준별 학습의 적용

본 연구자는 세 가지 유형의 수준별 교육과정 중에서 심화·보충형 수준별 학습을 선택하여 전공스터디 운영에 적용하였다. 대부분의 학습자들이 이 과목에서 객체지향프로그래밍을 처음 학습하게 되는데, 과거 이 과목의 수업을 진행한 경험을 비추어 볼 때 한 번의 개념 설명만으로는 완전한 개념 이해가 어려웠다. 반복적인 상세한 설명과 개념을 활용한 다양한 프로그래밍 실습이 수차례 반복되고 나서야 중위 그룹 이상의 학습자들이 이해하는 모습을 보았다. 그러나 정규 수업만으로는 하위 그룹 학생들에게는 개념 이해도 어려웠으며 중급 이상의 학습자들도 다양한 응용력을 기르기는 한계가 있었다. 따라서 학습에 대한 이해가 빠른 상위 그룹 학습자들에게는 수업 시간에 학습한 영역에 대해 심화 학습을 진행하고 이해도가 상대적으로 낮은 하위 그룹으로 갈수록 동일한 학습 영역에 대해 유사하거나 보다 쉬운 수준으로 보충학습을 실시하는 학습이 필요하였다.

따라서 본 연구자는 수업에서 학습했던 동일한 내용에 대해 학습자들의 수준에 따라 학습 이해도와 만족도를 높일 수 있는 쪽으로 스터디의 방향을 정하고 이에 적합한 심화·보충형 수준별 학습유형을 적용하게 되었다.

##### 3.2.4 연구 조사 도구

조사 도구는 전공스터디 참여자용과 미 참여자용으로 구분된 두 종류의 학생 설문지와 학습 전후의 학습인지도 평가를 위해 스터디 전후에 각각 사용된 두 종류의 평가지이다. 학생 대상 설문 내용은 대상 학생들의 기본 정보, 수강 동기에 대한 문항들과 운영 내용 및 방법, 구성된 수 등에 대한 평가를 묻는 문항, 자신의 참여 적극성 평가, 스터디 및 수업에 대한

만족도 평가 등의 16문항으로 구성되었다. 학생 대상 평가 문항은 객체지향언어 수업에서 다룬 내용을 근거로 기초 학습에서부터 고급 프로그래밍 능력까지 평가할 수 있도록 구성하였다.

문항들은 연구자가 직접 제작한 후 프로그래밍 교육 경력이 15년 이상인 전문가 5인의 검토를 받아 2차에 걸친 수정 보완 후 적용하였다.

### 3.3 심화·보충형 수준별 스터디의 운영

#### 3.3.1 정규 수업

본 연구의 대상이 되는 객체지향언어 수업은 M대학 컴퓨터 정보과 2학년의 1년 과정의 자바 프로그래밍 수업 중에서 첫 번째 학기 수업이며 전체 정규 교과과정 중 객체지향프로그래밍을 학습하는 첫 수업이다. 객체지향언어의 한 클래스의 학생 수는 20~35명이며 2개의 클래스를 대상으로 하여 총 55명의 학생이 연구 대상이 되었다. 객체지향언어 수업의 내용은 자바 기본 문법, 객체지향의 기본 개념 및 모델링 방법, 메소드 오버로딩, 생성자 오버로딩, 클래스 상속, 메소드 오버라이딩, 추상 메소드를 이용한 다형성, 인터페이스를 통한 다중상속, 자바의 기본 패키지 이해 및 사용, 사용자 패키지 작성으로 구성되며 이론과 실습을 병행하여 주당 4시간씩 진행되었다.

#### 3.3.2 스터디 구성 방법

객체지향언어 정규 수업에 참여하는 학생들 중에 참여를 희망하는 학생들을 대상으로 스터디를 운영하였다. 학기 중 스터디 운영 시작 시기는 학기초로부터 한 달 이상 경과하여 정규 수업에서 자바의 기본 문법 학습이 끝나고 객체지향 개념을 시작하기 직전에 시작하였다.

