

2012년 신 개정 정보 교육과정의 ‘문제 해결 방법과 절차’ 영역을 위한 수업 모형 개발

현태익*, 최재혁**, 이종희***

Development of Teaching Model for ‘Problem-solving methods and procedures’ section in the 2012’s revised Informatics curriculum

Tae-Ik Hyun *, Jae-Hyuk Choi **, Jong-Hee Lee ***

요 약

이 연구는 일반계고등학교 신 개정 정보 교육과정의 ‘문제 해결 방법과 절차’영역의 효과적인 교수·학습을 위한 수업 모형을 개발하고 그것의 효과성을 검증하는데 목적이 있다. 중학교에서 이 영역을 학습하지 못한 일반계 고등학생들을 위해 중학교 교육과정을 포함할 필요가 있고, 학습자의 발달 단계에 알맞은 프로그래밍 언어를 채택하여 인지적 부담을 줄일 필요가 있다. 재미있게 프로그래밍 할 수 있는 퍼즐을 학습 예제로 사용할 필요가 있다. 연구 수행을 위한 연구 방법은 ADDIE 모형에 따라 수행한다. 수업 모형의 프로그래밍 언어로는 파이썬을 선택한다. 이 수업 모형의 효과성을 평가하고자 컴퓨터 부 학생들을 대상으로 수업을 진행하였고, 컴퓨터교육과 예비 교사들이 모의수업을 하였다. 예비교사의 75%가 수업 모형에 만족하였다. 2012년 3월부터 23시간동안 일반계고등학교 정보 교과 수업에 적용하였다. 수업 전·후에 문제해결력 향상 여부를 알아보기 위해 PSI 검사를 하고, 중간고사 정보 점수와 비교하여 약간의 정상관계가 있다고 분석하였다. 따라서 개발한 수업 모형이 위 영역의 교수·학습에 효과가 있다고 분석한다. 그러므로 정보 교사들의 교수·학습 수업 모형의 지표가 되고, 예비 교사의 교육 자료로 활용하는 것을 제안한다.

▶ Keywords : 교육과정, 문제 해결 방법과 절차, 수업 모형

Abstract

-
- 제1저자 : 현태익 • 교신저자 : 이종희
 - 투고일 : 2012. 2. 23, 심사일 : 2012. 6. 1, 게재확정일 : 2012. 8. 14.
 - * 신라대학교 컴퓨터교육과(Dept. of Computer Education, Silla University)
 - * 신라대학교 컴퓨터교육과(Dept. of Computer Education, Silla University)
 - * 신라대학교 컴퓨터공학부(Dept. of Computer Engineering, Silla University)

The purpose of this study is to develop an effective teaching model for the "Problem solving methods and procedures" section in the revised academic high school informatics curriculum, verify its effectiveness, make the subject more effective and appealing to teachers as well as students. The model includes a middle school level informatics curriculum for the students who have yet to learn the section. This development follows the ADDIE model, and the Python programming language is adopted for the model. Using the model, classes were conducted with two groups: high school computer club students and undergraduate students majoring in computer education. Of the undergraduate students 75% responded positively to the model. This model was applied in the actual high school classroom teaching for 23 class-hours in the spring semester 2012. The Pearson correlation coefficient that verifies the correspondence between the PSI score and the informatics midterm exam grade is .247, which reflects a weak positive correlation. The result of the study showed that the developed teaching model is an effective tool in educating students about the "problem solving methods and procedures". The model is to be a cornerstone of teaching/learning plans for informatics at academic high school as well as training materials for pre-service teachers.

▶ Keywords : curriculum, problem-solving methods and procedures, Teaching Model

I. 서 론

새로운 국가· 사회적 요구인 문제 해결 능력 향상과 정보를 생산하는 주체로서의 인재 양성을 위해서는 컴퓨터 과학 원리 교육이 필요하다. Jennette M. Wing(2006)은 정보과학적 사고(CT: Computational Thinking)가 컴퓨터 과학자뿐만 아니라, 모든 사람에게 기본적으로 필요한 기술이라고 주장하였고[1], 정보과학적 사고는 컴퓨터 과학의 중심 개념이라 할 수 있다[2].

우리나라에서도 정보과학적 사고의 중요성을 고려하여 2007년 3월에 논리적인 사고력과 창의력 신장을 위한 컴퓨터 과학의 내용이 포함된 정보 교육과정으로 전면 개정하였다. 특히 2012년 신 개정 정보 교육과정에서는 정보과학적 사고력을 향상시킬 수 있는 '문제 해결 방법과 절차' 영역의 비중을 30%에서 35%로 강화하였다.

소양교육 중심 교육과정에서 정보 과학 원리 교육으로의 변환은 많은 현장 교사들에게 인지적 부담을 주고 있으며, 특히 신설된 '문제 해결 방법과 절차' 영역은 프로그래밍이 주된 내용이기 때문에 교사들에게도 많은 부담을 주고 있다.

일반계고등학교 교육과정에서의 정보과학적 사고력 향상을 위한 교수· 학습 모형에 대한 연구는 거의 이루어지지 않았다. 대부분의 선행 연구들은 초·중학생을 대상으로 문제 중심 학습 기반 로봇 프로그래밍의 효과 분석이나 집단 지성

활용 프로그래밍 수업 모형 개발을 통한 문제 해결력을 향상 연구로 한정되어 있다[3~6].

본 연구의 필요성은 다음과 같다.

첫째, 중·고교 교육과정을 비교하여 '문제 해결 방법과 절차' 영역의 교육 내용을 위계성 있게 재구성할 필요가 있다.

둘째, 프로그래밍 교육에서 인지 발달에 알맞으며, 문제 해결력 향상에 효율적인 프로그래밍 언어의 연구가 필요하다.

셋째, 어렵게 느끼는 프로그래밍을 흥미 있게 하기 위해 Keller의 ARCS 이론에 근거하여 동기 유발 할 수 있고 Csikszentmihalyi의 몰입(flow)을 느낄 수 있도록, 학습 내용의 예제로 사용할 퍼즐의 연구가 필요하다.

연구 수행을 위한 구체적인 연구 방법은 교수체제설계(ISD) 방법인 ADDIE 모형에 의거하여 수행하며, 개발 절차는 연구내용에서 표로 제시한다.

개발된 수업 모형은 2012년부터 일반계고등학교 정보 교과의 교수· 학습에 적용하여 수업의 효율성을 높이고, 또한 정보 교사들의 교수· 학습 수업 모형의 지표가 되고, 예비 교사의 교육 자료로 활용하고자하는 목적이 있다.

II. 이론적 배경

1. 2012년 신 개정 정보 교과 교육과정

1998년 발표된 7차 교육과정에서의 정보 교과 교육과정은 컴퓨터 소양 교육이 중심이었으며, 2007년 개정 정보 교과는 선택 교과로 컴퓨터 과학이 중심이 된 교육과정으로 개편되었다. 2012년 3월 21일 교육과학기술부 고시 제2012-3호로 선택 심화 교과로 변경 고시하였다[7]. 2007년 개정 교육과정과의 차이는 다음과 같다.

첫째, '정보 기술과 정보 윤리' 영역이 4영역에서 1영역으로, '정보 기기의 구성과 동작'은 2영역, '정보의 표현과 관리'는 3영역, '문제 해결 방법과 절차'는 4영역으로 바뀌었다. 이는 정보 윤리 교육의 중요성을 강화하기 위해 가장 먼저 배울 수 있도록 1영역으로 이동하게 되었다.

둘째, 중학교 교육과정이 '정보1', '정보2', '정보3'으로 나뉘어 있던 것을 '정보' 하나로 통일되었다.

