

## 객체지향형 교육용 프로그래밍 언어 '두리틀(Dolittle)'의 수학 교수-학습 활용 방안<sup>1)</sup>

황 우 형 (고려대학교)

김 경 미 (고려대학교 대학원)

본 논문은 최근 일본에서 개발된 객체지향형 교육용 프로그래밍 언어(EPL: Educational Programming Language) '두리틀(Dolittle)'을 소개하고, 두리틀을 수학 교수-학습에 활용하기 위한 최초의 연구이다. 두리틀은 LOGO의 거북 그래픽스(Turtle Graphics)와 인크리멘탈(Incremental) 프로그래밍 방식, 즉각적인 피드백 등 많은 교육적 이점을 수용하고, 현대 프로그래밍의 고급 기능들을 프로토타입(Prototype) 방식을 통해 어린 학생들도 쉽게 이해할 수 있게 한 텍스트기반의 한글 교육용 프로그래밍 언어이다. 본 논문에서는 LOGO와 두리틀의 활용 비교를 통하여 두리틀을 소개하고, 두리틀을 이용한 기하와 함수의 교수-학습을 위하여 연구자가 고안한 기하판과 좌표판, 삼각함수판의 활용 방안을 제안하고, 그 교수-학습에서 발생하는 문제점을 해결하기 위한 몇 가지 아이디어를 제안하고자 한다.

### I. 서 론

#### 1. 연구의 배경 및 목적

수학교육에 있어서 컴퓨터는 수학학습의 동기유발에서부터 추상적이고 형식적인 수학적 대상을 구체적으로 탐구할 수 있는 환경을 제공하는데 큰 역할을 하였다. Zimmerman과 Cunningham(1991)은 컴퓨터가 제공하는 시각적 이미지가 수학적인 모델과 실제 세계의 현상 사이의 중요한 연결고리의 역할을 한다고 주장하였다. 미국교사협의회(NCTM, 1989, 2000)는 1980년대부터 수학교육과정에 계산기와 컴퓨터 같은 테크놀로지의 활용을 적극 권장하였으며, 컴퓨터를 이용한 학생들의 문제 해결 능력을 강조하였다.

현대사회는 정보화 사회로서 머리 속의 단편적인 지식의 저장 양보다는 창의적인 사고와 합리적인 논리적인 판단과 문제 해결 능력을 요구하며, 최근의 수학교육은 구성주의 패러다임에 입각한 교수-학습 과정을 지향하고 있다. 이는 구체적인 조작 활동을 통하여 학생 개개인이 가능한 한 스스로 수학적 지식을 구성할 수 있어야 함을 의미한다. Freudenthal은 수학 학습 과정에서 학생들이 교사의 안내 하에 학습자 자신의 감정이 이입될 수 있는 현실에서 수학화 활동을 통해 수학적 내용을 재발명해 나가야 한다고 주장하는데 이는 현재 학교 환경으로는 많은 제약사항이 수반된다. 그러나

1) 본 연구는 2004년 한국학술진흥재단의 지원(KRF-2004-030-B00051)으로 수행되었음.

컴퓨터의 발달로 인하여 현실(Reality)의 개념도 크게 변하여 그 역할을 컴퓨터가 대신 해줄 수 있게 되었다(조한혁, 2003). 구체적으로 컴퓨터를 통해 실현된 LOGO 마이크로월드(Microworld)나 동적 기하 마이크로월드(Dynamic Geometry Microworld)는 제한된 학교 환경에서 학생들에게 좋은 탐구 환경을 제공한다. 특히 1960년대 후반 미국 MIT 대학의 인공지능(AI) 연구팀의 S. Papert가 개발한 LOGO 마이크로월드(Microworld)는 실세계의 단순화된 모형으로 학생들이 그 안에서 자기 수준에 맞춰 조작하고 탐구하는 과정을 통해 수학적 개념과 원리들을 이끌어 낼 수 있는 수학적 가상현실(Virtual Reality)을 말한다. Papert(1980)는 LOGO 마이크로월드(Microworld)라는 가상 세계에 어린 학생들에게 친숙한 거북이를 사용하여 기하의 직관적 경험 학습을 제공하였고, 고도의 인지과정을 요하는 재귀(recursion)과정을 통하여 어린 학생들에게 무한이라는 개념과 프랙탈과 같은 수학의 아름다움을 경험할 수 있는 기회를 제공하였다. LOGO 프로그래밍 활동이 아이들의 반영적 사고와 문제 해결 능력을 향상시키는 데 큰 도움이 된다는 것은 많은 선행연구를 통해 검증된 사실이다(Blackwelder, 1986; Clements, 1991; Cathart, 1990; Many, 1988; Schuyten & Valcke, 1990; Kern & Mauk, 1990; Swan & Black, 1990; Grandgenett, 1991; Nevile, 1992; 황우형, 1999 재인용).

그러나 최근 LOGO 프로그래밍 활동의 경험이 일반적 상황에서의 문제 해결로 전이되는 것에 대해서는 논의가 계속되고 있으며(Tetenbaum & Mulkeen, 1984; Metterer & Krasnor, 1986; Dalbey & Kinn, 1986; 백영균, 1988 재인용), LOGO 언어구조가 LISP의 개념을 계승해서 설계되었기 때문에 학습자가 조금이라도 복잡한 프로그램을 기술하려고 하면 재귀(recursion)와 리스트(list)같은 고도의 인지 개념을 학습해야 한다는 단점을 가지고 있다(兼宗進, 2003; 兼宗進 외 4명, 2001; 中谷 多哉子 외 4명, 2002; 兼宗進 외 4명, 2003; Ursula Wolz & Edward Conjura, 1994; Judith Gal-Ezer & David Harel, 1999). Bork(1989)는 LOGO 언어가 객체지향 언어가 아닌 낡은 언어구조를 가지고 있고, 거북기하 이상의 학습으로 진행되는 것이 어렵기 때문에 미래의 학교 교육용 언어로 적합하지 않다고 하였다(兼宗進, 2003). 그리고 LOGO 환경에서는 복수의 객체나 GUI 함수를 사용하도록 확장한 경우 프로그램이 복잡하게 되는 경향 등 LOGO의 교육적 의의를 높이 평가하는 반면, 구시대적인 언어 구조가 갖는 단점들을 지적하였다.

이에 근거해 일본의 가네무네(2003)는 현대 프로그래밍 언어의 경향인 객체지향(Object Oriented)에 프로토타입(Prototype) 방식과 거북 그래픽스(Turtle Graphics)를 도입하여 어린 아이들도 쉽게 프로그래밍 할 수 있는 교육용 프로그래밍 언어 '두리틀(Dolittle)'을 개발하였다. 두리틀은 텍스트 기반의 프로그래밍 환경이며 LOGO의 거북 그래픽스(Turtle Graphics)를 수용하여 고안한 것으로 LOGO의 교육적 이점을 모두 취하면서 현대 프로그래밍 언어의 다양한 기능을 모두 지니고 있는 언어이다. 기존의 교육용 프로그래밍 언어 LOGO가 수학의 기하영역에 중점적으로 적용 가능했던 것에 반해, 두리틀은 기하, 함수, 대수 등 수학 활용 영역이 LOGO보다 광범위하고, 다양한 GUI(Graphical User Interface) 기능으로 학생들의 흥미와 호기심을 유발시킬 수 있다. 또한 다중 명령어 방식의 한글 프로그래밍 언어로 어린 초등학생들도 쉽게 프로그래밍할 수 있다는 장점을 지니고 있다.

본 논문은 최근 일본에서 개발된 교육용 프로그래밍 언어 두리틀을 소개하고, 수학 교수-학습에 활용할 수 있는 방안을 모색하고자 한다.

## 2. 용어 정의

### 1) 객체지향 프로그래밍(Object-Oriented Programming)

객체(Object)는 실체(데이터)와 그 실체와 관련된 동작(절차, 특성, 기능) 등이 포함된 개념을 말한다. 객체지향 프로그래밍(Object-Oriented Programming)은 다양한 문제를 해결하고자 할 때, 객체를 정의하고, 생성된 객체를 중심으로 문제를 해결해 나가는 프로그래밍 방식이다.

기존의 절차적 프로그래밍은 동작과 절차를 중심으로 하고, 실체는 종속적으로 취급하여 현실 세계의 문제를 프로그램으로 표현하는 것이 곤란한 반면, 객체지향 프로그래밍은 실체와 동작을 객체로서 정의하고 객체 간의 메시지 교환에 주안점을 두기 때문에 프로그램이 단순화되고 생산성과 신뢰성이 높은 시스템을 구축할 수 있다. 또한 객체지향 프로그래밍 방식은 비교적 인간의 사고방식과 유사한 방식으로 프로그래밍하기 때문에 프로그램 작성이 쉽다는 교육적 이점을 가진다.

## II. 이론적 배경

### 1. 프로그래밍과 수학교육

컴퓨터과학교육 표준에 관한 연구를 진행하고 있는 ACM(Association of Computing Machinery, 2003)에 의하면, 컴퓨터과학(CS: Computer Science)은 컴퓨터와 알고리즘적 처리과정에 대한 연구로서 이 분야에 대한 원리와 하드웨어, 소프트웨어의 설계 및 응용 소프트웨어에 대한 연구와 더불어 컴퓨터가 사회에 미치는 영향까지도 포함하는 전문적인 학문으로 일반적인 수준에서 보급할 필요가 있다고 하였다(ACM, 2003). 현대 사회의 급속한 변화로 인하여 세계적으로 컴퓨터과학교육이 중요해지고 있으며, 해외에서는 컴퓨터과학교육을 위해 수학과 프로그래밍이 통합된 교과과정 설계 모델에 관한 연구 및 프로그래밍 교육 모델에 관한 연구가 활발히 이루어지고 있다(Ursula Wolz & Edward Conjura, 1994; Joy C. Kataoka & James R. Patton, 1996; Vicki L. Almstrum et. al., 2002; Kris Howell, 2003). 최근 많은 사람들이 컴퓨터과학교육과 수학교육을 위해 프로그래밍 학습의 지도를 강조하고 있다(ACM, 1991; Ursula Wolz, Edward Conjura, 1994; Mibrandt, G. 1995; Vicki L. Almstrum et. al., 2002). 그 이유로 컴퓨터 과학과 수학교육은 서로 상보적인 관계이고, 프로그래밍 활동이 알고리즘적 사고와 문제해결력을 신장시킨다는 점에서 컴퓨터과학교육과 수학교육에 모두 유용한 활동이라는 것이다.