본 연구자는 기존에 전공스터디의 팀 구성 방법에 많은 관심을 가지고 있었으므로 수년간 다양한 팀 구성 방법을 시도해 보았고 방법에 따른 학습 효과도 비교해보았다(3)(13). 연구 결과, 테스트 결과에 의한 강제적인 팀 구성보다는 테스트 결과와 학습자의 의사를 함께 반영한 경우가 보다 좋은 학습 효과를 거두었다(3). 따라서 이번 연구에서도 팀 구성 시 테스트 결과와 본인의 의사를 함께 반영하여 팀을 구성하였다.

평가 결과 상위 학습 수준의 팀을 1팀, 중간 수준의 학습 그룹을 2팀, 기초 수준의 학습 팀을 3팀으로 분류하여 총 세 개의 스터디 팀을 구성하였으며 본인의 의사를 함께 반영하여 표 2와 같이 분류하였다. 표 2에서 본인의사에 의한 상향 편입 비율이란 본인 의사에 의해 테스트 결과로 나온 자신의 수준보다 높은 학습 수준 팀에 편입한 경우이고 하향 편입 비율이란 반대로 자신의 수준보다 낮은 학습 팀에 편입된 경우이다. 단 상향 또는 하향 편입 시 한 단계의 이동만 허용하였다.

표 2. 스터디 팀 분류 기준

Table 2. Classification Standard for Study Groups

팀명	테스트 결과	인원 (명)	본인 의사에 의한 상향 편입 비율(%)	본인 의사에 의한 하향 편입 비율(%)
1팀	정답률 70%이상	16	6.25	0
2팀	정답률 50~70%	10	20	30
3팀	정답률 50%미만	5	0	20

#### 3.3.3 스터디 팀 분류를 위한 평가 시점

관련연구(6)을 통해 팀 분류를 위한 평가 시점으로서 진단 평가형보다는 형성평가형의 학습성취도가 높았으므로 팀 분류 시점은 자바의 기본 문법에 대한 학습이 어느 정도 진행된 뒤 그동안 학습한 내용을 가지고 형성평가를 통해 이루어졌다.

#### 3.3.4 스터디 팀 분류를 위한 평가 내용

평가 내용은 대부분의 학습자들이 평가 시점까지 객체지향 프로그래밍 학습에 대한 선수 지식이 없었으므로 자바 문법을 기반으로 하되 이전까지 학습한 절차적 프로그래밍 방식으로만 평가하였다. 구체적인 평가 내용은 프로그램 실행 환경 및 프로그래밍 절차, 기본 자료형 및 연산자, 변수 선언, 조건식 작성, 조건문을 사용한 프로그래밍, 배열의 선언 및 사용, 반복문과 조건문을 복합적으로 사용한 프로그래밍, 반복문과 조건문 및 배열을 복합적으로 사용한 프로그래밍으로 구성되었다.

#### 3.3.5 스터디 전 네 그룹의 학습 수준 비교

스터디 전, 학습자들의 학습 수준 차이를 검증하기 위해 스터디 그룹 세 팀과 스터디 미참여 그룹(4팀)에 대해 동일한 문항으로 형성평가를 실시하고 네 집단의 학습 수준차이를 비교 분석하였다. 네 팀 학습자들의 평균 점수를 가지고 Scheffe 테스트를 실시하여 95% 신뢰구간으로 검증한 결과가 표 3과 같다. 유의미하게 뚜렷한 학습 수준 차이가 나는 그룹은 1팀과 3팀 그리고 1팀과 4팀이다. 1·2팀 간, 2·3팀 간, 2·4팀 간 그리고 3·4팀 간에도 학습 수준의 차이는 있었으나 통계적으로 유의미한 차이를 보이지는 않았다. 단 스터디 전 1팀과 2팀은 스터디를 하지 않는 4팀보다 학습 수준이 높았으나 3팀은 4팀보다 수준이 낮았다.