셋째, '정보 기술과 정보 윤리' 영역은 20%에서 25%로, '문제 해결 방법과 절차' 영역은 30%에서 35%로 증가하였고, '정보 기기의 구성과 동작' 영역은 20% 그대로, '정보의 표현과 관리' 영역은 30%에서 20%로 감소하였다.

넷째, 정보 교과의 교육 내용을 단순화시키기 위해 각 영역의 항목 수가 중학교는 47개에서 34개로, 고등학교는 25개에서 22개로 평균 22% 감소되었다.

2. 문제 해결 방법과 절차 영역 교육과정

창의적 과학 교육을 위한 STEAM 교육에서 여러 분야의 문제 해결 방법과 절차에 관한 연구가 이루어지고 있다. 정보 교과에서도 정보과학적 사고 능력 향상 교육을 위하여 도입된 '문제 해결 방법과 절차' 영역을 2007년 개정 교육과정과 2012년 신 개정 교육과정을 비교하여 <표 1>과 <표 2>에 나타내었다.

두 교육과정의 차이점은 다음과 같다.

첫째, 고등학교 '문제 해결 방법과 절차' 영역의 교육과정에서는 '객체지향 프로그래밍' 요소가 삭제되었으며, '문제의 구조화' 요소에서 개념만을 기술하도록 하였다.

둘째, 고등학교 프로그래밍은 중학교에서 정보 교과를 배우지 않은 학생과 중학교에서 배운 프로그래밍 언어와 고등학

교에서 배우는 프로그래밍 언어가 다른 학생들을 위하여 프로그래밍의 기초 항목이 포함되었다.

셋째, 컴퓨터 과학의 개념 이해에서 필수적으로 알아야만 하는 이진트리와 그래프 탐색 부분을 고등학교 과정에 추가되었다.

넷째, 중학교 과정에서 배우기 어려운 알고리즘 구현 부분은 삭제되었다.

다섯째, 중학교 과정에서는 정렬과 탐색의 방법만 배우고 구현 부분은 고등학교에서 다루도록 하였다.

여섯째, 2007년 개정 교육과정에서는 중학교 '정보1'에서 프로그래밍을 배우고, '정보2'에서 알고리즘을 배우도록 하였으나, 2012년 신 개정 교육과정에서는 알고리즘을 먼저 배우고 프로그래밍을 배우도록 순서를 변경하여 프로그래밍 교육의 효율성을 높였다.

표 1. 고등학교 신·구 문제 해결 방법과 절차 영역 교육과정
Table 1. new and old high school 'problem-solving methods and procedures' section curriculum

2007년 개정 교육과정	2012년 신 개정 교육과정
<ul style="list-style-type: none"> ○문제해결전략 <ul style="list-style-type: none"> • 문제의 구조화 • 문제 해결 전략의 비교 ○구조적 프로그래밍 <ul style="list-style-type: none"> • 제어문의 활용 • 함수의 활용 ○객체지향 프로그래밍 <ul style="list-style-type: none"> • 객체지향의 개념 • 객체지향 문제분석 및 설계 	<ul style="list-style-type: none"> ○문제 해결 전략 <ul style="list-style-type: none"> • 문제의 구조화 • 문제 해결 전략과 분석 ○프로그래밍 <ul style="list-style-type: none"> • 프로그래밍의 기초 • 배열의 활용 • 함수의 활용 ○알고리즘의 응용 <ul style="list-style-type: none"> • 정렬과 탐색 알고리즘의 구현 • 이진 트리와 그래프의 탐색

표 2. 중학교 신·구 문제 해결 방법과 절차 영역 교육과정
Table 2. new and old middle school 'problem-solving methods and procedures' section curriculum

	2007년 개정 교육과정	2012년 신 개정 교육과정
정보 1	<ul style="list-style-type: none"> ○문제와 문제 해결 과정 <ul style="list-style-type: none"> • 문제의 분석과 표현 • 문제 해결 과정 ○프로그래밍의 기초 <ul style="list-style-type: none"> • 변수의 개념과 활용 • 자료의 입력과 출력 • 제어문의 이해 	<ul style="list-style-type: none"> ○문제 해결 방법 <ul style="list-style-type: none"> • 문제의 분석과 표현 • 문제 해결 과정 ○문제 해결 절차 <ul style="list-style-type: none"> • 알고리즘의 이해와 표현 • 알고리즘의 설계와 작성 • 정렬과 탐색 방법의 이해 ○프로그래밍의 기초 <ul style="list-style-type: none"> • 프로그래밍 언어의 이해 • 변수의 개념과 활용 • 자료의 입력과 출력 • 제어문의 이해
정보 2	<ul style="list-style-type: none"> ○알고리즘의 개요 <ul style="list-style-type: none"> • 알고리즘의 이해 • 알고리즘의 표현 ○알고리즘의 실제 <ul style="list-style-type: none"> • 알고리즘의 설계 • 알고리즘의 분석 • 알고리즘의 구현 	
정보 3	<ul style="list-style-type: none"> ○자료의 정렬 <ul style="list-style-type: none"> • 자료의 정렬 방법 • 정렬 알고리즘의 구현 ○자료의 탐색 <ul style="list-style-type: none"> • 자료의 탐색 방법 • 탐색 알고리즘의 구현 	

3. 프로그래밍 언어 교육

교육의 목적 중의 하나는 학습자가 교육을 통해 습득한 내용을 일상의 문제 상황에 전이시켜 문제 상황을 해결할 수 있는 문제 해결 능력을 기르는데 있다.

정미연(2008)은 문제 해결에서 ‘문제’는 구체적이고 확실한 해결의 방법을 구하기 어렵고, 문제 해결을 위해서 다단계에 걸친 다양한 사고가 요구되는 문제라고 하였다[8].

Polya는 문제 해결 과정을 ‘문제의 이해, 계획의 작성, 계획의 실행, 반성의 단계’로 나누고 있으며, 어느 한 단계라도 빠뜨리게 되면 바람직하지 못한 결과를 가져올 수 있다고 하였다[9].

김현정(2004)은 문제 해결 능력은 실제로 자신이 해결할 문제에서 주의 집중해야 할 사고 과정을 순서 짓고, 실행, 반성하는데서 발달하게 된다. 즉, 알고리즘적 사고를 통해 문제 해결 능력이 발달한다고 하였다[10].

알고리즘적인 사고력과 문제 해결력을 키우기 위한 프로그래밍 언어 교육에서 프로그래밍 언어는 학생들에게는 배우기 어려운 영역이다. Jenkins, Tony(2003)는 프로그래밍을 어렵게 만드는 요인으로 다중 기술, 다중 과정, 언어, 교육적인 생소함, 흥미, 편견과 이미지 등을 열거하고 있다[11].

프로그래밍 교육은 학습자의 인지적 수준과 교수자의 교수 방법에 따라 학업 커다란 영향을 줄 수 있다. 따라서 학습자의 인지 발달 단계를 고려하여 학습자들에게 적합한 언어를 선정하여 교육해야 한다.

알고리즘적인 사고력과 문제 해결력을 키우는 것이 목적인 초, 중, 고등학교 교육에서는 교육용 프로그래밍 언어(EPL: Educational Programming Language)의 사용을 권장하고 있다. 범용 언어는 문법의 내용이 복잡하고 그 양이 방대하며, 다양한 패키지화된 어플리케이션이 존재한다. 하지만 그 목적이 알고리즘적 사고력과 문제 해결력을 키우는 것이므로 오히려 쉽고 배우기가 용이해야 한다[12].

진영학, 허민, 김영식 (2010)은 중학교 8종 교과서를 분석하여 다음 <표 3>과 같이 제시하였다[13].