그러나 국내 수학교육에서는 컴퓨터와 인터넷의 보급이 보편화되면서 여러 가지 수학 소프트웨어의 개발과 인터넷 활용을 통한 교수-학습 방법에 대한 연구는 증가하고 있지만, 프로그래밍 활동에 관한

연구(류희찬, 1994; 백영균·우인상, 1994; 황우형, 1999; 조한혁, 2003)는 매우 극소수이며, 컴퓨터 환경에서의 수학교육에 대한 전문적인 교수-학습 방법 및 교재 개발에 관한 연구 또한 매우 미흡한 실정이다. 수학교육 선진국에서는 오래 전부터 수학교육과정에 계산기와 컴퓨터 같은 테크놀로지의 활용을 적극 권장하였으며 LOGO와 같은 프로그래밍 활동에 관한 연구를 통해 프로그래밍 활동이 학생들의 수학적 사고력과 문제해결력, 메타인지 등 중요한 인지적 능력을 향상시킨다는 결과를 얻어내었다(Papert, 1980; George Lukas & Joan Lukas, 1986; Ursula Wolz & Edward Conjura, 1994; Judith Gal-Ezer & David Harel, 1999; NCTM, 2000). 그러나 국내에서는 LOGO 프로그래밍 활동의 많은 수학교육적 이점에도 불구하고 실제 학교 현장에 LOGO 프로그래밍 활동이 적용되지 못하였다. 그 이유는 LOGO 언어가 개발된 시기가 1980년대 초이므로 그 당시 국내에서는 학교에 컴퓨터 학습 환경을 제공해 줄 만큼의 재정적 여건이 마련되지 못하였고, 컴퓨터가 보편화된 지금에 와서는 LOGO 언어가 현대의 응용 소프트웨어 시스템에 부합되지 않으며, 다양하고 흥미로운 게임에 접해 있는 현대 아이들의 흥미를 끌기에는 역부족이기 때문이다. 따라서 현재 국내 수학교육에는 LOGO와 같이 쉽고 간단한 프로그래밍 과정을 통해 학생들이 수학적 개념과 문제해결력을 신장시킬 수 있는 프로그래밍 언어의 개발이 시급하며, 이에 따른 교수-학습 방법 및 교재 개발이 이루어져야 한다.

## 2. 교육용 프로그래밍 언어의 조건

프로그래밍 지도의 진정한 목표는 특정한 컴퓨터 언어의 문법 학습이 아니라 알고리즘의 이해와 표현 그리고 실행이다(조한혁, 1991). 특히 수학교육에서 프로그래밍 활동의 목적은 프로그래밍 언어의 형식적인 면보다는 활동에 내재된 수학적 내용에 초점을 두어야 한다.

다음은 선행 연구를 통해 교육용 프로그래밍 언어의 조건(조한혁, 1991; 이종영, 1994; 兼宗進, 2003; 兼宗進 외 4명, 2001; 中谷 多哉子 외 4명, 2002; 兼宗進 외 4명, 2003)과 수학교육용 프로그래밍 언어의 조건(박춘란, 1991)으로 제시된 내용을 기초로 정리한 것이다(김경미, 2004). 프로그래밍 활동을 수학 교수-학습에 적용하기 위해서는 교육용 프로그래밍 언어의 조건과 더불어 수학교육용 프로그래밍 언어의 조건을 모두 만족시키는 언어를 선택하여야 한다.

<표 1> 교육용 프로그래밍 언어와 수학교육용 프로그래밍 언어의 조건

| 교육용 프로그래밍 언어의 조건         | 수학교육용 프로그래밍 언어의 조건           |
|--------------------------|------------------------------|
| 간결하고 프로그래밍 작성이 용이해야 함.   | 다중 명령어 방식의 한글 프로그래밍 언어이어야 함. |
| 확장성이 있는 알고리즘 언어이어야 함.    | 함수 표현이 용이해야 함.               |
| 기본적인 컴퓨터원리를 학습할 수 있어야 함. | 그래픽 표현이 용이해야 함.              |
| 호환성과 전이성이 있어야 함.         | 학교 수학의 내용과 부합되어야 함.          |
| 친근감을 주는 대화용 언어이어야 함.     | 다양한 방법으로 문제를 해결할 수 있어야 함.    |
| 네트워크를 체험할 수 있어야 함.       | 이해하기 쉽고, 언어의 습득시간이 짧아야 함.    |

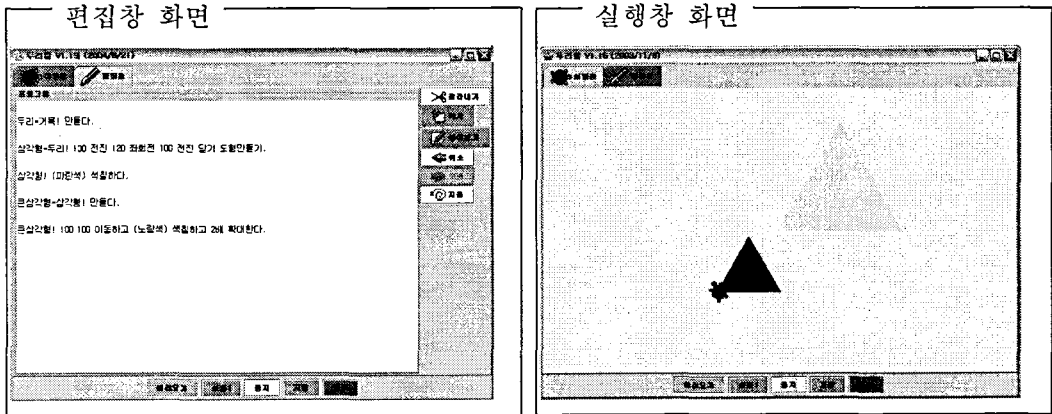
본 연구에서 소개하는 두리틀은 다중 명령어 방식의 한글 프로그래밍 언어로 동일한 의미의 다중 명령어 집합에서 학습자가 자신의 인지 수준에 맞는 명령어를 선택하여 사용할 수 있으며, 한글 어순의 간결한 문장으로 표현되기 때문에 어린 초등학생들도 쉽게 프로그래밍 할 수 있는 언어이다. 또한 기존의 타 언어에 비하여 함수적 표현이 용이하며, 거북 그래픽스(turtle graphics)환경으로 그래픽 표현도 용이하다. 두리틀은 컴퓨터교육을 목적으로 개발된 언어이지만 많은 수학적 요소를 포함하고 있으며, 두리틀 언어의 수식 표현이 수학에서의 수식 표현과 거의 유사하므로 변수와 함수의 개념학습에도 큰 도움을 줄 수 있다. 두리틀이 컴퓨터 환경과 프로그래밍 언어로서 가지는 수학교육적 한계는 타 언어와 마찬가지로 수반하고 있지만, 교육용 프로그래밍 언어의 조건과 수학교육용 프로그래밍 언어의 조건에 모두 부합되는 언어로 수학 교수-학습의 적용에 긍정적인 효과가 기대된다.

### 3. 두리틀 소개

두리틀(Dolittle)은 2001년 일본의 가네무네가 컴퓨터 교육을 위한 목적으로 개발한 객체지향형 교육용 프로그래밍 언어(OOEPL: Object-Oriented Educational Programming Language)이다. 두리틀은 어린 학생들도 쉽게 이해할 수 있는 텍스트 기반의 교육용 프로그래밍 언어이며, 언어 설계에 있어 LOGO에서 거북 그래픽스(turtle graphics)의 아이디어와 인크리멘탈(incremental) 프로그래밍 방식, 즉각적인 피드백 등 많은 교육적 아이디어를 수용하였다. 두리틀은 LOGO의 교육적 이점을 수용하면서 최근 프로그래밍 경향인 객체지향 패러다임에 부합되는 언어를 고안하기 위하여 Self 언어의 프로토타입(Prototype) 방식을 수용하여 어린 학생들이 어려운 객체지향 개념(클래스, 상속, 인스턴스 등)을 모르고도 쉽게 객체지향 프로그래밍을 체험할 수 있도록 고안한 언어이다.

2001년 11월 13일에 일본판 두리틀 정식 버전을 발표하였으며, 2003년 6월에는 초기 두리틀 1.0 버전의 인터페이스 및 명령어를 업그레이드하여 1.13 버전을 발표하였다. 그리고 2003년 8월부터 우리나라에서 처음으로 명령어 한글화 작업을 진행하였으며, 2003년 11월 8일 두리틀 한글 1.16 버전을 공식발표하였다. 그 후 두리틀 명령어 한글화 연구(최해심 외, 2004) 및 컴퓨터교육과 수학교육의 실제 적용을 위한 연구(권대용 외, 2004a; 권대용 외, 2004b; 길혜민 외, 2004; 최은조 외, 2004; Susumu Kanemune et. al., 2004; Shuji Kurebayasi et.al., 2004; 김경미, 2004; 장혜선 외, 2004; 박소영 외, 2004)가 한일 공동 연구로 활발하게 진행되었다. 그 결과 최근 2004년 8월 21일 발표된 1.19j 버전에서는 일본의 웹 서버로의 자료 업·로딩이 가능하고, 일본에서 작성한 소스코드가 한국 소스코드로 자동 변환되기 때문에 일본 학생과 한국 학생 간의 실시간 협동 프로젝트가 가능하며, 멜로디 기능도 첨가되어 학생들에게 컴퓨터, 수학뿐만 아니라 다양한 영역의 탐구 학습을 체험할 수 있는 교육용 프로그램으로 개발되고 있다.

두리틀의 실행 화면은 <그림 1>과 같이 편집창과 실행창으로 구성되며, 우측의 편집버튼과 하단의 명령버튼으로 구성되어 있다.

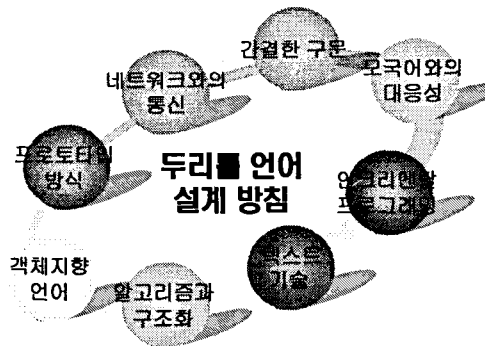


<그림 1> 두리틀 편집창과 실행창

### III. 객체지향형 교육용 프로그래밍 언어 '두리틀(Dolittle)'의 수학 교수-학습 활용 방안

#### 1. 두리틀 언어의 특징

兼宗進 외(2001)는 현대 컴퓨터 언어의 경향에 알맞은 교육용 프로그래밍 언어가 갖추어야 할 특징에 대하여 다음과 같이 제시하고 이 기준에 맞추어 객체지향형 교육용 프로그래밍 언어 두리틀을 설계하였다.



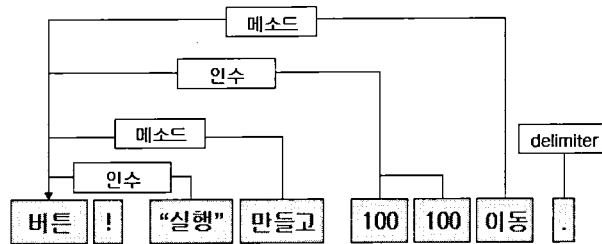
<그림 2> 두리틀 언어의 특징

두리틀의 가장 큰 특징은 상속이나 클래스와 같은 고도의 추상적인 개념 이해가 필요치 않는 범위 내에서 객체지향 개념을 도입하고 있다는 것이다. 객체를 복사하여 사용함으로써 원래 객체의 성

질을 계승하는 방식을 이용하고 있다. 두리틀 언어 설계 방침을 통한 두리틀 언어의 세부적인 특징은 다음과 같다(권대용 외, 2004a).