표 3. 스터디 전 네 집단의 학습수준 차이 검증

Table 3. Verification about Differences of Four Groups' Learning Levels before Studying

집단명 (i)	비교집단명 (j)	평균차 (i-j)	표준오차	유의확률
1	2	14.8750	7.3259	0.261
1	3	30.8750	9.3111	0.018
1	4	22.15277	5.8654	0.005
2	3	16.0000	9.9540	0.467
2	4	7.27777	6.8402	0.770
3	4	-8.7222	8.9340	0.812

### 3.3.6 스터디 운영 방법

전공스터디의 학습 내용은 정규 수업에서 학습한 내용과 동일한 영역을 학습하였다. 운영 방법은 팀별 학습자들의 수준에 따라서 상위 학습자 그룹은 보다 심도 있는 개념 이해와 고난이도의 다양한 프로그래밍 실습을 하였고 하위 그룹으로 갈수록 개념에 관한 추가적인 설명과 보다 쉬운 프로그래밍 실습을 통해 응용력을 키울 수 있도록 구성하였다. 스터디 기본 운영 형태는 한 학습 영역에 대해 담당교수의 개념 설명, 질의응답과 문제 풀이 그리고 프로그래밍 과제 부여 및 정답 확인으로 구성되었다. 기본 운영 형태는 세 팀 모두 유사하나 학습 수준별로 비중을 두는 부분이 조금씩 달랐다.

## IV. 연구 결과

### 4.1 스터디 종료 후 학습 결과 평가 방법

#### 4.1.1 학습효과 평가 영역

스터디가 종료된 시점에서 수준별 전공스터디의 학습 효과를 조사하기 위해서 세 팀의 스터디 그룹과 스터디에 참여하지 않은 그룹의 학습자들에게 동일한 문항으로 구성된 평가지를 통해 평가를 실시하였다. 평가 영역은 총 24영역으로 자바의 기본 문법에서부터 개념이해 및 프로그래밍 응용 능력을 평가하는 문제들로 구성되었으며 다음 표 4와 같다.

표 4. 학습효과 분석을 위한 학습 평가 영역  
Table 4. Learning Areas Evaluated for Analyzing Learning Effect

문제	학습유형	평가 내용
Q12-3	단순지식	변수 선언문 작성
Q12-4		배열의 선언 및 생성 사용
Q12-6		조건식 작성
Q12-11	프로그래밍 응용	반복문을 사용하여 프로그램 작성
Q12-13		반복문과 조건문의 중첩 사용
Q12-15		중첩 반복문을 사용한 프로그램 작성
Q12-32	개념을 적용한 프로그래밍 응용	메소드 오버로딩 프로그래밍
Q12-38		메소드 오버라이딩 프로그래밍
Q12-43		추상 메소드를 통한 다형성의 구현
Q12-46	개념이해	인터페이스를 통한 다중상속 구현
Q12-51		기본 패키지를 사용한 프로그래밍
Q12-52		사용자 정의 패키지 작성
Q12-21	개념이해	객체지향 문제에서 객체 추출
Q12-22		데이터와 메소드를 통한 객체 정의
Q12-23		객체 정의를 통한 클래스 정의
Q12-24		클래스의 멤버변수의 의미와 역할
Q12-25		클래스의 생성자의 의미와 역할

문제	학습유형	평가 내용
Q12-26		클래스의 메소드의 의미와 역할
Q12-27		객체의 선언, 생성 및 사용
Q12-31		메소드 오버로딩 개념 이해
Q12-36		클래스 상속을 통한 클래스 정의
Q12-37		메소드 오버라이딩 개념이해
Q12-40		하위클래스 생성자에서 상위클래스 생성자 호출
Q12-42		다형성의 의미

#### 4.1.2 평가지 채점 기준

대상 학생들에게 실시한 평가지 채점 기준은 각 문제마다 5점을 만점으로 하였으며, 문제에서 제시한 의미를 완전히 파악하고 프로그래밍 형식도 완전할 경우에는 5점, 의미 파악은 완전하나 형식에서 미미한 오류가 있는 경우에는 4점, 의미만 파악한 경우 3점, 형식과 의미를 부분적으로만 파악한 경우 2점, 문제에 대한 의미와 형식을 전혀 모르는 경우 1점으로 점수를 부여하였다.