진영학 등은 “각각의 프로그래밍 언어들은 장점과 단점을 함께 가지고 있기 때문에 학습자의 인지 발달, 학습 동기 등 학업 성취에 영향을 주는 다양한 구인을 고려하여 적절한 프로그래밍 언어를 선택해야 할 것이다.”라고 하였다.

표 3. 정보교과서에서 사용한 프로그래밍 언어의 종류와 특징
Table 3. Kinds and individuality of using programming language in informatics textbooks

구분	종류	특징
금성출판사/ (주미래엔)	스크래치	<ul style="list-style-type: none"> * 직관적 인터페이스로 인지 부하 감소 * 시간적, 상호 작용적 프로그래밍 환경 * 현업에서 널리 사용되기에는 한계가 있음
두산동아 영진출판사	의사코드	<ul style="list-style-type: none"> * 문법에 의한 제약 없으므로 작성 쉬움 * 프로그래밍 언어 교육을 위해 사용할 경우, 언어를 쉽게 이해할 수 있음 * 문서화 및 검토용 적합, 결과 미확인
생능출판사/ 삼양미디어	파이썬	<ul style="list-style-type: none"> * 인터프리터 언어, 사용법이 간단 * 현업에서 널리 사용되는 언어로서 학교급간 교육과정의 연계가 가능함 * 문법이 간단하여 인지 부하를 감소시킴
지학사/ 천재교육	C 언어	<ul style="list-style-type: none"> * 컴파일러 언어, 사용법이 다소 복잡함 * 현업에서 널리 사용되는 언어로서 학교급간 교육과정의 연계가 가능함 * 복잡한 문법으로 인지 부하를 가져옴

4. 수업 모형 선행 연구

수업 모형은 교수·학습 절차 또는 단계를 지칭하는 용어로, 교수·학습 모형, 교수 모형, 교수·학습 유형 등과 같은 의미로 사용된다. 계획, 진단, 지도, 평가 등과 같이 학습 단계를 하나의 모형으로 보기도 한다. 조이스와 웨일(Joyce & Weil)은 수업 모형을 교육과정이나 교과과정을 구성하거나, 수업 자료를 선정하고, 교사의 행위를 안내하는데 이용될 수 있는 형태나 계획이라고 정의하였다[14].

정보 교육을 위한 수업 모형의 선행 연구는 다음과 같다. 이송현, 김갑수(2008)는 초등학생을 위한 프로젝트 기반 프로그래밍 수업모형을 개발하고 적용하였다[3].

배학진, 이은경, 이영준(2009)은 문제 중심 학습 기반 프로그래밍 교수·학습 모형을 개발하였다[4].

송정범, 백성혜, 이태욱(2009)은 로봇 프로그래밍 학습이 남, 여학생들에게 몰입 수준을 올려 문제 해결력을 높일 수 있는지를 연구하였다[5].

김길모, 김성식(2011)은 집단지성을 활용한 문제 중심 기반 프로그래밍 수업 모형을 개발하여 학습자의 문제 해결력과 프로그래밍 태도에 어떠한 영향을 미치는지 분석하였다[6].

신설된 ‘문제해결방법과 절차’ 영역의 성취기준과 평가기준이 필요하였기 때문에 김종혜, 김경훈, 이원규(2007)은 개정된 중학교 정보 교육과정의 교수학습방법과 평가 방법의 가이드라인을 제시하는 연구가 있었다[15].

박정선(2010)은 중학교 개정 정보 교육과정에 연계하여 정보영재를 위한 심화형 교육과정을 개발하였다.[16].

위의 선행 연구들은 대체로 초·중학생을 대상으로 문제 중심 학습 기반 로봇 프로그래밍이나 집단 지성 활용 프로그래밍 수업 모형을 개발하여 문제 해결력 향상을 연구하였다.

일반계고등학교에서는 2012년부터 7차 개정 '정보' 과목이 적용된다. 4가지 영역 중에서 '문제 해결 방법과 절차' 영역이 가장 어려운 영역으로, 효과적인 수업 모형이 필요하다. 본 연구에서는 2012년 신 개정 교육과정을 바탕으로 고등학교의 교육과정을 재구성하여, 일반계고등학교 학생들의 문제 해결력 향상을 위한 수업 모형을 개발한다.

III. 연구 내용

1. 수업 모형 설계 원리

첫째, 중학교 과정을 학습하지 못한 학습자들을 위하여 중·고등학교의 교육과정을 재구성한다.

둘째, 수업 내용의 예를 실생활과 관련된 문제 상황 및 문제를 설계해서 프로그래밍을 통한 문제 해결 활동이 되도록 한다. 특히 ARCS 모형에 의해 학습자의 흥미 유발과, Csikszentmihalyi의 몰입으로 프로그래밍을 재미있게 학습하기 위하여 퍼즐을 이용한다.

셋째, 학습자의 인지 발달 특성에 맞는 프로그래밍 언어를 채택한다. 중·고등학교의 교과서를 분석 비교하여 교육용 프로그래밍 언어인 파이썬을 채택하여 프로그래밍 한다.

넷째, 문제 해결 방법과 절차 영역의 학습 시간을 연간 학습 시간의 35%인 23시간으로 한다.

다섯째, 교수·학습의 형태는 수업 단계별 특성에 따라 적합한 형태로 설계한다. 학습자의 학습 양식에 유의하여, 개별 학습과 동료 학습, 짝 프로그래밍 등을 적용한다.

2. 모형 개발 절차

교수체제설계(Instructional System Design: ISD)의 기본 과정은 분석(Analysis), 설계(Design), 개발(Development), 실행(Implementation), 평가(Evaluation)로 나누어진다. 일반계 고등학교 정보 교육과정의 문제 해결 방법과 절차 영역의 수업 모형을 <그림 1>의 절차에 따라 개발한다.

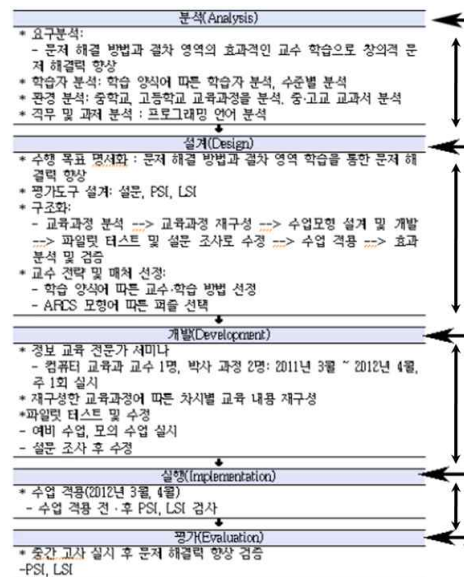


그림 1. ISD ADDIE 모형에 따른 개발 절차
Fig 1. develop process with ISD ADDIE model

3. 교과서 사용 프로그래밍 언어 비교 및 채택

2012학년도부터 일반계 고등학교에서 채택되는 '정보' 교과서 7종을 분석하였다. 본 연구자가 분석한 교과서에서 사용된 언어는 <표 4>와 같다.

표 4. 고등학교 정보 교과서에서 사용하는 언어
Table 4. Using programming language in informatics textbook of highschool

출판사	사용언어	출판사	사용언어
형설출판사	C 언어	(주)이오북스	스크래치
천재교육	C 언어	천재교육	파이썬
삼양미디어	C 언어	(주)미래엔	파이썬
씨마스	C 언어		

<표 4>에서 보면 가장 많이 사용하고 있는 언어는 C 언어이다. 하지만 C 언어를 사용하고 있는 대부분의 교과서는 다른 언어도 같이 소개하고 있다.