첫째, 간결한 구문이다.

계층적인 구문을 피하고 이해하기 쉬운 구문으로 되어있다. 일반 프로그래밍 언어와 달리 예약어가 없어서 변수 사용에 어려움이 없으며 메소드 호출을 “!”를 사용하여 <그림 3>과 같이 연속적으로 사용할 수 있다.



<그림 3> 두리틀 구문 형태

둘째, 한국어로서의 이점을 가진다.

영어의 예약어 등이 전혀 없고 기본적인 기호(괄호, 콤마, 마침표 등)를 포함한 한국어와 한국어의 어순으로 프로그래밍 된다. 또한 두리틀은 다국어를 지원하며 소스코드의 한국어와 일본어는 프로그램에서 자동으로 변환이 가능하다.

셋째, 인크리멘탈 프로그래밍 방식이다.

종래의 Basic과 LOGO가 교육용 언어로 사용되어져 왔던 요인의 하나는 인터프리터 방식으로 코딩 결과를 바로 확인할 수 있다는 점이었다. 두리틀도 이 이점을 계승해서 한 행만으로 그 나름대로의 동작을 기술할 수 있으며, 이것을 그대로 메소드로 정의할 수 있다.

넷째, 텍스트 기술의 프로그래밍 방식이다.

프로그래밍 및 문제해결 과정에서 이루어지는 고도의 논리적 표현을 위해서 두리틀은 텍스트 입력력을 통해서 소스코드를 기술하고 있다. 더 나아가서 두리틀은 직접 코드 작성이 어려운 초등학생들의 프로그래밍 교육을 위해서 비주얼 형태의 코딩을 할 수 있는 연구도 병행하여 진행되고 있다(장혜선 외, 2004).

다섯째, 구조화된 알고리즘을 기술할 수 있다.

수치, 문자열, 배열 등의 기본 데이터를 취급하고 반복과 조건 분기 등의 제어구조를 갖추어 절차에 상응하는 메소드를 정의하는 것으로 구조화된 알고리즘을 기술할 수 있다. 이것을 통하여 계산기의 기초 개념을 체험하는 것이 가능하다.

여섯째, 객체지향형 프로그래밍 언어이다.

두리틀은 현대의 소프트웨어 개발 방법론의 핵심 주제인 객체지향 방식을 통해 프로그래밍을 할

수 있다. 학습자가 성취감을 느끼게 하기 위해서는 이와 같은 객체지향 개념이 적용된 교육용 언어가 필요하다. 현재 객체를 쉽게 파악하여 프로그램 구조 파악에 도움을 줄 수 있는 연구도 이루어지고 있다(박소영 외, 2004).

일곱째, 프로토타입 방식이다.

두리틀은 직접 클래스를 생성할 수는 없으며, 제공되는 프로토타입의 객체를 사용하여 프로그래밍을 하게 된다. 이와 같은 프로토타입 객체(예: 거북, 버튼, 타이머)를 이용한 학습은 학습자로 하여금 정의된 객체를 사용하여 문제를 해결하게 하고, 불필요하고 어려운 프로그래밍 기술을 요구하지 않아 교과 내용 또는 학습 주제에 집중시킬 수 있다는 점에서 교육적 효과를 가진다.

여덟째, 네트워크와의 통신이 가능하다.

인터넷과 전자메일의 보급에 의해 컴퓨터의 활용이 네트워크와 떨어질 수 없다. 두리틀은 네트워크를 통한 데이터 교환 및 공유 프로그래밍을 경험할 수 있다.

## 2. 두리틀 적용 사례

가네무네(2003)는 두리틀 언어를 개발하고, 중학교 2학년 132명을 대상으로 2학기에 걸쳐 11시간 수업을 실시하였다. 연구 결과, 개념 이해율이 86.1%와 74.3%로 나왔으며 난이도가 올라감에도 불구하고 흥미도와 달성도가 감소하지 않고 증가하였다. 가네무네 외(2004)는 최근 교사 교육과 기업 교육에 두리틀을 적용하고 있으며, 중등 컴퓨터과학교육을 위한 방안으로 프로그래밍 학습의 중요성을 강조하고 꾸준히 두리틀 언어를 수정·보완하고 있다.

이 밖에도 일본에서는 학생들을 대상으로 소그룹별 미니 로봇 콘테스트를 개최하여 학생들이 화면상의 그래픽 프로그램을 실제 두리틀 로봇에 전송하여 동작하게 하고, 문제점을 해결하는 과정에서 로봇 제어 뿐 아니라 문제 해결력과 협동심을 기를 수 있는 기회를 제공하고 있다. <그림 4>는 일본에서 개최된 미니 로봇 콘테스트에서 학생들이 로봇이 통과할 다리를 설계하고 실험하는 장면이다(가네무네, 2003).



图 14: 콘테스트の様子

<그림 4> 일본의 미니 로봇 콘테스트



그러나 일본에서는 두리틀의 수학 교수-학습을 위한 연구는 진행되지 못한 상태이므로, 현재 우리나라에서 수학 교수-학습을 위한 교육적 아이디어를 제시하고 일본에서는 그에 따른 기술 지원을 하고 있다.

우리나라에서 두리틀을 적용한 사례는 아직 미흡하지만 모두 좋은 교육적 효과를 얻었다. 길혜민 외(2004)는 중학교 1학년 학생 26명을 대상으로 6차시의 두리틀 수업을 한 결과 버튼과 애니메이션 효과 등으로 그림판 같이 간단한 소프트웨어 작성이 용이하여 학생들의 흥미도와 관심이 매우 높게 나타났으며, 학업성취도와 차후 학습으로의 연계성이 우수하다는 교육적 효과를 검증하였다.

최은조 외(2004)는 정보화 교육으로부터 소외되어있는 성인들을 대상으로 여러 가지 신체적, 정신적인 특성상의 장애요인을 극복한 프로그래밍 교육의 적용 가능성을 알아보기로 성인 여성 30명을 대상으로 총 3차시의 두리틀 수업을 진행하였다. 연구 결과, 참여자들은 수업시간 내내 높은 흥미도와 결과에 대한 만족도를 유지하였으며 컴퓨터에 대한 자신감이 향상하였다.

수학교육에서의 발표된 활용 사례는 아직 없지만, 현재 두리틀을 이용한 수학 교수-학습 적용 사례 연구가 진행되고 있다.

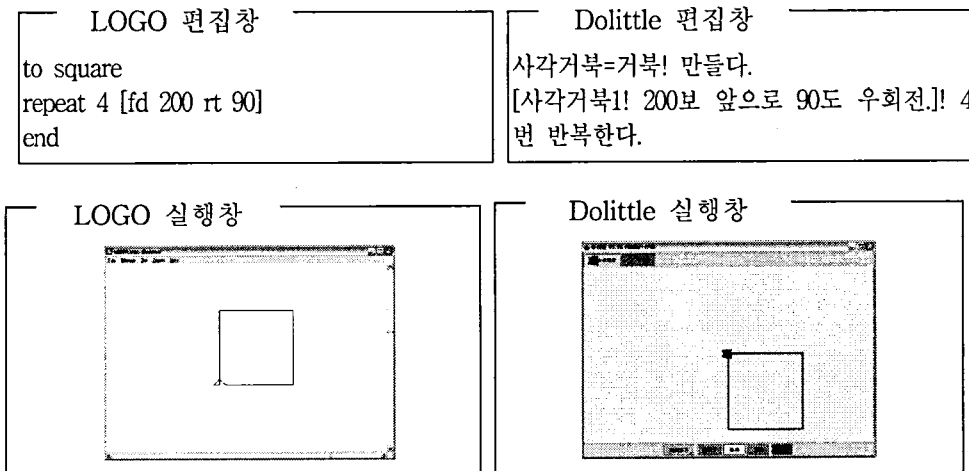
### 3. LOGO와 두리틀의 비교

두리틀은 오늘날 소프트웨어의 기술에 있어서 중요한 위치에 있는 객체지향의 사고방식에 기반한 교육용 프로그래밍 언어이다. 종래 교육현장에서 사용되어 왔던 Basic과 LOGO가 플로우차트로 대표되는 '리스트(List)'라는 '계산기의 원리'를 중심으로 프로그램을 기술하는 것과는 다르게 두리틀은 동물과 버튼 등 실세계의 '사물'에 해당하는 '객체(Object)'를 단위로 한 프로그램을 기술하는데 이것은 비교적 인간과 가까운 사고방식으로 프로그램을 한다는 점에서 현대 컴퓨터 언어에서 지향하는 패러다임에 부합되는 것이다(가네무네, 2003). LOGO에서는 거북 객체 하나만을 사용하여 도형을 그리는 반면, 두리틀은 여러 거북 객체를 생성하여 각 객체에게 역할을 분담하여 그리거나 혹은 어떤 기능을 수행하게 할 수 있으며, 거북 객체 뿐 아니라 버튼, 텍스트 필드, 타이머, 리스트, 배열, 라벨, 선택 메뉴, 시리얼 포트, 멜로디 등 다양한 객체를 통해 기능적으로나 시각적으로 다양한 프로그램 작성이 가능하다. 특히 타이머(시계) 객체는 여러 개의 객체를 정확한 수학 계산에 의하여 동작하게 하여 객체들 간의 애니메이션 구현이 가능하며 이 밖에도 수학적 요소가 많이 내포되어 있는 객체이다.

두리틀은 최근 개발된 교육용 프로그래밍 언어로 아직 수학 교수-학습에 활용되기 위한 방안 및 적용 사례가 없기 때문에 본 절에서는 LOGO와 두리틀을 실제 수학 교수-학습에 활용할 수 있는 예제들의 비교 분석을 통하여 두리틀의 프로그래밍 방법을 소개하고, 수학 교수-학습에 두리틀의 활용 가능성을 알아보려 한다.

### 1) 도형 그리기

다음은 LOGO와 두리틀에서 정사각형을 그리는 방법이다. LOGO와 두리틀 모두 거북 그래픽스(Turtle Graphics)를 이용한 프로그램으로 간단한 명령어만을 사용하여 도형을 그릴 수 있다. LOGO와 두리틀은 처음 시작 위치는 같지만 LOGO의 거북이 머리방향은 위를 향하고 있는 반면, 두리틀의 거북이 머리방향은 오른쪽을 향하고 있다. <그림 5>는 정사각형의 예로 두 언어 모두 반복명령어("repeat", "반복한다.")를 사용하여 불필요한 명령어 기술을 요구하지 않는다는 장점을 가지고 있다.