### 4.2 스터디 참여여부에 따른 두 집단의 학습 결과 비교

#### 4.2.1 스터디 후 두 학습 집단의 학습결과 비교

전공스터디를 통한 심화·보충형 수준별 학습의 효과를 알아보기 위해 스터디 참여 학생 집단과 미참여 학생 집단의 평가지 결과 점수를 분석하였다. 두 집단 간의 점수에 대한 차이를 검증하기 위해 독립표본 T검정을 실시한 결과는 표 5와 같다.

5개의 평가 영역을 제외한 19개의 영역에서 스터디 참여 학생의 평균점수가 미참여 학생의 평균점수에 비해 유의미하게 높은 것으로 나타났다. 나머지 5개의 영역도 역시 참여 학생들의 점수가 높았으나 통계적으로 유의미하게 높은 점수는 아니었다. 결과적으로 스터디를 참여한 학생들의 학습결과가 미참여 학생들의 학습결과보다 월등히 좋다고 해석할 수 있다.

스터디 참여 집단과 미참여 집단 사이에 유의미한 차이를 보이지 않은 문항들은 변수 선언문이나 배열의 선언 및 사용과 같은 기초적인 선언문이나 객체지향 문제에서 객체의 추출 및 정의 같이 단순 지식이나 단순한 개념 이해의 영역에 해당된다고 볼 수 있다. 이와 같은 영역의 학습은 수준별 학습을 통하지 않고 단지 정규 수업만으로도 그 학습 효과가 유사하다는 것을 알 수 있다.

반면에 스터디 참여 집단의 학습결과가 미참여 집단보다 유의미하게 우수하다는 결과를 보인 프로그래밍 영역은 개념의 심도 있는 이해를 필요로 하는 학습 영역이나 개념을 활용한 응용 프로그래밍 작성 분야가 이에 해당된다.

표 5. 스테디 후 두 집단의 학습결과 차이 비교  
Table 5. Comparison of Differences of Learning Results between Two Groups after Studying

문제	전공스테디참여 여부	N	평균	표준편차	평균의 표준오차	t	유의확률 (양쪽)
Q12-3	참여	28	4.68	1.188	.225	.643	.523
	미참여	21	4.43	1.535	.335		
Q12-4	참여	28	4.14	1.508	.285	1.920	.061
	미참여	21	3.24	1.786	.390		
Q12-6	참여	28	4.25	1.236	.234	2.042	.047
	미참여	21	3.38	1.746	.381		
Q12-11	참여	28	4.39	1.197	.226	2.406	.020
	미참여	21	3.33	1.880	.410		
Q12-13	참여	28	4.07	1.654	.313	1.429	.160
	미참여	21	3.33	1.958	.427		
Q12-15	참여	28	3.86	1.433	.271	2.658	.011
	미참여	21	2.62	1.830	.399		
Q12-21	참여	28	4.00	1.333	.252	1.617	.113
	미참여	21	3.14	2.351	.513		
Q12-22	참여	28	3.39	2.200	.416	1.647	.106
	미참여	21	2.29	2.493	.544		
Q12-23	참여	28	3.00	1.944	.367	3.246	.002
	미참여	21	1.33	1.528	.333		
Q12-24	참여	28	3.18	2.262	.427	3.232	.002
	미참여	21	1.19	1.940	.423		
Q12-25	참여	28	3.00	2.211	.418	3.428	.001
	미참여	21	1.05	1.596	.348		
Q12-26	참여	28	3.71	2.052	.388	4.078	.000
	미참여	21	1.38	1.884	.411		
Q12-27	참여	28	3.36	1.929	.364	3.581	.001
	미참여	21	1.29	2.101	.458		
Q12-31	참여	28	3.54	2.202	.416	3.694	.001
	미참여	21	1.29	1.978	.432		
Q12-32	참여	28	3.54	2.202	.416	3.694	.001
	미참여	21	1.29	1.978	.432		
Q12-36	참여	28	3.43	1.971	.372	3.890	.000
	미참여	21	1.29	1.821	.397		
Q12-37	참여	28	3.61	2.132	.403	2.393	.021
	미참여	21	2.05	2.418	.528		
Q12-38	참여	28	3.61	2.132	.403	2.393	.021
	미참여	21	2.05	2.418	.528		
Q12-40	참여	28	2.82	2.109	.399	2.559	.014
	미참여	21	1.33	1.880	.410		
Q12-42	참여	28	3.96	1.972	.373	3.939	.000
	미참여	21	1.57	2.271	.496		
Q12-43	참여	28	3.50	2.152	.407	2.898	.006
	미참여	21	1.67	2.244	.490		
Q12-46	참여	28	3.89	1.931	.365	3.488	.001
	미참여	21	1.76	2.343	.511		
Q12-51	참여	28	2.18	2.019	.382	2.513	.015
	미참여	21	.81	1.692	.369		
Q12-52	참여	28	2.25	2.171	.410	2.049	.046
	미참여	21	1.05	1.830	.399		