중학교 교과서에서 사용하는 언어와 고등학교에서 사용하는 언어를 비교하고, 본 연구자가 설문 조사한 결과를 토대로, 파이썬을 일반계고등학교 '정보' 교과 4단원인 '문제 해결 방법과 절차' 영역의 새로운 수업 모형에 사용할 언어로 채택하였다. 그 이유는 다음과 같다.

첫째, 피아제의 인지 발달 이론에 의하면, 고등학생은 형식적 조작기(formal operational stage) 단계이다. 이 단계에서는 가설, 연역적 사고와 체계적 사고를 하는 단계로 피아제가 이러한 사고 발달에 적합한 코드 작성 언어이기 때문이다.

둘째, 인터프리터 언어로서 사용법이 간단하다.

셋째, 현업에서 널리 사용되는 언어로서 학교급간 교육과정의 연계가 가능하다.

넷째, 문법이 간단하여 인지 부하를 감소시킬 수 있다.

4. 교수 · 학습 내용 재구성

교과서 내용을 분석하여 23차시로 내용을 재구성하였다. 1학기 17주 매주 2시간 운영으로 학기당 34시간이므로 1년에 68시간이 할당된다. 이 중에 4단원 '문제 해결 방법과 절차' 영역에 35%를 할당하면 약 23시간이 된다. 본 연구에서의 교육과정은 중학교 과정과 고등학교 과정을 통합한 교육과정으로 23시간을 3영역으로 나누었다.

'문제해결 전략' 영역에서는 기본적인 알고리즘을 학습한다. 알고리즘 표현과 작성법을 중심으로 4시간을 배정하였다. 학생들의 이해를 돕기 위해서는 적절한 교수·학습 방법을 적용하여야 한다.

'프로그래밍' 영역에서는 프로그래밍의 정의와 기본적인 제어 구조, 변수, 함수 설명을 하고, 또한 기본적인 원리를 피아제 언어로 구현하도록 한다. 기본형과 제어문을 익히는 사용 단계, 수정 단계, 창조 단계를 거쳐 문제 해결력을 향상시킬 수 있도록 한다. 그리고 학생들의 흥미도를 높일 수 있도록 퍼즐을 활용한 프로그래밍 교육을 시행한다. 학생들이 가장 중점적으로 학습하고 문제를 해결하여야 할 시간을 많이 주어야 하기 때문에 11시간을 배정하였다.

'알고리즘의 응용' 영역에서는 정렬과 탐색 알고리즘을 학습하고 파이썬으로 구현한다. 그리고 트리와 그래프를 실생활에서 나타나는 예를 가지고 이해할 수 있도록 8시간을 배정하였다.

학생들에게 인지적 부담을 주는 프로그래밍 학습에서 흥미를 유발할 수 있도록 Keller의 ARCS 모형[17]에 근거하여 <그림 2>에서와 같이 호기심(주의 집중)을 유발시키고 동기(관련성) 부여를 할 수 있는 퍼즐을 학습의 예제로 여러 문헌을 참고하여 선별하여 사용하였다[18 - 21].

퍼즐을 활용한 교육은 흥미 있고 정보과학적 사고를 경험할 수 있다고 한다[22]. 프로그래밍 교육에 퍼즐을 적용하면 좀 더 흥미 있고, 기술 수준과 난이도가 높은 과제를 해결해 봄으로써 Csikszentmihalyi의 몰입(flow)을 느낄 수 있어

[23], 정보과학적 사고 능력 향상에 도움이 될 것이다.

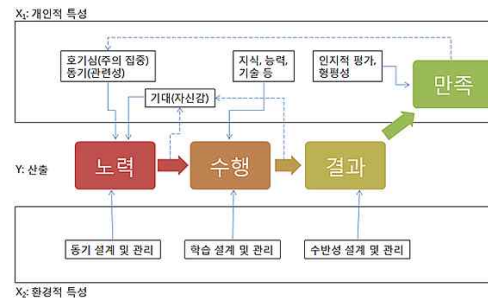


그림 2 동기 및 수행 거시 모형
Fig 2 Macro Model of Motivation and Performance

5. 수업 모형 개발

전문가의 조언과 협의에 의해 시간을 편성하고 교수·학습 내용을 재구성하여 '문제 해결 방법과 절차' 영역의 교육과정을 위해 개발한 수업 모형을 대영역 별로 나누어 제시한다.

<표 5-1>은 '문제 해결 전략' 영역의 수업 모형으로 중학교 과정 '문제 해결 방법'과 '문제 해결 절차' 영역을 포함하였다.

표 5-1. '문제 해결 방법 및 절차' 영역 수업 모형 - 1 영역
Table 5-1. Teaching model for 'problem-solving methods and procedures' section - part 1.

중영역	소영역	세부 내용	차시별 주제	학습 내용
I. 문제 해결 전략 (4)	01. 문제의 구조화 (3)	1. 문제의 구조화	① 문제 해결 및 구조화 방법	1) 문제 해결 방법 2) 문제의 구조화 방법
		2. 알고리즘 설계와 작성	② 알고리즘의 표현	1) 알고리즘 이해하기 2) 알고리즘 표현 방법
		3. 알고리즘 설계와 작성	③ 알고리즘 설계 및 작성	1) 알고리즘 작성의 중요성 2) 알고리즘 작성의 기본 구조 3) 알고리즘 설계
	02. 문제 해결 전략과 분석 (1)	1. 알고리즘 성능 분석	④ 알고리즘 성능 분석 방법	1) 알고리즘 분석의 필요성 2) 알고리즘의 효율성 분석 3) 수행 시간 분석 기준

교수자는 학습자들이 문제해결전략으로 알고리즘적 사고에 익숙하도록 학습자의 학습 양식에 적합한 교수방법으로 지도한다. <그림 3>과 같이 미흡한 학생은 동료 학습을 적용하

여 보충 및 피드백으로 성취도를 높이고, 학습자는 알고리즘 설계에 익숙하도록 협동학습 등을 통해 학습한다.

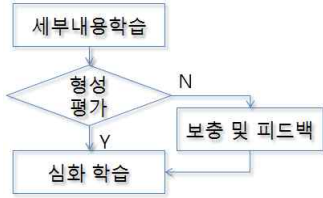


그림 3. 차시별 학습 순서도
Fig 3. Flow chat for a period

<표 5-2>는 '프로그래밍' 영역의 수업 모형이다. 파이썬 언어를 채택하였고 퍼즐을 예제 문제로 사용하였다. 차시별로 시범 · 실습법, 동료 교수법, 협동 학습 등 세부 내용에 알맞은 교수·학습 방법을 적용한다.

표 5-2 '문제 해결 방법 및 절차' 영역 수업 모형 - 2 영역
Table 5-2. Teaching model for 'problem-solving methods and procedures' section - part 2

중영역	소영역	세부 내용	차시별 주제	학습 내용
II. 프로그래밍 (11)	01. 프로그래밍의 기초(6)	1. 프로그래밍 언어의 이해	⑤ 프로그래밍 언어 소개, 프로그래밍	1) 구조적 프로그래밍 2) 객체 지향 프로그래밍
		2. 파이썬 프로그래밍	⑥ 기본 자료형	1) 변수의 개념과 활용 2) 숫자형(Number) 3) 문자열(String)
			⑦ 자료형	3) List 4) Set 5) Tuple 6) Dictionary
		3. 자료의 입력과 출력	⑧ 입력과 출력	1) 자료 입력 2) 자료 출력
	4. 제어문의 이해	⑨ 조건문의 이해와 활용	1) 조건문의 이해 2) 조건문의 활용	
		⑩ 반복문의 이해와 활용, 기타 제어문	1) 반복문의 이해 2) 반복문의 활용	
	02. 배열의 활용 (2)	1. 배열이란?	⑪ 배열의 기본	1) 배열의 개념과 필요성 2) 배열의 선언과 처리
		2. 배열의 활용	⑫ 배열을 이용한 프로그램	1) 1차원 배열 프로그램 2) 2차원 배열 프로그램
	03.	1.	⑬ 함수의	1) 함수의 개념과

함수의 활용 (2)	함수	개념과 필요성 이해	효율성 이해 2) 지역 변수와 전역 변수 3) 함수의 종류
	04. 종합	2. 함수 활용 1. 실습	⑭ 함수를 활용한 프로그램

일반계고등학교 학생들은 아직 프로그래밍에 대한 경험이 거의 없다. 그러므로 <그림 4>에 제시한 다음의 단계로 과제별 학습을 한다.