<그림 5> LOGO와 두리틀의 정사각형 그리기

### 2) 변수의 활용

다음은 변수를 사용하여 정사각형을 그리는 방법이다. LOGO에서는 지역변수를 「:X」 형태로 표현하며 전역변수를 「X」 형태로 표현함으로써 변수를 구분하는 반면, 두리틀에서는 「X」의 표현으로 전역변수와 지역변수의 구분을 구문상의 위치로 구별하고 있다. 두리틀은 학교 수학에서 사용하는 변수 형태와 동일하게 사용한다. 다른 점이 있다면, 변수의 값을 참조하고 싶을 때에는 (X)처럼 괄호를 묶어 표현한다. 다음은 정사각형의 한 변의 길이를 X라는 변수로 받아 그려주는 예제이다. 프로그램 작성시 LOGO의 경우 편집창과 실행창 이 외에도 COMMAND창이 따로 있어 아래와 같이 함수 호출을 따로 기술하며, 두리틀은 함수 선언과 함수 호출을 모두 편집창에 기술한다. 어떤 방법이 학생들에게 좀 더 쉽게 인지되며, 학습 효과가 있는지는 추후 연구를 통해 알아보아야 할 것이다.

| LOGO 편집창  |               | Dolittle 편집창   |
|---|---------------|--|
| <pre>to square :side :angle repeat 4 [fd :side rt :angle] end</pre> |               | 거북1=거북! 만든다.<br>거북1: 사각형=[변의길이 각도<br>[거북1! (변의길이) 전진 (각도) 우회전!] 4번 반복한다.].<br>거북1! 100 90 사각형. |
| COMMAND   | square 100 90 |  |

<그림 6> LOGO와 두리틀의 변수를 사용한 정사각형 그리기

<그림 6>은 변의 길이와 각을 변수로 만들어 원하는 변의 길이와 각을 가진 사각형을 그리는 예제이다. 두리틀에서는 변수의 성질을 이해하기 쉽게 변수의 이름을 수학용어로 표현할 수도 있다. 이런 변수의 표현은 한글 프로그래밍 언어만이 가질 수 있는 이점이며, 어린 학생들에게는 쉽게 프로그램을 이해할 수 있게 한다는 점에서 두리틀의 큰 장점이다.

### 3) 재귀에 대한 고찰

LOGO는 문제를 해결함에 있어 재귀(Recursion)의 개념을 가장 많이 사용하는 언어이다. 그러나 최근 재귀의 개념이 어린 학생들에게는 너무 어렵고 추상적인 개념이기 때문에 프로그래밍 입문 과정에서 재귀 개념을 가르치지 않아야 한다는 의견이 높아지고 있다(兼宗進, 2003; 兼宗進 외 4명, 2001; 中谷 多哉子 외 4명, 2002; 兼宗進 외 4명, 2003; Ursula Wolz & Edward Conjura, 1994; Judith Gal-Ezer & David Harel, 1999). 재귀의 개념은 어린 학생들이 이해하기에는 너무 구조적인 개념으로 처음 학생들이 문제를 재귀를 사용하여 접근하도록 하는 방법보다는 반복 명령어를 사용하여 문제를 해결한 후 좀 더 간결하고 세련화시키기 위하여 재귀를 사용하는 과정이 필요하다. 그러나 현재 LOGO의 활용 사례 대부분이 간단한 문제도 어려운 개념의 재귀를 사용하고 있다. 그것은 자칫 어린 학생들에게 많은 인지 부담을 요구할 수 있으며, 그로 인해 프로그래밍 탐구에 대한 학생들의 흥미와 호기심을 저하시킬 수 있다. 그에 반해 두리틀은 “반복하다” 명령어 또는 “타이머” 객체를 사용하여 LOGO에서 재귀적 용법을 사용한 예를 구현할 수 있다<sup>2)</sup>. 이 방식은 학생들에게 구조에 도달하기 위한 중간단계를 체험할 수 있는 기회를 제공한다는 점에서 의의가 있다.

### 4) 원에 대한 고찰

원은 거북기하(Turtle Geometry), 유클리드 기하, 데카르트 기하의 관점에서 살펴볼 수 있다. LOGO에서의 원은 내재적 기하 원리를 기반으로 한 거북 기하(Turtle Geometry)의 대표적인 예이다. 거북기하는 정다각형의 극한으로 원을 그리는데 신체동조적인 특성을 가진다. 유클리드 기하에서의 원은 ‘임의의 한 정점에서 같은 거리에 있는 점들의 집합’으로 정의되는 대상으로서 원 밖의 어떤 것,

2) 김경미(2004). 객체지향형 교육용 프로그래밍 언어 '두리틀(Dolittle)'의 수학교육 활용을 참고.

즉 한 정점을 기준으로 하는 표현이다. 데카르트 기하에서는 유클리드적 개념이  $(x-a)^2 + (y-b)^2 = r^2$ 과 같은 대수적 식으로 표현되어 좌표평면에 그려지는 것을 말한다. 두리틀에서는 LOGO의 거북기하(Turtle Geometry) 관점의 원을 그릴 수 있을 뿐 아니라 유클리드 기하의 개념으로 만들어진 “원” 명령어를 이용하거나  $(x-a)^2 + (y-b)^2 = r^2$ 와 같은 방정식을 이용하여 좌표평면에 데카르트 기하 관점의 원도 그릴 수 있다. 다음은 두리틀 환경에서 원을 그려본 예제로 다음과 같이 다양하게 원을 그리고 탐구할 수 있는 기회를 제공한다. 점에서 수학교육의 큰 활용 가능성을 엿볼 수 있다.

첫째, LOGO 방식의 원 그리기(거북기하)

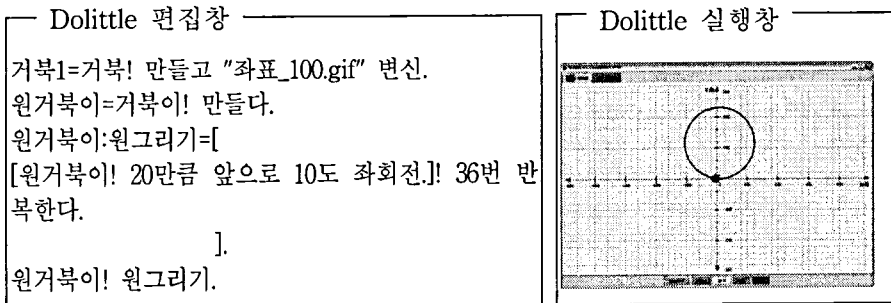
둘째, “원” 명령어 사용하여 그리기(유클리드 기하)

셋째, 원의 방정식  $x^2 + y^2 = r^2$ 을 사용하여 그리기(데카르트기하)

넷째, 원을 도형화하여 평행이동하기

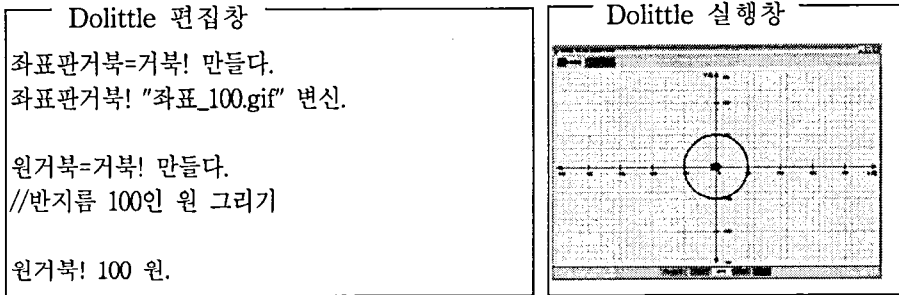
다섯째, 원의 각도 보여주며 그리기

다음은 위 다섯 가지 원 그리기 방법을 두리틀에서 실제 프로그래밍 한 결과이다. 첫 번째 방법은 거북기하 방식의 원을 두리틀에서 그린 예이다. 「원거북이! 20만큼 앞으로 10도 좌회전.」! 36번 반복한다.」와 같이 입력하여 원을 그릴 수 있는데 <그림 7>에서는 임의로 “원그리기”라는 메소드를 정의하여 그린 경우이다. 이 예제를 여러 가지 길이와 각을 변수로 받아서 그리도록 프로그래밍할 수 있으며, 버튼을 누르면 원을 그리게 하는 프로그램 등 다양한 프로그램을 작성할 수 있다. 이 예제에서 학생들은 거북이가 걸어가는 길이와는 상관없이 거북이의 방향을 얼마나 도느냐에 따라 반복 횟수가 달라진다는 것 즉, 도형의 외각의 합이 360도라는 것을 실제 조작 활동을 통해 탐구할 수 있다.



<그림 7> 두리틀의 거북기하 방식의 원

두리틀은 LOGO에서와 같이 다각형을 확장하여 원을 그릴 수 있을 뿐 아니라 “원”이란 메소드가 두리틀 프로그램에 내장되어 있어 <그림 8>과 같이 어린 초등학생들도 쉽게 원을 그릴 수 있다. <그림 8>의 「Dolittle 편집창」에서 「원거북! 100 원.」은 “원거북”이란 거북이 위치를 중심으로 반지름이 100인 원을 그리라는 것을 의미한다. 즉 유클리드 기하의 관점에서 원을 정의한 것으로 거북이가 임의의 한 정점 역할을 한다.



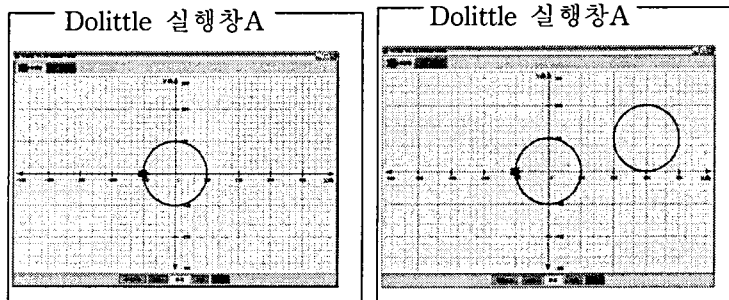
<그림 8> “원” 명령어를 사용한 원

두리틀에서는 원의 방정식  $x^2 + y^2 = r^2$ 을  $y$ 에 관한 함수로 변형하여 원을 그릴 수도 있다. <그림 9>는 중등 수학 과정에서의 활용 예제로 원의 방정식  $x^2 + y^2 = 100^2$ 을 다음과 같이 함수로 변형하여 좌표평면에 나타낼 수 있다.

$$x^2 + y^2 = 100^2 \begin{cases} y = \sqrt{100^2 - x^2}, (y \geq 0) \\ y = -\sqrt{100^2 - x^2}, (y \leq 0) \end{cases}$$

와 같이 원의 방정식  $x^2 + y^2 = 100^2$ 을  $y$ 에 관하여

정리하고, 정의역  $\{-100 \leq x \leq 100\}$  범위 안에서 거북이를  $(x, y)$  좌표로 각각 200번 반복 위치시켜 함수를 그린다. 이 예에서는 처음 거북이의 위치를  $x$ 좌표가 -100일 때  $y$ 좌표 값을 계산하여 위치하도록 하였다. 초기 거북이 위치 설정을 따로 프로그램한 이유는 두리틀에서 거북 객체의 속성으로 처음 객체가 생성될 때 위치 설정이 좌표  $(0, 0)$ 인 화면 중앙으로 되어 있기 때문이다.