4.2.2 스테디 참여 전후 두 집단의 학습 향상도 비교

전공스테디 참여집단과 미참여 집단을 대상으로 스테디 전과 후에 각각 실시한 테스트를 통해 학습 결과를 분석한 결과는 그림 1과 같다.

사전과 사후 평가의 문제 난이도가 다르다하더라도 동일한 문제로 두 집단을 평가하였으므로 상대적인 비교가 가능하다. 그림 1에서 스테디 참여 집단은 사전보다 사후 평가에서 평균 점수가 약간 높아진 반면 스테디 미 참여 집단은 오히려 사후에 실시한 평가에서 평균 점수가 훨씬 낮아졌다. 따라서 사후 평가 문제의 난이도가 높다하더라도 전공스테디에 참여한 학습자들의 상대적인 학습 수준 향상을 확인할 수 있다.

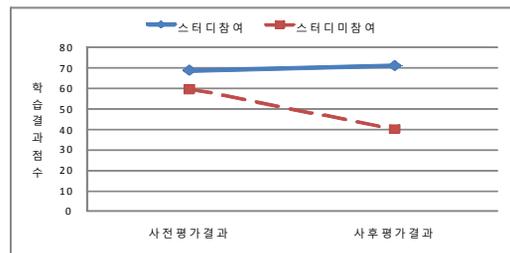


그림 1. 스테디 참여 전후 두 집단의 학습향상도 비교  
Fig. 1. Comparison of Degrees of Learning Improvement before and after Studying

4.3 네 학습 집단의 학습 결과 비교

4.3.1 네 학습 집단 간의 문항별 학습 결과 비교

전공스테디에 참여한 1, 2, 3팀의 학습자들과 미참여 그룹인 4팀의 학습자들에 대해 Scheffe 테스트를 실시하여 95% 신뢰 구간으로 네 집단의 학습 결과를 비교한 결과, 집단 간 유의미한 학습 능력 차이를 보인 경우만을 따로 정리한 것이 표6이다.

표 6. 스테디 후 평균 점수를 통한 네 그룹들의 학습수준 차이 비교  
Table 6. Comparison about Differences of Four Groups' Learning Levels by average points after Studying

집단명 (i)	비교집단 (j)	평균차 (i-j)	표준오차	유의확률
1	2	15.4615	10.5063	0.544
1	3	24.1282	13.1444	0.350
1	4	40.8424	8.8149	0.000
2	3	8.6666	13.6811	0.939
2	4	25.3809	9.5969	0.087
3	4	16.7142	12.4294	0.617

표 6에서 보면 스테디 팀들 중에 최상위 수준의 학습자들로 구성된 1팀과 스테디를 하지 않은 4팀 간에는 총 24개의 테스트 영역 중에 14개의 학습 영역에서 유의미한 차이를 보임으로서 1집단이 더 우수함을 보였다. 그러나 스테디 그룹들인 1, 2, 3 팀들 간에 유의미한 학습 차이를 보인 영역으로는 1, 2팀 간에는 두 영역, 1, 3팀 간에 세 영역에서 유의미한 차이를 보였으며 2, 3팀 간에는 유의미한 차이를 보인 영역이 없었다. 네 집단의 보다 정확한 학습 결과의 차이를 검증하기

위해 사후 테스트 결과로 나온 네 집단 학습자들의 평균 점수를 가지고 Scheffe 테스트를 실시하여 95% 신뢰구간으로 검증한 결과가 표 7과 같다.