첫째, 주어진 과제를 잘 읽어 이해하고, 문제를 해결하기 위하여 문제를 구조화한다.

둘째, 가장 알맞은 알고리즘을 작성한다. 작성한 알고리즘의 오류가 없는지를 확인한다. 만약 오류가 있으면, 다시 분석하여 구조화한 후 오류를 수정한다.

셋째, 파이썬 언어로 프로그램을 구현한다. 이 단계에서 아직은 프로그램 작성에 익숙하지 않은 학생들이므로, 미리 작성한 코드를 참고하도록 한다. 즉, 교사가 미리 작성한 코드에서 중요한 부분을 비워두고 학생들에게 채워 넣도록 한다.

넷째, 만약 해결하지 못하면, 다시 사용 단계의 기본 학습인 사용 단계를 교사나 동료의 도움을 받아 보충 학습을 한 후, 수정 단계를 해결하도록 하고 심화 문제를 가지고 직접 코드를 작성하도록 하는 창조 단계로 간다. 혹은 주어진 문제를 더욱 발전시켜 업그레이드 할 수 있도록 한다. 직접 코드를 작성하기에 힘들어 하는 학생은 교사 혹은 동료가 도와서 해결할 수 있도록 한다.

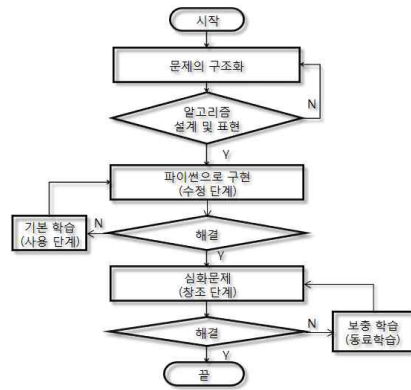


그림 4. 과제별 학습 순서도
Fig 4. Flow chat for a problem

<표 5-3>은 '알고리즘의 응용' 영역의 수업 모형이다.

표 5-3. '문제 해결 방법 및 절차' 영역 수업 모형 - 3 영역
Table 5-3. Teaching model for 'problem-solving methods and procedures' section - part 3

중영역	소영역	세부 내용	차시별 주제	학습 내용
III. 알고리즘의 응용 (8)	01. 정렬과 탐색 알고리즘의 구현 (4)	1. 정렬 알고리즘 구현	⑯ 버블 정렬, 선택 정렬 알고리즘 구현	1) 버블 정렬 알고리즘 1) 선택 정렬 알고리즘 2) 선택 정렬 알고리즘 구현
			⑰ 삽입 정렬 알고리즘 구현	1) 삽입 정렬 알고리즘 2) 삽입 정렬 알고리즘 구현 3) 정렬 방법의 비교
		2. 탐색 알고리즘의 구현	⑱ 순차 탐색 알고리즘의 구현	1) 순차 탐색 알고리즘 2) 순차 탐색 알고리즘의 구현
			⑲ 이진 탐색 알고리즘의 구현	1) 이진 탐색 알고리즘 2) 이진 탐색 알고리즘의 구현
	02 이진트리와 그래프의 탐색 (4)	1. 이진트리	⑳ 이진트리의 표현/이진트리의 순회	1) 이진트리의 정의 2) 이진트리의 표현 방법 ① 중위순회, ② 전위순회 ③ 후위순회
			㉑ 이진 탐색 트리의 삽입, 삭제	1) 삽입, 삭제, 대체 노드 찾기 2) 이진 탐색 트리의 시간 복잡도
		2. 그래프	㉒ 그래프의 개요와 용어, 그래프의 표현 방법	1) 그래프의 정의와 역사 2) 그래프의 기본 용어 3) 그래프의 종류 1) 그래프 표현의 예 -인접 행렬, -인접 리스트
			㉓ 그래프의 탐색, 최단 경로 문제	1) 깊이 우선 검색 2) 너비 우선 검색 1) 다익스트라 알고리즘

진트리와 그래프의 탐색 학습은 프로그램으로 구현하지는 않고, 학습자들의 흥미와 관심을 유도하기 위해 언플러그드 학습법[24]을 적용한다.

IV. 실험 및 결과

1. 모의 수업 적용

2011년 3월부터 12월까지 부산 시내 일반계고등학교에서 창의적 체험활동 컴퓨터 부 학생들을 대상으로 개발한 수업 모형을 적용하였다. 또한 2011년 2학기 부산의 S 대학교 컴퓨터교육과 학생들에게 재구성한 교육과정 수업 모형을 소개하고 학생들로 하여금 모의수업을 진행하도록 하였다.

2. 설문 조사

수업 모형의 효과 분석과 프로그래밍 언어의 적합성 여부를 조사하기 위해 컴퓨터부 학생 31명(고등학교 1학년 14명, 2학년 17명)과 컴퓨터교육과 학생 21명을 대상으로 설문 조사를 실시하였다. 설문지는 전문가의 도움을 받아 고등학생용과 예비 교사용으로 구분하여 고등학생용은 8문항, 예비 교사용은 10문항으로 구성하였으며, 설문 문항은 <표 6>과 같다.

표 6. 설문 문항 내용
Table 6. Contents of question subject

문항	내용	비고
1	프로그래밍 경험	공통
2	중학교 과정에 적합한 언어	공통
3	고등학교 과정에 적합한 언어	공통
4	중·고 학습 언어의 동일성	공통
5	파이썬 언어의 흥미도	공통
6	퍼즐 활용 교육의 효과	공통
7	내용의 난이도	고등학생
8	학습량의 적절성	고등학생
7-1	수업 모형의 적합성	예비 교사
8-1	차시별 수업 분량의 적절성	예비 교사
9	정보과학적 사고력 습관 형성	예비 교사
10	자신감과 성취감 고취	예비 교사

<그림 5>는 <표 5-3>의 수업 모형에 대한 과제별 학습 순서도이다.

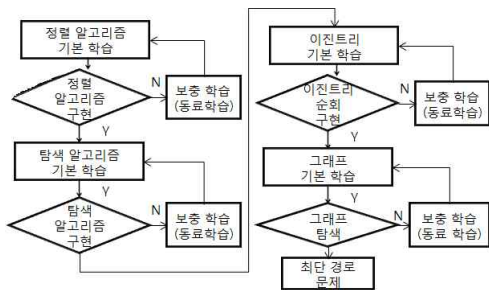


그림 5. 과제별 학습 순서도
Fig 5. Flow chat for a problem

이진트리와 그래프의 필요성 및 표현 방법을 알기 위해 탐구 학습 혹은 협동 학습으로 기본적인 개념을 확립시킨다. 이

[문항 1]의 설문 결과, 고등학생은 31명 중 활동 전에 프로그래밍 경험이 있는 학생은 16%로, 84%의 고등학교 학생들은 정규 교육과정에서 프로그래밍 과정이 없기 때문에 프로그래밍 언어를 배우지 못한 것으로 조사되었다. 사범대학 컴

퓨터교육과 2, 3학년으로 구성된 대부분의 예비 교사들은 C, 비주얼 베이식, Java, 스크래치, 파이썬을 학습하였다고 답하였고, 가장 자신 있게 사용할 수 있는 언어로 C와 스크래치를 선택하였다.