<그림 9> 원의 방정식을 이용한 원 그리기와 도형화를 통한 원의 평행이동 실행결과

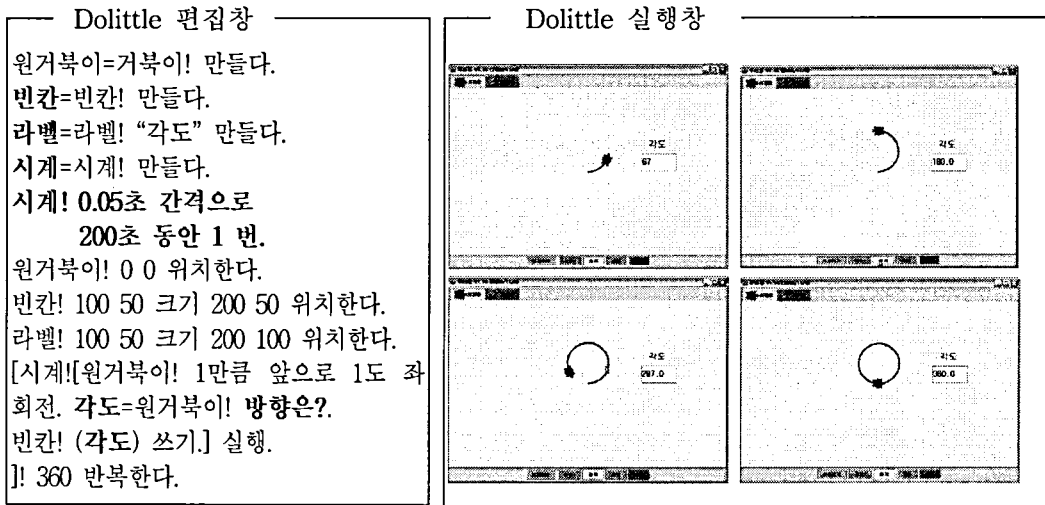
두리틀에서는 <그림 10>의 「Dolittle 편집창B」에서처럼 원의 자취를 도형화하여 복제한 후 평행이동할 수도 있다.

| Dolittle 편집창A   | Dolittle 편집창B  |
|---|--|
| <pre> 거북1=거북! 만들다. 거북1! "좌표_100.gif" 변신.  //초기 거북이 위치 설정 stx=(-100). sty = (sqrt(10000-(stx*stx))). 거북2= 거북! 만들다. 거북2! 날다 (stx) (sty) 위치하고 앉는다.  //원의 방정식 그리기 거북2: 원 =[  x  [ y1 = (sqrt(10000-(x*x))). 거북2! (x) (y1) 위치한다. x = x +1.!]! 200 반복한다. [(x)==100]! 이라면 [     [x = x -1.     y2 = 0-(sqrt(10000-(x*x))).     거북2! (x) (y2) 위치한다.     ]! 200 반복한다.]실행. ].  거북2! -100 원.</pre> | <pre> 거북1=거북! 만들다. 거북1! "좌표_100.gif" 변신. //초기 거북이 위치 설정 stx=(-100). sty = (sqrt(10000-(stx*stx))). 거북2= 거북! 만들다. 거북2! 날다 (stx) (sty) 위치하고 앉는다. //원의 방정식 그리기 거북2: 원 =[  x  [y1 = (sqrt(10000-(x*x))). 거북2! (x) (y1) 위치한다. x = x +1.!]! 200 반복한다. [(x)==100]! 이라면 [     [x = x -1.     y2 = 0-(sqrt(10000-(x*x))).     거북2! (x) (y2) 위치한다.     ]! 200 반복한다.] 실행. 거북2! -100 원. //원의 평행이동 원도형=거북2! 도형만들기. 원평행이동=원도형! 만들다. 원평행이동! 300 100 이동.</pre> |

<그림 10> 원의 방정식을 이용한 원 그리기와 도형화를 통한 원의 평행이동 편집창

두리틀에서는 도형화한 원은 확대 및 축소시킬 수 있고, 원하는 절대 좌표로 위치시킬 수 있으며, 다양한 색으로 칠할 수도 있다. 단, 함수를 도형화하기 위해서는 거북이의 자취가 연속적으로 그려져야 한다. 「Dolittle 편집창B」는 원의 자취를 도형화하여 복제한 후 x축으로 300만큼, y축으로 100만큼 평행 이동한 예제이다. 이 예제는 고등학교 10-나 2. 원의 방정식 단원과 3. 도형의 이동에서 다양하게 활용될 수 있다.

다음 <그림 11>의 「Dolittle 편집창」은 거북이의 방향 값을 이용하여 원의 중심각을 보여주며 원을 그리는 예제이다. 두리틀에 “방향은?”이라는 명령어는 현재 거북이의 머리 방향 값을 읽어온다. 「빈칸! (원거북이! 방향은?) 쓰기.」는 원거북이의 현재 방향 값을 읽어 “빈칸!”이란 텍스트필드 객체에 보여준다. 위 예제와 같이 원거북이의 방향 값을 “각도”라는 변수에 저장하여 「빈칸! (각도) 쓰기.」와 같이 프로그래밍할 수 있다. 그리고 <그림 11>의 경우에는 위에 보여주었던 예제와는 다르게 타이머(Timer) 객체도 함께 사용하여 원을 그린 것으로 거북이의 속도를 조절할 수 있는 예제이다.



<그림 11> 원 각도 보여주며 그리기

LOGO와 두리틀은 컴퓨터를 이용하여 수학적 사고력과 문제 해결력, 수학과 수학 학습에 대한 긍정적인 태도를 길러 주는 등 수학교육의 목표를 달성하기 위한 방안으로서의 역할이 큰 프로그래밍 언어이다. 두리틀은 기존의 LOGO 언어보다 어린 학생들이 습득하기 쉬우며, 다양한 GUI 함수로 학생들의 흥미와 호기심을 이끌어낼 수 있다. 또한 두리틀의 프로그래밍 과정에는 간단한 수학 내용일 지라도 항상 수학적 문제가 발생한다. 특히 두리틀의 절대좌표 화면 구성과 타이머(Timer) 객체는 그 자체가 수학적 탐구 대상이 될 수 있다. 예로 두리틀의 “위치한다” 명령어를 쓰기 위해서는 두리틀 화면의 절대좌표를 알아야 하며, 타이머 객체의 생성시 타이머의 작동 간격과 시간, 횟수 등을 사용자가 정의할 수 있는데 그 과정에 시간에 대한 계산 과정이 반드시 필요하다.

LOGO와 두리틀의 수학교육 활용 예의 비교 고찰 결과 두리틀은 LOGO에서 가장 중점적인 재귀 절차와 리스트의 개념이 어린 아이들에게는 인지하기가 너무 어렵다는 최근의 연구 결과들을 받아들여 재귀 절차를 통한 문제 해결보다는 타이머나 조건문을 사용한 반복을 사용하는데 그것은 두리틀과 LOGO의 가장 큰 차이점이라고 할 수 있다. 그리고 LOGO에서의 변수 표현은 수학에서의 표현과 매우 다르게 표현되어 학생들에게 이중의 인지부담을 안겨주는 반면 두리틀에서는 학교 수학시간에 사용하는 친숙한 한글 수학용어를 사용할 수 있으며, 수식 표현이 수학에서의 수식 표현과 거의 유사하므로 학습 연계의 효과를 얻을 수 있다. 또한 두리틀은 LOGO의 거북기하 외에도 다양한 방법으로 원을 탐구할 수 있는 환경을 제공하며, 다양한 GUI 함수를 이용해 학생들은 자신의 개성과 수준에 맞는 문제 해결 방법을 찾아 프로그래밍 할 수 있다는 점에서 LOGO보다 훨씬 수학교육적 가치가 있다.

교육용 프로그래밍 언어 LOGO와 두리틀은 다음과 같이 간략하게 비교 정리할 수 있다.

<표 2> LOGO와 두리틀 비교

|         | 로고(LOGO)  | 두리틀(Dolittle)  |
|---------|---|--|
| 개발시기    | 1969년   | 2003년  |
| 개발자     | Seymour Papert  | Susumu Kanemune(兼宗進)   |
| 형태      | 구조적·절차적 프로그래밍   | 객체지향 프로그래밍   |
| 언어지원    | 파생된 다른 종류의 언어   | 언어 자체에 다국어 지원  |
| 어순      | 영문 어순   | 한글 어순  |
| 스레드     | 지원안함  | 지원함  |
| 재사용성    | 어려움(상속 불가능)   | 용이(상속 가능)  |
| 프로그램 구조 | 복잡  | 간결   |
| 특징      | <ol style="list-style-type: none"> <li>1. 구조적·절차적 언어</li> <li>2. 거북 그래픽스(Turtle Graphics)</li> <li>3. 재귀적 언어</li> <li>3. 즉각적인 피드백을 통한 오류 수정 용이</li> </ol> | <ol style="list-style-type: none"> <li>1. 객체지향형(Object-Oriented) 언어</li> <li>2. 프로토타입(Prototype)</li> <li>3. 인크리멘탈 프로그래밍(Incremental Programming)</li> <li>4. 거북 그래픽스(Turtle Graphics)</li> <li>4. 다국어 지원(영어,일본어,한국어)</li> <li>5. 1: N 명령어 방식의 한글 EPL</li> <li>6. 분산공유를 통한 네트워크 통신 가능</li> <li>7. 다양한 GUI 함수</li> <li>8. 타이머를 이용한 애니메이션</li> </ol> |
| 수학 활용영역 | <ol style="list-style-type: none"> <li>1. 초·중등 기하영역</li> <li>2. 변수 개념의 대수영역</li> <li>3. 재귀(recursion)를 이용한 프랙탈</li> </ol>                                 | <ol style="list-style-type: none"> <li>1. 초·중등 기하영역</li> <li>2. 초·중등 대수영역</li> <li>3. 초·중등 함수영역</li> <li>4. 로봇제어</li> <li>5. 시뮬레이션 및 게임 제작</li> <li>6. 분산 공유를 통한 네트워크</li> </ol>   |

#### 4. 기하·함수 교수-학습을 위한 두리틀의 활용 방안

수학 교수-학습에 두리틀 언어를 활용하는 방법은 크게 두 가지가 있다. 하나는 학생들 스스로 두리틀 언어를 이용하여 직접 프로그래밍 활동을 하는 과정에서 수학적 개념을 학습하는 것이고, 다른 하나는 교사 또는 수학교육 연구자가 고안해 놓은 두리틀 활용 프로그램을 수학 교수-학습에 활용하는 것이다. 두 가지 활용 방법 모두 높은 수학교육적 효과를 창출할 수 있으므로, 두 가지 활용 방안에 대한 연구가 함께 이루어져야 한다. 두리틀 분석 결과 두리틀은 초·중등 수학 교과 영역의 전반적인 부분에 모두 활용 가능하며, 특히 기하영역과 함수영역에서 높은 활용 가능성이 발견되었다. 따



라서 본 논문에서는 연구자가 고안한 기하판과 좌표판, 삼각함수판을 이용한 초등 기하영역과 중등 함수영역에 두리틀을 활용할 수 있는 방법을 소개하고자 한다. 이것은 거북 객체의 “변신” 메소드에서 아이디어를 얻은 것으로, 거북 객체의 이미지를 기하판/좌표판 이미지로 바꿈으로써 학생들에게 기하/함수를 탐구할 수 있는 환경을 제공할 수 있다.