표7에서 보면 1팀과 4팀 간에만 대부분 유의미한 학습 결과의 차이를 보이고 있다. 이것은 표3에서 보였던 스테디를 시작하기 전 1·3팀 간 그리고 1·4팀 간의 학습 수준 차이를 고려해 볼 때, 의미 있는 결과라고 할 수 있다. 즉, 심화·보충형 수준별 학습을 통해 1·3팀 간에는 학습자들의 프로그래밍 학습 수준 차이가 줄어들었다고 볼 수 있다.

스테디 1팀과 스테디 미참여자 4팀 간에 유의미한 학습 결과 차이를 보인 학습 영역들을 큰 범주로 구분해보면 다음과 같은 다섯 가지 범주에서 학습 능력의 차이를 보였다. 첫째는 객체지향개념의 심도 있는 이해를 통해서만이 가능한 객체지향 모델링, 즉 설계한 객체에 대해 클래스를 정의하여 클래스의 멤버변수와 생성자, 메소드들을 정의하고 정의된 클래스를 통해 객체를 선언, 생성 및 사용하는 영역이다. 둘째로 클래스 상속에 의한 클래스 정의, 셋째로 추상메소드를 통한 다형성의 개념 이해 및 프로그래밍 구현, 넷째로 인터페이스를 사용한 다중상속의 구현 그리고 다섯 번째로는 패키지 사용 및 사용자 패키지 작성이다. 따라서 1팀과 4팀 간에 유의미한 차이를 보인 영역들을 종합해보면 심도 있는 객체지향 프로그래밍 개념 이해와 이를 바탕으로 개념을 문제 해결에 적용하여 프로그래밍을 하는 영역들이다. 또한 인터페이스를 통한 다중상속의 구현은 클래스 상속의 개념과 연계되어 있고, 그리고 추상클래스와도 유사성이 있으므로 여러 개념이 연계되어 있는 영역들에서도 학습 능력의 차이가 났다고 할 수 있다.

또한 스테디 하위 그룹인 3팀의 점수와 미참여 그룹인 4팀의 문항별 점수 평균을 비교했을때 유의미한 차이를 보인 문항은 없었으나 Q12-38과 Q12-43을 제외한 모든 문항에서 3팀의 점수가 더 높았다. 이 두 예외적인 문항은 메소드 오버라이딩의 구현과 추상메소드를 통한 다형성의 구현 부분으로 개념을 적용한 프로그래밍 영역이라는 공통점이 있다.

표 7. 스테디 후 네 집단 간의 영역별 학습수준 차이 검증  
Table 7. Verification about Differences of Learning Levels by Areas among Four Groups after Studying

문제	우수집단 (i)	유의미한 차이가 나는 비교집단 (j)	평균차 (i-j)	표준오차	유의확률
Q12-4	1	2	2.100	0.627	0.017
	1	4	1.762	0.526	0.017
Q12-11	1	4	1.590	0.533	0.042
Q12-15	1	4	1.919	0.550	0.012
Q12-23	1	4	2.513	0.590	0.001
Q12-24	1	4	2.348	0.738	0.026

문제	우수집단 (i)	유의미한 차이가 나는 비교집단 (j)	평균차 (i-j)	표준오차	유의확률
Q12-25	1	4	2.645	0.679	0.004
	1	4	2.311	0.715	0.023
Q12-26	2	4	2.319	0.778	0.042
	1	4	2.484	0.712	0.012
Q12-36	1	4	2.253	0.688	0.021
Q12-42	1	4	3.044	0.735	0.002
	1	3	3.538	1.053	0.017
Q12-43	1	4	2.872	0.706	0.003
	1	4	2.315	0.754	0.034
Q12-51	1	2	2.038	0.672	0.037
	1	3	3.538	0.841	0.002
	1	4	2.729	0.564	0.000
Q12-52	1	3	3.000	0.996	0.039
	1	4	1.952	0.668	0.048