[문항 2]의 설문 결과는 <그림 6>과 같다, 스크래치 언어를 중학교 교육과정에 가장 적합한 언어라고 응답하였다. 이는 스크래치 언어가 배우기 쉽고 흥미를 유발할 수 있는 그래픽 위주의 프로그래밍 언어이기 때문으로 분석된다.

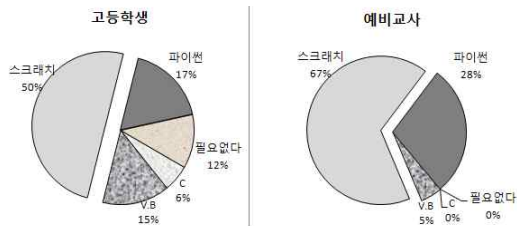


그림 6. 중학교 과정에 적합한 언어
Fig 6. Appropriate EPL for middle school curriculum

[문항 3]의 설문 결과는 <그림 7>과 같다.

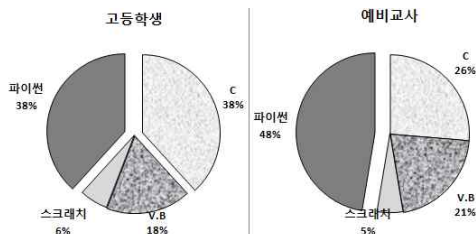


그림 7. 고등학교 과정에 적합한 언어
Fig 7. Appropriate EPL for high school curriculum

파이썬이 가장 적합한 언어로 응답한 이유는 고등학교 교육과정에 포함된 '함수와 배열의 활용'이 가능하면서도 C언어보다 배우기 쉬운 언어이기 때문으로 분석된다.

[문항 4]의 설문 결과는 <그림 8>과 같다, 고등학생의 50%, 예비대학생의 55%가 다른 언어를 배우는 것이 좋다고 응답하였다. 이는 학습 위계에 맞는 다양한 문제 해결 방법과 절차를 배울 수 있기 때문으로 분석된다. 따라서 문항2-4의 결과, 중학교에서는 스크래치, 고등학교에서는 파이썬 언어가 가장 적합한 언어인 것으로 분석되었다.

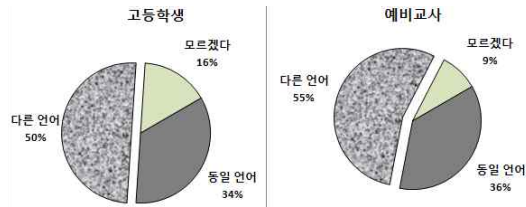


그림 8. 중·고 학습 언어의 동일성
Fig 8. Identity of EPL for middle and high school students

[문항 5]의 설문 결과는 <그림 9>와 같다.

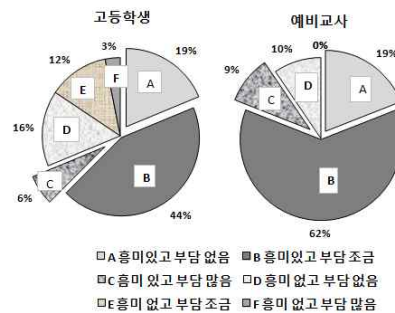


그림 9. 언어 학습에 대한 흥미와 부담감
Fig 9. Interest and burden in EPL learning

파이썬 언어를 배우는데 느끼는 흥미도와 부담감에 대해 고등학생의 44%, 예비 교사의 62%가 흥미는 있으나 배우는데 조금의 부담을 갖고 있다고 응답하였고, 고등학생과 예비 교사의 19%가 흥미 있고 부담이 없다고 응답하였다. 따라서 프로그래밍 언어를 배울 때, 흥미를 유지하면서도 부담감을 줄일 수 있는 교수법의 연구가 요구되고 있다.

[문항 6]의 설문 결과는 <그림 10>과 같다. 퍼즐 활용 프로그램 수업이 효과적이라고 한 이유는 다양한 논리적 사고력이 가능하기 때문으로 조사되었다. 따라서 퍼즐을 활용한 프로그래밍 교육이 상당한 효과가 있는 것으로 분석되었다.

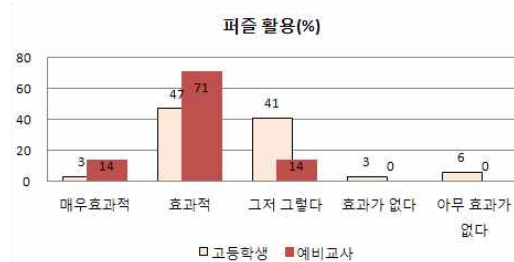


그림 10. 퍼즐활용의 수업 효과
Fig 10. Learning effect of puzzle using

[문항 7] 및 [문항 8]의 설문 결과는 <그림 11>과 같다. 새로 구성된 교육과정 내용의 난이도는 고등학생의 50%가 알맞다, 34%가 어렵다고 답하여, 난이도는 적절한 것으로 조사되었다. 또한 학습량의 적절성에 대한 설문 결과 55%가 약간 학습량이 적었다, 44%가 적절했다고 응답하였다. 하지만 이 결과는 창의적 체험 활동 컴퓨터부에서 활동한 흥미와 관심을 가진 학생들의 응답이므로 일반 학생들에게는 학습량이 적절하거나 조금 많을 것으로 추정된다.

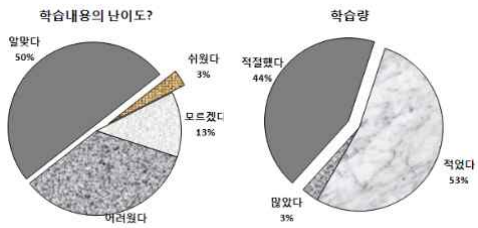


그림 11. 학습내용의 난이도와 학습량
Fig 11. Difficulty and quantity of contents

[문항 7-1]의 설문 결과는 <그림 12>와 같다.

본 연구에서 개발한 교육과정 수업 모형의 적절성에 대한 설문에서 예비 교사의 75% 이상이 적합하다고 응답하였다. 고등학생의 설문 [문항 7, 8]과 [문항 7-1]의 설문 결과를 종합해 볼 때, 본 수업 모형이 난이도, 학습량 적절성 면에서 적절한 것으로 나타나, 이 수업 모형이 2012년 일반계고등학교 정보 교과와 '문제 해결 방법과 절차' 영역의 수업 모형으로 적합한 것으로 분석되었다.

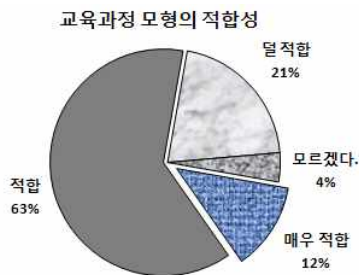


그림 12. 교육과정 모형의 적합성
Fig 12. Suitability of curriculum model

문항 [8-1]에서 시간을 늘려야 하는 차시와 내용 및 그 이유에 대한 내용은 <표 7>과 같다.

전반적으로 프로그래밍에서 개념 이해가 조금 어렵고 실습 시간이 필요한 부분인 '배열과 함수의 활용', '퍼즐 활용 프로그래밍', '정렬 알고리즘 구현'에 수업 시간이 조금 부족한 것으로 조사되었다.