### 1) 기하판(Geoboard)를 이용한 초등 기하 영역 활용

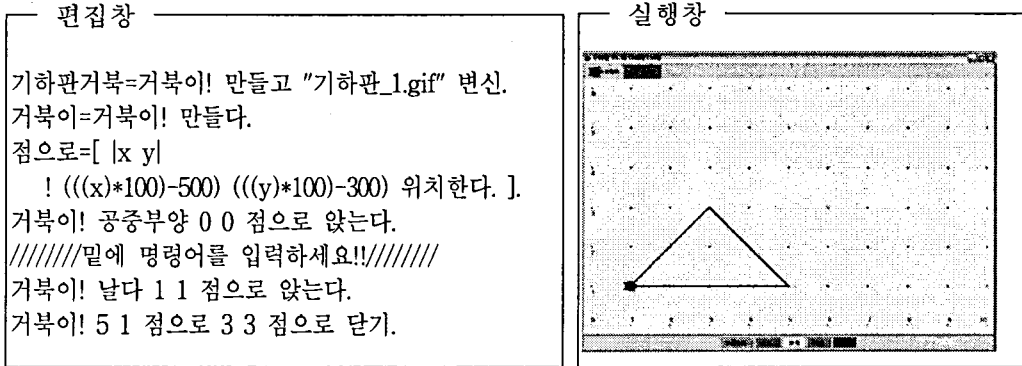
현재 기하교육은 형식기하와 직관기하가 단절되어 공간 감각력을 통한 창의적 사고력을 개발시키는 것이 어렵고, 학생의 눈높이에 맞는 기하교육이 이루어지고 있지 못하다. 이를 해결하기 위한 방안으로 활동 중심의 수학교육이 등장하고 컴퓨터를 활용한 탐구형 소프트웨어가 개발되고 있다. 두리틀은 기하영역에 있어 LOGO의 거북 기하(Turtle Geometry) 방식을 수용하여 활용할 수 있다. 초등 수학 교과내용 영역 중 도형 영역은 대부분 두리틀로 활용 가능하며 다른 영역도 교사가 만든 프로그램을 통해 쉽게 개념을 학습할 수 있기 때문에 거의 모든 영역에 대하여 활용가능하다고 볼 수 있다.

그러나 어린 초등학생이 두리틀 프로그래밍을 하기에는 약간의 제한사항이 수반된다. 초등학교 1학년에서 3학년까지는 각의 크기에 대한 개념을 아직 학습하지 않은 단계이므로 두리틀의 거북 기하 명령어를 사용하는 것은 바람직하지 않다. 따라서 본 연구자는 저학년의 도형 학습에는 기하판(Geoboard)의 이미지와 두리틀의 “위치한다” 명령어를 사용하여 학습할 수 있는 방안을 연구하였다. 두리틀에서는 거북 객체를 생성한 후에 기하판 이미지로 변신시키면 기하판 교수-학습 환경이 쉽게 만들어진다. 기존의 기하판과 두리틀의 기하판이 다른 점은 기존의 기하판 환경에서는 실제 기하판에 고무줄로 연결하여 원하는 도형을 그리는 반면 두리틀의 기하판 환경에서는 학생들이 원하는 점의 좌표를 알아야 한다는 것이다. 초등학교 저학년에게는 절대좌표를 이용하여 도형을 그리는 것은 쉬운 일이 아니다. 더욱이 절대좌표 평면에서 음수를 사용해야 한다는 점은 초등 수학에 두리틀을 활용 시 단점이 될 수 있다. 또한 두리틀 화면의 길이 단위는 픽셀이므로 접해보지 못한 단위에 대하여 학생들로 하여금 혼란을 야기시킬 수 있다. 따라서 본 연구자는 초등 저학년 학생들을 위하여 음수의 절대 좌표를 없애고, 픽셀 단위의 정수 사용을 0과 자연수 사용으로만 제한하도록 고안하였다.

학생들이 두리틀 환경에서 수학 학습을 할 때 기계적인 조작 활동보다는 수학적 탐구와 개념 형성에 집중할 수 있도록 해야 한다. 수학 개념을 형성하는데 불필요한 요인들을 제거해 주는 것은 수학 학습에 있어 매우 중요한 부분이다. 따라서 양수 부분만 표현하는 기하판 이미지 “기하판\_1.gif”와 “기하판.dtl” 이란 두리틀 프로그램 파일을 만들어 학생들이 “기하판.dtl” 파일을 열어 학습할 수 있도록 하였다. 음수 사용의 문제점과 픽셀 단위의 문제를 없애기 위해 “기하판.dtl” 파일에는 다음과 같은 내용의 프로그램이 저장되어 있다.

“점으로” 라는 메소드는 변수  $x$ ,  $y$ 에 화면 좌표 값을 인수로 받는다. “점으로” 메소드는 받은 두 개의 좌표 값에 100을 곱한 뒤 각각 -500 -300을 해준다. 학생들은 「거북이! 3 4 점으로.」와 같이

간단하게 원하는 점의 위치를 입력하여 도형을 그릴 수 있다. 거북이의 자취를 감추고 싶을 때는 「거북이! 날다 5 5 점으로 앉는다.」와 같이 입력하면 된다.

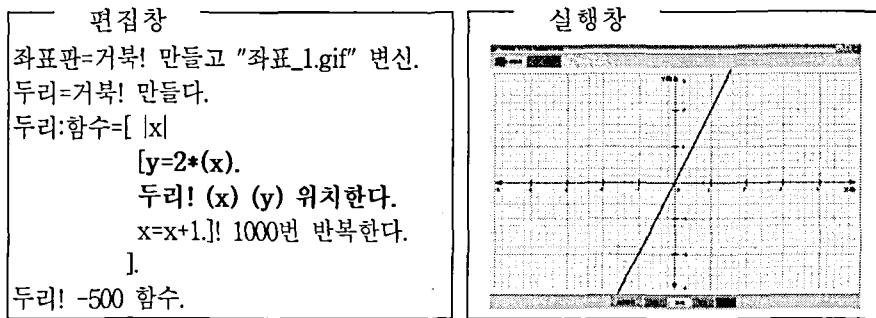


<그림 12> “기하판.dtl” 파일

기하판(Geoboard)을 사용한 수학교육 활용은 여러 가지 도형을 그리며 도형의 성질을 학습할 수 있을 뿐 아니라 그린 도형의 넓이 학습에도 매우 유용하다. 다음은 넓이가 같은 삼각형들을 그리는 활동이다. <그림 12>의 편집창에서 「거북이! 날다 1 1 점으로 앉는다. 거북이! 5 1 점으로 3 3 점으로 닫기.」의 부분을 「거북이! 5 1 점으로 6 3 점으로 닫기.」 또는 「거북이! 5 1 점으로 1 3 점으로 닫기.」와 같이 한 꼭지점의 x 좌표만 변경시켜 본다. 다양한 삼각형 모양이지만 모두 넓이가 4로 같다. 학생들은 기하판의 4개의 점을 이은 정사각형의 넓이를 1로 생각하고 삼각형에 넓이가 1인 사각형이 몇 개 들어가는지 탐구해 봄으로써 올바른 넓이 개념을 형성할 수 있다.

## 2) 좌표판과 삼각함수판을 이용한 중등 함수 영역 활용

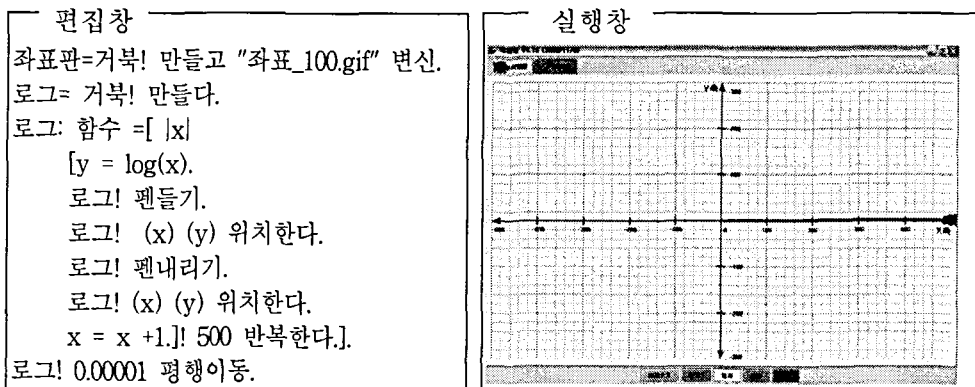
수학에 있어 함수영역은 어떤 영역보다도 중요하게 여겨져 왔다. 기존 LOGO는 함수영역에 활용되기에는 많은 제약사항이 있어 널리 활용되지 못하였지만, 두리틀은 간단한 이미지 변신을 통해 함수를 탐구할 수 있는 환경을 제공한다는 점에서 큰 의의가 있다. 연구자는 두리틀 프로그래밍 분석과 두리틀 프로그램 환경 분석을 통해 두리틀을 함수 영역에 활용할 수 있도록 하였다. 두리틀에서는 쉽고 간단하게 함수를 그릴 수 있으며, 다양한 함수의 탐구 활동이 가능하다. 다음은  $y=2x$  함수의 그래프를 두리틀에서 그리는 예제이다. 두리틀에서는 <그림 13>과 같이 간단하게 함수의 그래프를 그릴 수 있다. 위 예제는 x좌표를 인수(Parameter)로 받아 함수를 그려주는 “함수”라는 메소드를 정의한 것이다. x좌표가 1만큼 증가할 때마다 함수식의 조건에 맞는 x좌표와 y좌표에 거북이를 위치시켜 함수를 그린다.



<그림 13> 1차 함수 그리기

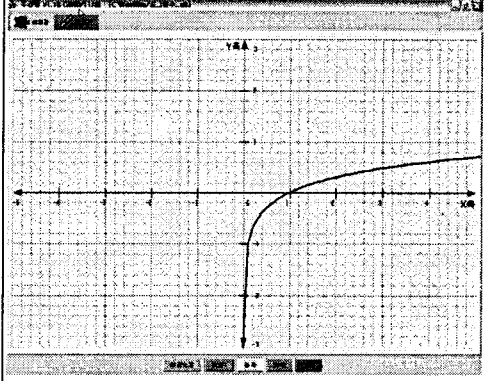
위 예제에서 볼 수 있듯이 두리틀은 수학에서의 수식 표현과 거의 유사하게 표현할 수 있으며, 프로그래밍 과정 속에서 함수의 원리도 터득할 수 있다는 큰 장점을 가지고 있다. 그러나 두리틀이 수학교육을 위해 고안된 언어가 아니라는 점에서 약간의 제한점을 가지고 있는 것 또한 사실이다. 본 연구자는 두리틀을 함수 영역에 활용하는데 있어 몇 가지 제한점들을 발견하고 해결방안을 모색하였다. 지수, 로그함수를 예로 두리틀 환경에서 함수를 그리는 과정에서의 제한점과 그 해결방안에 대하여 고찰하도록 하자.