이번에는 스테디 그룹 내의 팀들 간에 유의미한 차이를 보인 학습 영역을 분석해보면 1팀과 2팀간에는 자바에서 객체로 취급하고 있는 배열의 선언 및 생성 그리고 사용에 관한 학습 영역에서 유의미한 차이로 1팀이 우수했으며, 1팀과 3팀간에는 추상 메소드를 통한 다형성의 구현 영역에서 역시 1팀이 유의미하게 우수하였다. 이 경우 주목할 만한 것은 스테디 그룹 중 하위 수준팀인 3팀의 학생들이 다형성의 개념 이해 부분에 대해서는 1팀과 유의미한 차이가 없었으나 다형성의 개념을 적용한 프로그래밍 부분에서는 학습 차이가 난다는 점이다. 3팀은 이 영역에 대해 1, 2, 4팀에 모두 뒤졌는데 1팀과는 유의미한 학습 차이가 났고 2, 4팀과는 근소한 차이가 났다. 또한 2, 3팀간에는 대부분 영역에서 유사한 점수 결과를 보였는데 단 개념을 적용한 프로그래밍 응용 영역에서는 한 문제만 제외하고 근소하나마 2팀이 모두 우수하였다. 이를 통해 하위 학습자들의 개념을 활용한 프로그래밍 능력은 심화·보충형 수준별 학습을 통해서 중급이상의 학습자들에 근접해가기는 하지만 학습 수준 차이를 없애기는 어려워 보였다.

마지막으로 스테디 그룹 내의 팀들 간에 유의미한 학습 차이가 나는 영역은 스테디 미참여 그룹과 마찬가지로 패키지 사용 및 사용자 패키지 작성 영역이며 이 경우 1, 2, 3팀들 간에 모두 유의미한 차이를 보였다. 그런데 평가지의 대부분의 문제들은 수업과 스테디 각 팀에서 모두 다루었던 학습 영역이나 패키지 영역은 수업에서는 다루었으나 스테디에서는 다루지 않았다. 그 결과 눈여겨 볼만한 점은 스테디에서 다루었던 학습 영역들과는 달리, 다루지 않았던 학습 영역에 대해서는 스테디 팀들인 1, 2, 3팀들 간에 모두 유의미한 학습 차이가 났다는 점이다. 이 결과로 다양한 수준의 학습자들이 심화·보충형 수준별 학습을 한 학습 영역에 대해서는 대부분 유의미한 차이를 보이지 않았으나 스테디를 하지 않은 영역에 대해서는 유의미한 학습 수준의 차이를 보인다는 주장을 뒷받침하게 되었다.

## IV. 결 론

결론적으로 여러 수준의 학습자들에게 심화·보충형 수준별 학습으로 진행되었던 전공스터디가 효과적이었던 프로그래밍 학습 영역은 개념의 심도 있는 이해를 통해 이를 적용한 프로그래밍 능력을 키우는 부분과 여러 학습 영역들의 복합적인 연계지식을 통해 프로그래밍을 하는 영역이었다. 따라서 전공스터디의 효과를 보다 높이기 위해서는 단순지식이나 기본적인 개념 이해는 정규 수업에서 실시하고 과외의 심화·보충형 수준별 학습을 통해서 개념을 적용한 프로그래밍 학습과 여러 개념을 연계한 지식들을 학습자들의 이해 수준에 따라 점차적으로 학습하도록 계획하여 프로그래밍 능력을 키울 수 있도록 운영하는 것이 효과적이라고 할 수 있다.

또한 심화·보충형 수준별 학습을 통해 스터디 참여 학습자들의 전체 프로그래밍 학습 영역에서 학습 수준이 향상되었음은 물론이고, 하위 수준의 학습자들이 상대적으로 가장 많은 학습 향상도를 이루었다. 스터디 하위 학습자들은 스터디 중상위 학습자들보다 개념을 활용한 프로그래밍 응용 능력에서 상대적으로 뒤쳐졌으나 스터디에 참여하지 않은 학습자들과 비교해서는 대부분의 학습 영역에서 우수한 학습 결과를 보였다.