표 7. 시간을 늘려야 하는 차시
Table 7. Expand period

차시 및 내용	이유
3. 알고리즘 설계 및 작성	학습 주제 어려움(14%)
12. 배열	학습 주제 어려움(14%)
14. 함수	학습 주제 어려움(24%) 실습 시간 확보(14%)
15. 퍼즐 활용 프로그래밍	실습 시간 확보(14%) 다양한 학습 방법 적용(19%)
16. 정렬 알고리즘 구현	학습 주제 어려움(19%) 학습할 양이 많아(14%)

[문항 9]와 [문항 10]의 설문 결과는 <그림 13>과 같다.

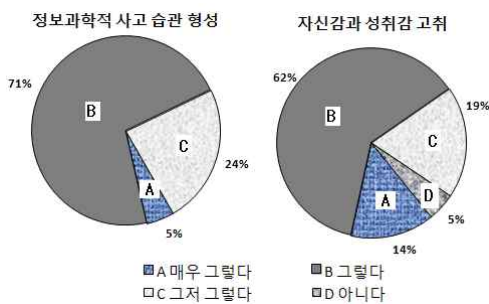


그림 13. 정보과학적 사고 습관 형성 및 자신감과 성취감 고취
Fig 13. Form practice of CT and inspiration of confidence and accomplishment

이러한 수업 모형으로 수업 한 후의 정보과학적 사고 형성과 자신감과 성취감 고취에 대해 똑 같이 76%가 '그렇다' 이상의 긍정적인 답변을 하였다. 이는 중 · 고등학생들이 '문제 해결 방법과 절차' 영역의 프로그래밍 교수 · 학습을 통해, 정보과학적 사고력으로 해결하는 습관이 형성되고, 자신감과 성취감이 높아질 수 있음을 의미한다. 따라서 프로그래밍을 통해 창의력을 향상시킬 수 있는 프로그래밍 교수 · 학습의 중요성을 확인할 수 있게 되었다.

3. 수업 적용 및 문제해결력 향상 검증

모의 수업 적용과 모의 수업 실시로 나타난 문제점을 개선한 후 2012년 3월 2일부터 부산시내 일반계고등학교 2학년 3개반(인문반) 정보 수업에 적용하였다. 2009 개정 교육과정에 따라 집중이수제를 실시함으로써 주당 4시간씩 6주간에 걸쳐 23차시를 실시하였다. 수업 적용 전, 후에 Heppner와 Petersen(1982)이 고안하고 김은희의 논문에서 사용된 문

제해결력 인식도 검사(Personal-Problem Solving Inventory, 신뢰도 $\alpha = .90$)를 하였다[25].

표 8. 수업 적용 전·후의 문제해결력 인식도
Table 8. PSI of before and after the classes

		평균	N	표준편차	t
대응	사전 점수	122.82	88	16.153	-4.025
	사후 점수	127.43	88	17.281	

사전, 사후 평균차이를 분석한 결과, 자유도 87에서 t값이 -4.025, 유의확률이 .000으로 $p < .05$ 이다. 그러므로 “유의수준 .05에서 신 개정 정보 교육과정의 ‘문제 해결 방법과 절차’ 영역을 위해 개발한 수업모형 적용은 문제해결력 변화에 차이가 있다.”라는 결론을 내릴 수 있다.

수업 적용 후 5월에 중간고사를 실시하였다. 중간고사에서 34문항 중 30문항을 ‘문제 해결 방법과 절차’ 영역에서 출제하였다. 중간고사 정보 시험은 총 응시 인원은 95명 이었고 평균 성적은 42.25점 이었다. 학생들의 PSI 사후 점수와 정보 성적을 분석한 결과를 <표 9>에 나타내었다. 이들의 상관관계는 Pearson 상관계수가 .247 이고 $p < .05$ 이기 때문에 통계적으로 유의미하지만 약간의 정 상관관계가 있다고 볼 수 있다. 문제 해결 능력 향상을 위한 정보 교과 내용이 처음 적용되었기 때문에 상관관계가 약하게 나온 것으로 분석된다.

표 9. PSI 점수와 정보 성적과의 상관계수
Table 9. correlation coefficient of PSI marks vs. Informatics marks

	중간고사
PSI사후	.247(+)

* $p < .05$

4. LSI와 정보 성적과의 상관관계 분석

Kolb의 학습 유형[26]에 따른 검사를 사전·사후에 실시하여 학습자 유형과 정보 성적과의 집단별 평균 분석을 한 결과를 <표 10>에 나타낸다.

사후 검사의 학습자 유형 중 융합자의 평균 점수가 전체 평균보다 약 10점이 높다. 융합자와 수렴자는 사람보다는 물리적 대상에 더 큰 관심을 보이며 자연과학 계통의 특성을 가진다. 반면에 적응자와 분산자는 사물보다 사람에 더 관심을 가지고 인문사회적인 특성을 가지는 경향이 있다.

연구자가 출제한 중간고사 문제는 87.6%가 알고리즘과

파이썬 프로그래밍 문제이고, 13%가 불 대수와 논리 연산이기 때문에 자연과학 계통의 특성을 가진 융합자의 평균 점수가 가장 높게 나오고, 수렴자의 점수도 높은 것으로 분석한다.

표 10. 학습 양식 별 정보 성적과의 평균 분석
Table 10. average analysis of LS vs. informatics record

LSI	평균	N	%	표준편차
적응자	41.71	47	53.3 %	16.70
분산자	39.47	11	12.2 %	12.57
융합자	53.85	12	13.3 %	12.26
수렴자	45.07	19	21.1 %	22.17
	43.77	90	100 %	17.37

<표 11>에서는 수업 적용 전, 수업 적용 후 LSI 검사에서 학습유형자의 교차 현황을 나타낸다. 사후에 가장 많은 학습 유형은 인문사회적인 특성을 가진 적응자가 53%로 나타났다. 본 연구에 참여한 학생들이 모두 인문반 학생들이기 때문으로 분석된다.

사전, 사후에 학습 유형이 변한 학생은 총 35%이다. 여기서 적응자의 31%가 수렴자로 변한 것은 ‘문제 해결 방법과 절차’ 영역이 자연과학 계통의 특성에 조금 밀접한 것이기 때문이라고 분석한다. 또한 분산자에서도 자연과학 계통의 특성을 가지는 수렴자로 16%가 변했다. 이것도 위와 같이 인문사회적인 특성인 분산자가 ‘문제 해결 방법과 절차’ 영역의 학습이 정보과학적 사고 능력의 향상을 가져왔다고 분석한다.

표 11. LSI 사전·사후 비교
Table 11. before and after LSI

		LSI 사후				전체
		적응자	분산자	융합자	수렴자	
L S I 사 전	적응자	77%	9%	8%	31%	49%
	분산자	13%	64%	8%	16%	19%
	융합자	4%	9%	67%	16%	16%
	수렴자	6%	18%	17%	37%	16%
		53%	12%	13%	21%	89%

V. 결론

2012년부터 실시되는 일반계 고등학교 ‘정보’ 교과와 ‘문제 해결 방법과 절차’ 영역을 효과적으로 교수·학습하여 정보과

학적 사고를 향상시키기 위한 수업 모형을 개발하였다. 서론에서 언급한 필요성에 따라 본 연구에서는 중학교 교육과정과 고등학교 교육과정을 연계한 '수업 모형'을 개발하였다. 일반계고등학교 학생들과 컴퓨터교육과 예비교사에게 수업모형을 적용하여 효과를 검증하였다.

수업 모형에 따라 2012년 3월부터 23차시 수업에 적용한 후 중간고사를 실시하였다. 수업 전, 후의 문제 해결능력 검사(PSI)를 실시한 결과, 자유도 87에서 t 값이 -4.025 , 유의확률이 $.000$ 으로 $p < .05$ 이다. 그러므로 "유의수준 $.05$ 에서 '일반계 고등학교 정보 교육과정'의 '문제 해결 방법과 절차' 영역을 위해 개발한 수업모형 적용은 문제 해결력 변화에 차이가 있다."라는 결론을 내릴 수 있다.