두리틀 프로그램에는 지수함수(exp)와 로그함수(log), 삼각함수(sin, cos, tan)가 모두 내장되어 있어 쉽게 그릴 수 있다는 큰 장점을 가지고 있다. 그러나 두리틀 화면의 단위가 픽셀 단위이기 때문에 다항함수를 제외한 지수, 로그함수와 삼각함수를 그릴 때는 그것이 문제가 된다. <그림 14>는 두리틀에 내장되어 있는 "log" 함수를 사용하여 함수를 그린 예제인데, 이 같은 환경에서는 학생들이 로그함수의 성질을 학습하고 탐구하기가 어렵다. 로그함수가 항상 (1, 0)을 지난다는 것을 학생들이 <그림 14>에서는 알아보기가 힘들다. 또한 다음과 같이 프로그래밍 한 경우는 로그함수를 도형화시켜 평행 이동시키는 것이 불가능하다.



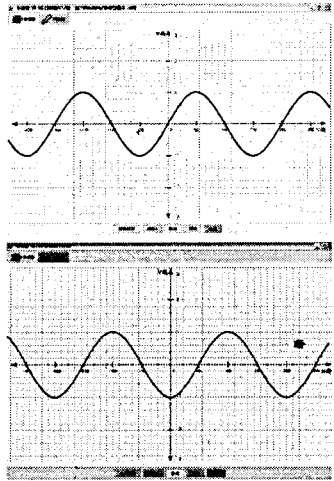
<그림 14> 수정 전 로그함수

따라서 연구자는 학생들이 좀 더 알아보기 쉽고 다양한 탐구 활동이 가능할 수 있도록 좌표판 이미지를 다시 고안하였으며, 새로운 함수 그래프 프로그램을 작성하였다. 좌표판은 100픽셀을 1단위로 바꾸어 교과서에 있는 좌표평면과 유사하게 만들어, 학생들이 좌표 값을 정확하게 볼 수 있도록 하였다. 또한 <그림 15>의 「편집창」을 보면, 거북이의 초기값을 우선 지정하게 함으로써 함수를 도형화시켜 평행 이동이 가능하도록 하였다. 그리고 함수의  $x$ ,  $y$  좌표 값에 각각 100을 곱하여 함수를 확대하여 보여주게 함으로써 단위 불일치로 오는 문제점을 해결하였다. 특히 이렇게 하는 경우 수학에서 사용하는 함수식과 거의 유사하게 표현할 수 있다는 장점이 있다. 다음은 수정된 지수함수의 프로그램 내용과 실행 결과이다.

| 편집창   | 실행창   |
|---|---|
| <pre> 좌표거북=거북! 만들고 "좌표_1.gif" 변신. stx=(0.001). sty = log(stx). //초기화 로그= 거북! 만들다. 로그! 날다 (stx*100) (sty*100) 위치하고 앉는다. 로그: 함수 =[  x        [ y = log(x).       로그! (x*100) (y*100) 위치한다.       x = x +(0.1). ]! 500 반복한다.       ]. 로그! 0.001 함수.           </pre> |  |

<그림 15> 수정 후 로그함수

두리틀은 다수의 거북 객체를 생성하여 이미지로 변신할 수 있기 때문에 다양한 환경으로 바꿀 수 있다. 초등 기하 교육에 있어서는 여러 단위의 기하판(1단위, 10단위, 100단위)을 사용하여 다양한 탐구 활동을 할 수 있으며, 중등 함수 교육을 위해서는 여러 단위의 좌표판(1단위, 5단위, 10단위, 100단위)을 사용하여 함수의 그래프 변화를 탐구할 수 있다. 지수, 로그함수 뿐 아니라 삼각함수 또한 <그림 16>과 같이 쉽게 그래프의 변화를 확인할 수 있다는 점에서 두리틀의 수학교육 활용 가능성은 매우 크다고 할 수 있다.

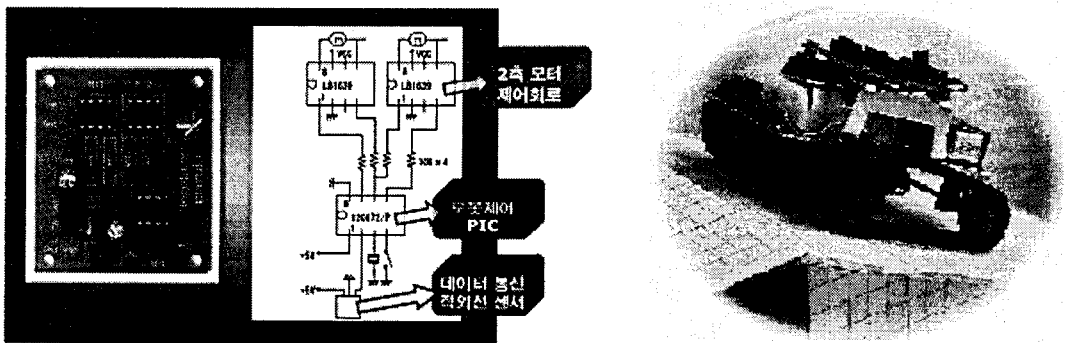
| 편집창  | 실행창  |
|--|--|
| <pre>좌표거북=거북! 만들고 "삼각함수판.gif" 변신. 삼각 = 거북! 만들다 . stx=(-700). sty = sin(stx). //초기화 삼각! 날다 (stx) (sty) 위치하고 앉는다.  삼각: 함수 =[  x                  [y = sin(x).                 삼각! (x) (y*100) 위치한다.                 x = x +1.!] 1100 반복한다.                 ]. 삼각! -700 함수 . 삼각도형=삼각! 도형만들기. 삼각도형! 90 0 이동.</pre> |  |

<그림 16> 함수  $y = \sin(x)$ 의 그래프 그리기와 평행 이동

### 5. 두리틀 로봇의 활용

두리틀 언어의 큰 장점 중에 하나가 로봇 제어가 쉽고 간단하다는 것이다. 두리틀로 입력된 로봇 제어 명령은 컴퓨터의 직렬포트에 연결된 적외선 장치를 통해 로봇으로 전송된다. 두리틀의 로봇 제어 기능은 매우 단순하고 손쉽게 프로그램을 수정할 수 있는 장점을 가지고 있다. 로봇 프로그래밍은 학생들의 흥미와 호기심을 유발할 수 있으며, 프로그래밍 과정에서 문제해결력을 향상시킨다.

다음은 두리틀로 제어하는 교육용 로봇 아키텍처와 실제 두리틀 로봇의 사진이다.



<그림 17> 두리틀로 제어하는 교육용 로봇 아키텍처와 두리틀 로봇

두리틀 로봇을 수학교육에 활용할 수 있는 방법은 다양하다. 기하 교육에 있어 화면 속에서 거북이가 그린 도형을 로봇이 실제 그리는 것을 보여주는 것만으로도 학생들의 높은 흥미를 유발할 수 있으며, 두리틀 프로그램을 조작하여 로봇을 제어하면서 많은 수학적 내용을 습득할 수 있다. 특히 두리틀 로봇은 각을 입력하여 로봇을 제어하는 것이 아니라 시간으로 제어를 하기 때문에 다양한 변수로 인한 문제들을 해결하는 과정에서 학생들은 수학적 사고력과 문제해결력을 키울 수 있다.

두리틀 로봇으로 다양한 도형을 그릴 수 있으며, 프랙탈과 같은 수학의 아름다운 현상들을 로봇으로 구현해보는 것도 좋은 경험이 될 것이다. 이산수학의 한 붓 그리기, 오일러 회로, 헤밀톤 회로를 로봇을 이용해 그려보거나, 로봇의 분산공유를 이용하여 로봇들 간의 협동 작업 등 교육에 있어 로봇의 활용 가능성은 매우 크다. 특히 두리틀 로봇은 기존의 로봇보다 가격이 저렴하므로 손쉽게 구입할 수 있다는 장점이 있다. 다음은 두리틀 로봇을 이용하여 삼각형과 사각형을 그리는 예제이다.

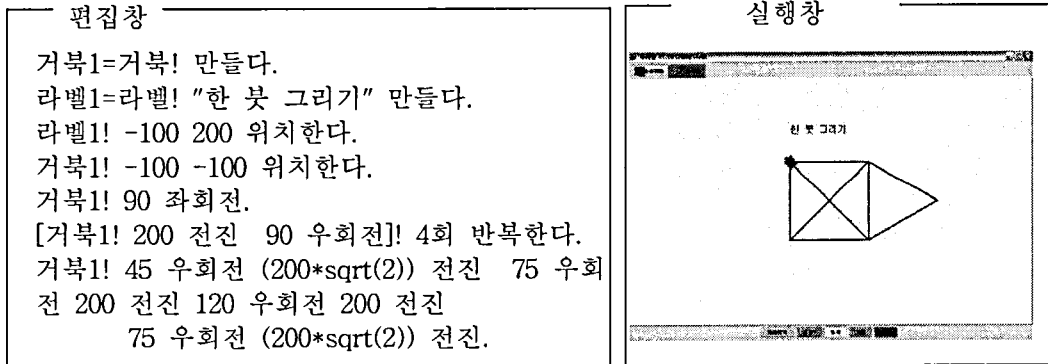
| 삼각형   | 사각형  |
|---|--|
| 로봇=시리얼_포트! 만든다.<br>로봇: 사각형그리기= [!로봇시작<br>15 전진 30 오른쪽전진<br>15 전진 30 오른쪽전진<br>15 전진 30 오른쪽전진<br>로봇끝].<br>로봇! "com1" 포트열기.<br>로봇! 사각형그리기.<br>로봇! 로봇실행.<br>로봇! 포트닫기. | 로봇=시리얼_포트! 만든다.<br>로봇: 삼각형그리기= [!로봇시작<br>4 반복시작<br>10 전진 20 오른쪽전진<br>반복끝<br>로봇끝].<br>로봇! "com1" 포트열기.<br>로봇! 삼각형그리기.<br>로봇! 로봇실행.<br>로봇! 포트닫기. |

<그림 18> 로봇을 이용한 삼각형, 사각형 그리기

로봇 명령어는 두리틀 명령어와 별도로 존재하는데, “포트열기”는 로봇에게 명령을 전달하기 위한 송신기의 사용 초기화를 의미하며, “포트닫기”는 송신기를 종료한다는 것을 의미한다. “로봇시작”과 “로봇끝”은 로봇 명령의 시작과 끝을 알리는 것이다. <그림 18>의 삼각형은 한 변의 길이가 15인 정삼각형을 그리는 것이다. 「15 전진 30 오른쪽전진」은 15만큼 전진한 후에 30초 동안 오른쪽 바퀴만 앞으로 진행하라는 명령이다.