향후 연구에서는 심화·보충형 수준별 학습의 효과를 높이기 위한 팀 분류 방법과 팀 분류 시기에 대해 비교·분석해보고 수준별 학습자들의 학습 수준 자가 인식 정도가 학습 효과에 미치는 영향에 대해서도 연구해보고자 한다.

## 참고문헌

- [1] 안유정, "맞춤형 학습 유형에 따른 학습자들의 객체지향 프로그래밍 개념 이해도 분석," 한국정보처리학회, 제 33회 춘계학술발표대회 논문집, 춘천대학교, 대한민국, 261-263쪽, 2010년
- [2] 안유정, "컴퓨터 프로그래밍 수업에서 맞춤형 학습 유형과 취업률과의 관계에 관한 연구," 한국정보처리학회, 제 32회 추계학술발표대회 논문집, 건국대학교, 대한민국, 265-266쪽, 2009년.
- [3] 안유정, 김경아, "수준별 맞춤형 전공 동아리 운영에서 학습 효과에 영향을 미치는 요인 분석," 명지전문대학 정보기술연구소 논문집 제 32호, 83-96쪽. 2008년.
- [4] 안유정, 김경아, "컴퓨터 프로그래밍 수업에서 학습자 수준에 따른 맞춤형 교수 사례 연구," 명지전문대학 정보기술연구소 논문집 제 30호, 271-278쪽, 2006년.
- [5] 김경아, 김계현, 구윤모, "프로그래밍 언어 교육에 있어서 적용 가능한 효과적 교수-학습 방법에 대한 연구," 명지

- 전문대학 정보기술연구소 논문집 제 4집, 63-71쪽, 2004년.
- [6] 한일환, 안미리, 차재혁, "수준별 교육과정을 위한 학습 모형의 비교(Visual Basic 반복문을 중심으로)," 한국컴퓨터교육학회, 컴퓨터교육학회논문지, 제 4권, 제 1호, 119-126쪽, 2001년.
- [7] 김미량, "컴퓨터 프로그래밍 교육에 적용 가능한 효과적 교수방법의 탐색적 대안," 한국컴퓨터교육학회, 컴퓨터교육학회논문지, 제 5권, 제 3호, 1-9쪽, 2002년.
- [8] 손경아, "문제중심학습(Problem-Based Learning)을 적용한 비주얼베이직(VisualBasic) 프로그래밍 학습의 사례연구," 컴퓨터교육학회논문지, 제 5권, 제 1호, 17-25쪽, 2002년
- [9] 김경아, 문남미, "수준별 프로그래밍 교육을 위한 단계별 클러스터링 기반 추천시스템," 한국컴퓨터정보학회논문지, 제 15권, 제 8호, 51-57쪽, 2010년.
- [10] 김혜은, 유석중, "수준별 학습과 학습 관심도를 고려한 학습평가시스템," 한국컴퓨터정보학회논문지, 제 13권, 제 6호, 69-76쪽, 2008년.
- [11] 김정렬, 한희정, "원어민 영어 보조교사 활용 수업이 학습자의 수준별 영어 학습에 미치는 영향," 교육과정평가원, 교육과정평가연구 제 12권, 125-144쪽, 2009년.
- [12] 교육개발원, 수준별 교육과정안, 1996년.
- [13] 김경아, 안유정, "프로그래밍 언어 교육에 있어서 적용 가능한 효과적 평가 방법에 대한 탐색," 명지전문대학 정보기술연구소 논문집 제 6집, 41-52쪽, 2006년.

## 저자 소개



### 안 유 정

1991 : 이화여자대학교 이학사.  
 1995 : 이화여자대학교 이학석사.  
 2000 : 홍익대학교 이학박사.  
 2001 - 현재 : 명지전문대학 컴퓨터 정보과 부교수  
 관심분야 : 객체지향 프로그래밍, 수준별 교육, 컴퓨터교육



### 김 경 아

1990 : 이화여자대학교 이학사  
 1992 : 이화여자대학교 이학석사  
 2001 : 이화여자대학교 공학박사  
 2002 - 현재 : 명지전문대학 컴퓨터 정보과 부교수  
 관심분야 : 프로그래밍 언어, 이러닝, 추천시스템, 컴퓨터교육