중간고사 필기시험 점수와 PSI 점수는 Pearson 상관계수가 $.247 > 0$ 이기 때문에 약한 정 상관관계가 있다고 볼 수 있다. 문제 해결 능력 향상을 위한 정보 교과 내용이 처음 적용되었기 때문에 상관관계가 약하게 나온 것으로 분석한다.

적용자의 31%, 분산자 16%가 수렴자로 변했다. '문제 해결 방법과 절차 영역'의 학습이 정보과학적 사고 능력의 향상을 가져왔다고 분석한다.

위의 결과로 본 연구에서 개발한 수업 모형이 2012년부터 적용되는 일반계 고등학교 정보 교과의 '문제 해결 방법과 절차' 영역의 교수·학습에 효과가 있음이 판명되었다.

신 개정 교육과정에서 새롭게 포함된 '문제 해결 방법과 절차' 영역의 프로그래밍 교육을 통한 문제 해결력 향상을 위하여 인지 발달에 적합한 EPL의 적용을 연구하여, 파이썬을 적용한 수업모형을 개발하고, 수업에 적용하여 효과가 있다는 것을 검증하였다는 점에서 시사하는 점이 있다고 하겠다.

'문제 해결 방법과 절차' 영역의 교수·학습 방법과 수업 모형이 부족한 정보 교사들의 어려움을 조금이나마 덜고 도움이 되고자 본 연구의 수업 모형을 제안한다. 또한 이 수업 모형은 예비 교사의 교육에도 활용될 수 있을 것이라고 제안한다.

참고문헌

- [1] Wing, J. M. "Computational thinking," *Communications of the ACM*, Vol. 49, No. 3, pp. 33-35, 2006.
- [2] Sook-Young Choi, "An Analysis of 'Informatics' Curriculum from the Perspective of 21st Century Skills and Computational Thinking," *Korea Association of Computer Education*, Vol. 14, No. 6, pp. 19-30, Nov. 2011.
- [3] Seungheon Lee and Kapsu Kim, "Development and Application of Teaching Model on Project-Based Programming for Elementary Students," *Korea Association of Computer Education*, Vol. 11, No. 2, pp. 23-33, Mar. 2008.
- [4] HakJin Bae, unKyoung Lee and oungJun Lee (2009). A Problem Based Teaching and Learning Model for Scratch Programming Education. *Korea Association of Computer Education*, Vol. 12, No. 3, pp. 11-22, May. 2009.
- [5] JeongBeom Song, SeoungHey Paik and TaeWuk Lee, The Effect of Robot Programming Learning considered Gender Differences on Female Middle School Student's Flow Level and Problem Solving Ability, *Korea Association of Computer Education*, Vol. 12, No. 1, pp. 45-55, Jan. 2009.
- [6] Kil-Mo Kim and Seong-Sik Kim, Development of a PBL-based Programming Instruction Model Using Collective Intelligence, *Korea Association of Computer Education*, Vol. 14, No. 2, pp. 23-32, Mar. 2011.
- [7] <http://ncic.kice.re.kr/ration.dwn.ogf.inventoryList.do#>
- [8] Jeoung, Mi Yeoun, "The Effects of Squeak Based Algorithm Learning on Student's Problem Solving Ability", a master's thesis, Korea National University of Education, 2008.
- [9] <http://home.comcast.net/~mrtwhs/mash/polya.pdf>
- [10] Kim, Hyun Joung, "A study on the direction of teaching algorithm", a master's thesis, Ewha Woman University, 2004.
- [11] Jenkins, Tony, *How to program using C++*, England: Palgrave Macmillan, 2003.
- [12] Seung Wook Yoo, "Application of EPL in informatics curriculum for K-12," a thesis for a doctorate, Korea Univ, 2008.
- [13] Younghak Jin, Min Huh and Yungsik Kim, Comparative Content Analysis of Middle School Informatics Textbooks and Suggestions for Improvement, *Korea Association of Computer Education*, Vol. 13, No. 3, pp. 25-34, May. 2010.
- [14] Joyce, B. & Weil, M, *Model of teaching*, New York: Prentice Hall, 1980.

[15] Jonghye Kim, Jyunghoon Kim and Wongyu Lee, "A study on the method of developing achievement and assessment standards for the 'Problem-solving methods and procedures' section in the revised Junior-high School Informatics curriculum," Korea Association of Computer Education, Vol. 11, No. 6, pp. 39-51, Nov. 2008.

[16] Park, JeongSun, "A Fundamental Study on Developing the Enrichment Curriculum for Gifted Children in Informatics Education Using Delphi Method -'Problem-solving method and its Procedure' in the Revised Version of the 7th National Middle School Curriculum," a master's thesis, Korea National University of Education, 2010.

[17] John M. Keller, Motivational Design for Learning and Performance - The ARCS Model Approach, Springer, 2010.

[18] Jae-Hyuk Choi, "Fun and Easy C Language with Puzzle," Honkrung Publishing Company, 2010.

[19] Jae-Hyuk Choi, "Python Programming for Creativity elevation," Silla Univ. Press, 2009.

[20] Si-guk Rou, Jae-Hyuk Choi, Young-Hyan Jo, "Puzzle and Creative Problem Solving ability," Naeha press, 2009.

[21] Jennifer Campbell, Paul Gries, Jason Montojo, Greg Wilson, Dae-yoep Lee translation, "Practical Programming," Acon Press, 2010.

[22] JeungMi Song, "Puzzle Education Based on Computational Thinking Using Scratch," a master's thesis, Korea Univ, 2011.

[23] M. Csikszentmihalyi, Flow: The psychology optimal experience, New York: Harper & Row, 1990.

[24] Tim Bell, Ian H. Witten, Mike Fellows, Boby Adams, Jane Mckenzie, "Computer Science Unplugged", http://csunplugged.com/sites/default/files/activity_pdfs_full/CS_Unplugged-en-10.2006.pdf

[25] Heppner, P.Paul. & Petersen, Chris.H, The development and implication of a personal

problem-solving inventory, Journal of counseling Psychology, Vol.9, No.3, pp. 66-75, Jan. 1982.

[26] D. A. Kolb. Experimental Learning: Experience as the Source of Learning and Development, Englewood Cliffs, NJ : Prentice-Hall, 1984.

저 자 소 개



현 태 익

1982: 부산대학교 사범대학
외국어 전공 교육학사.
1998: 신라대학교 교육대학원
컴퓨터교육과 교육학석사.
2002: 신라대학교 일반대학원
컴퓨터정보공학과 박사 수료
1982 ~ 현 재: 부산동고등학교 교사
신라대학교
컴퓨터교육과 겸임교수
관심분야: 컴퓨터교육
Email : e1009@silla.ac.kr



최 재 혁

1984: 경북대학교 전자공학과
컴퓨터공학 전공 공학사.
1986: 경북대학교 전자공학과
컴퓨터공학 전공 공학석사.
1994: 경북대학교 컴퓨터공학과
공학박사
1989 ~ 현 재: 신라대학교
컴퓨터교육과 교수
관심분야: 컴퓨터교육,
한국어정보처리 등
Email : jhchoi@silla.ac.kr



이 종 희

1978: 경북대학교 전자공학과
전산공학 전공 공학사
1984: 경북대학교 전자공학과
전산공학 전공 공학석사
1990: 경북대학교 전자공학과
전산공학 전공 공학박사
1988 ~ 현 재: 신라대학교
컴퓨터정보공학부 교수
관심분야: 컴퓨터공학
Email : jhlee@silla.ac.kr