<그림 19>는 두리틀 로봇으로 한 붓 그리기를 한 예제이다. 두리틀 로봇의 단점은 정확한 각을 표현할 수 없다는 것인데 이것은 추후 연구를 통하여 해결해야 할 점이다. 그리고 두리틀 로봇 제어 프로그래밍은 두리틀 문법과 다른 문법을 가지고 있어서 로봇을 제어하려면 또 다른 문법을 배워야 하는 문제가 있다. 따라서 두리틀 문법을 그대로 이용하여 로봇을 제어할 수 있도록 하였다. 현재 두리틀에서 거북 객체를 이용하여 코드를 작성하면 로봇을 제어하는 코드로 변환하는 Parser를 설계

구현한 상태이다(권대용, 2004b).



<그림 19> 두리틀 로봇을 이용한 한 붓 그리기

### III. 결론

기존의 수학교육에서는 인지(cognitive) 학습만이 강조되었고, 학생들의 동기를 유발시킬 수 있는 경험(experiential)학습은 배제되어 왔다. 이런 상황에서 두리틀을 활용한 교수-학습은 학생들에게 수학교과 내용에 대한 인지학습보다는 수학이 내재된 경험학습을 제공할 수 있다는 점에서 매우 큰 의미가 있다. 학생들은 두리틀로 만들어진 프로그램을 통해 다양한 수학적 경험을 체험할 수 있으며, 학생들이 직접 두리틀 언어로 프로그래밍 하는 과정에서 문제해결력과 수학적 사고력, 반성적 사고 등 중요한 인지능력을 향상시킬 수 있다. 두리틀 프로그래밍 활동은 다양한 수학 탐구 활동의 마이크로월드(Microworld)를 통해 수학의 유용성과 심미성을 체험할 수 있는 기회를 제공하고, 두리틀 로봇과 다양한 GUI(Graphical User Interface) 함수의 시각적 효과는 현대 학생들의 높은 흥미를 유발할 수 있으며, 수학 과목에 대한 학생들의 부정적인 성향을 변화시킬 수 있다. 또한 두리틀은 분산 공유를 통한 네트워크 체험이 가능하기 때문에 소그룹별 협동학습을 통해 학생들의 의사소통력을 향상시킬 수 있으며, 역할 분담 활동을 통해 사회성을 함양시킬 수 있다.

본 연구를 통해 수학 교수-학습을 위한 두리틀의 활용 영역이 기하, 함수, 대수(변수), 이산수학 영역 등 수학의 광범위한 영역에서 활용이 가능하다는 것을 알 수 있었다. 특히 초등 기하 영역과 중등의 함수 영역에서 높은 활용 가능성이 발견되었다. 초등 영역에서는 기하판 이미지와 아이들이 조작하기 쉬운 인터페이스를 고안하기 위한 몇 가지 아이디어를 통해 기하 영역에 대한 다양한 교수-학습을 가능하게 하였으며, 중등 영역에 있어서는 함수의 그래프를 그릴 수 있는 좌표판과 삼각함수판 이미지를 고안하고, 픽셀 단위의 컴퓨터 환경에서 발생하는 몇 가지 문제점들을 해결함으로써 수학 기호의 함수 정의 표현에 최대한 가깝게 기술할 수 있도록 하였다.

두리틀 환경이 수학 교수-학습에 직접적으로 활용되기에는 아직 많은 제한점을 수반하고 있지만, 기존의 LOGO보다는 더욱 다양하고 풍부한 마이크로월드(Microworld)의 환경을 제공해 줄 수 있다는 점에서 큰 의미를 가진다. 특히 학생들에게 수학 교과 내용의 학습 뿐 아니라 다양한 문제 해결력과 창의적 사고, 직관적 사고를 향상시킬 수 있는 기회를 제공할 수 있다는 점에서 큰 의의가 있다. 두리틀의 개발 취지 목적이 수학교육을 위해 만든 언어가 아니라는 점에서 수반되는 제한점과 문제점들은 앞으로 꾸준한 두리틀 분석 연구와 적절한 교수-학습 방법의 연구 및 두리틀의 적용 사례 연구를 통해서 해결해 나가야 할 것이다.

## 참 고 문 헌

- 권대용 · 길혜민 · 염용철 · 유승욱 · susmu kanemune · yasushi kuno · 이원규 (2004a). 중등컴퓨터과학 교육을 위한 객체지향형 EPL 두리틀의 적용과 평가. 한국컴퓨터교육학회논문지 7(6), pp.1-12.
- 권대용 · 염용철 · 유승욱 · 이원규 (2004b). 두리틀 프로그래밍과 로봇 프로그래밍의 일원화. 한국컴퓨터교육학회 학술발표논문집 8(1), pp.361-364.
- 길혜민 · 최해심 · 유승욱 · 이원규 (2004). 객체지향형 교육용 프로그래밍 언어 'Dolittle'의 적용과 평가. 한국컴퓨터교육학회 학술발표논문집 8(1), pp.103-108.
- 김경미 (2004). 객체지향형 교육용 프로그래밍 언어 '두리틀(Dolittle)'의 수학교육 활용. 고려대학교 대학원 석사학위 논문.
- 박소영, 권대용, 염용철, 유승욱, 이원규 (2004). "객체지향형 교육용프로그래밍언어 '두리틀' 시각화 프로그래밍 지원 시스템", 컴퓨터교육학회 하계 학술대회, pp.403-408.
- 박춘란 (1991). 수학 교육용 프로그래밍 언어의 설계와 활용, 서울대학교 대학원 석사학위 논문.
- 백영균 (1988). 컴퓨터 프로그래밍에 대한 심리학적 접근-LOGO를 중심으로. 교육공학연구 4(1), pp.145-165.
- 백영균, 우인상 (1994). LOGO 프로그래밍의 수업방법이 문제해결력에 미치는 효과에 관한 연구. 교육공학연구 9(1), pp.73-90.
- 류희찬 (1994). LOGO를 통한 수학 학습: 그 내용과 방법. 한국수학교육학회지 시리즈 A <수학교육> 33(2), pp.197-207, 서울: 한국수학교육학회.
- 이종영 (1994). 프로그래밍 맥락에서 변수 개념 지도의 가능성. 서울대학교 대학원 석사학위논문.
- 혜선, 권대용, 염용철, 유승욱, 이원규 (2004). "객체지향형 교육용프로그래밍언어 '두리틀'을 위한 객체트리 설계 및 구현", 컴퓨터교육학회 하계 학술대회, pp.399-402.
- 조한혁 (1991). 교육용 언어의 설계에 대한 연구. 서울대학교 사대논총 제 43호.
- \_\_\_\_\_ (2003). 컴퓨터와 수학교육. 한국수학교육학회 시리즈 A <수학교육> 42(2), pp.177-191, 서울: 한국수학교육학회.



- 최은조, 염용철, 유승욱, 이원규 (2004), "성인교육에 있어서 객체지향형 EPL 두리틀 적용 및 분석", 컴퓨터교육학회 하계 학술대회, pp.371-374.
- 최해심, 염용철, 유승욱, 이원규 (2004), "객체지향형 교육용프로그래밍언어 두리틀의 다중 예약어 지원 체계", 컴퓨터교육학회 하계 학술대회, pp.365-370.
- 황우형 (1999). LOGO 언어의 중등수학교육 활용방안. 한국수학교육학회 시리즈 A <수학교육> 38(1), pp.15-35, 서울: 한국수학교육학회.
- 兼宗進 & 御手洗 理英 & 中谷 多哉子 & 福井 眞吾 & 久野 靖 (2001). 學校教育用オブジェクト指向言語「ドリトル」の設計と實装(학교교육용 객체지향언어 'Dolittle'의 설계와 구현). 情報處理學會論文誌 42(11), pp. 78-90.
- 兼宗進(가네무네) (2003). 教育利用を目的としたオブジェクト指向言語の研究(교육이용을 목적으로 한 오브젝트 지향 언어의 연구). 쓰쿠바대학 박사학위논문.
- 兼宗進 & 中谷多哉子 & 御手洗理英 & 福井眞吾 & 久野靖 (2003). 初中等教育におけるオブジェクト指向プログラミングの 實践と評価(초중등교육에서 객체지향형 프로그래밍의 실천과 평가). 情報處理學會論文誌 44(13), pp.58-71.
- 中谷 多哉子 & 兼宗 進 & 御手洗 理英 & 福井 眞吾 & 久野 靖 (2002). 오브젝트스트림: 오브젝트指向言語 による初中等プログラミング教育の提案(화면을 뒤흔쳐나온 오브젝트: 자립형로봇을 활용한 정보교육의 제안). 情報處理學會論文誌 42(6), pp 1610-1624.
- ACM(2003). *A Model Curriculum for K-12 Computer Science*. Final Report of the ACM K-12 Education Task Force Curriculum Committee. <http://acm.org/education/k12>.
- George Lukas & Joan Lukas (1986). *LOGO: Principles, Programming, & Projects*. California : Brooks/Cole.
- Joy C. Kataoka & James R. Patton (1996). *Integrated Programming and Mathematics : An Attractive Way to Plan for Generalization*. Retrieved May 7, 2004, from [http://www.cldinternational.org/c/@s2\\_lkXLqxJWM/Pages/mathseries.html](http://www.cldinternational.org/c/@s2_lkXLqxJWM/Pages/mathseries.html)
- Judith Gal-Ezer & David Harel (1999). What (Else) Should CS Educators Know?. Communications of the ACM 41(9), pp.77-84.
- Kris Howell (2003). First Computer Languages. JCSC 18(4), pp.317-331.
- National Council of Teachers of Mathematics. (1989). *Curriculum and evaluation standards for school mathematics*. Reston, Va.: NCTM.
- National Council of Teachers of Mathematics. (2000). *Principles and Standards for School Mathematics*. Reston, Va.: NCTM.
- Papert, Seymour (1980). *Mindstorm : Children, Computers, and Powerful ideas*. New York, Basic Books. 류희찬 역, *로고와 아동*.

- Susumu Kanemune, Wongyu Lee & Yasushi Kuno (2004). International Collaboration in School Based on Multilingual Programming Language, *IPSJ* **2004(9)**, pp.7-12.
- Shuji Kurebayasi & Susumu Kanemune (2004). Comparison between robot control programming and application programming in a junior high school, *IPSJ* **2004(9)**, pp.21-28.
- Ursula Wolz & Edward Conjura (1994). Integrating Mathematics and Programming into A three tired Model for Computing Science Education. *SIGSCE(Special Interest Group on computer Science) Bulletin ACM*, pp.223-227.
- Vicki L. Almstrum, Orit Hazzan, David Ginat & Tom Morley (2002). Import and Export to/from Computing Science Education : The Case of Mathematics Education Research. *ACM SIGCSE* **34(3)**, pp.193-194.
- Zimmermann, W. & Cunningham, S. (1991). What is Mathematics Visualization?’, in Zimmerman, W. & Cunningham, S.(Eds.), *Visualization in Teaching and Learning Mathematics*. Mathematical Association of America. pp.1-8.