

컴퓨터 프로그래밍을 활용한 수학학습자료 개발에 관한 연구

- 중학교 2학년을 중심으로 -

이 선 란(신광여중)

I. 서 론

A. 연구의 필요성과 목적

어느 시대의 교육개혁이든지 개혁은 그 시대의 사회적 성격과 밀접히 관련된다. 정보화 시대로 특 징지워지는 현대 사회의 모든 분야에서 수 많은 정보의 선정, 조직, 저장 및 활용 등을 하는데 있어서 필수적인 도구는 컴퓨터라 할 수 있다. 따라서, 이제 우리는 학교 컴퓨터 교육시대의 중요성을 생각하 며 국민 컴퓨터 교육 시대로 전환하여야 할 시점에 있다고 볼 수 있다(이태욱,1991). 이에 부응하여, 수학교육에서도 현대의 정보화 사회에 맞는 수학적 인재들을 양성하는데 일익을 담당해야 하리라고 본다.

Bright,G.W. 와 Cliffliao,Y.K.(1991)는 "학생들은 컴퓨터 프로그래밍 활동을 통하여 몇 가지 인지적 인 기능 즉 추론 기능(reasoning skills), 논리적 사고(logical thinking) 그리고 계획하는 기능(planning skills), 일반적인 문제 해결 기능 등을 얻을 수 있다."고 제시하고 있다. 또한 수학교육 관점에서, Shu mway(1984)는 특별한 수학적 활동에 대하여 유사한 프로그래밍 활동을 해 봄으로써 얻어지는 프로그 래밍의 다양한 효과를 컴퓨터 소양, 특수한 수학적 개념, 일반적인 수학적 개념, 수학적 사고, 문제 해 결, 논리적 추론 등 여섯가지로 분류하여 제시하였고, Hatfield 와 Kieren(1972)도 실험연구로 중학생 들에게 2년 동안 4개의 수학 단원을 가르치면서 컴퓨터 프로그래밍을 지도하였는데, 실험집단과 통제 집단이 차이점을 첫 해에는 발견하지 못하였지만 두번째 해에는 수 단원에서 실험집단이 더 우수함을 발견하였다.

이러한 연구 결과로 인하여 수학교육에 관련된 최근의 저널이나 학회지에서는 프로그래밍을 수학 교육에 활용하는 방안과 그 효과에 관한 연구, 수학교육을 위한 새로운 프로그램의 개발 등을 소개하 고 있으며, 더우기 몇몇 선진국들은 학교 교육에 컴퓨터 프로그래밍을 활용하자는 주장들을 수학과 교육과정과 교과서에 반영해오고 있다. 일본에서는 1995년도 부터 사용할 새로운 수학 교육과정에서 고등학교의 수학을 수학 I, 수학 II, 수학 III, 수학 A, 수학 B, 수학 C 로 구분하면서 수학 A, 수학 B 에서 순서도, 프로그래밍, 컴퓨터를 이용한 계산, 알고리즘 등의 내용으로 컴퓨터의 응용을 포함하고 있다 (이종락, 1991). 그리고 미국의 중·고등학교용 수학 교과서들을 살펴보면, 간단한 BASIC 언어의 지도 와 동시에 이들을 활용하여 수학적 문제를 해결하도록 하는 컴퓨터 프로그래밍 활동 단원을 학습내용 의 수준과 적절하게 맞추어 제시하고 있다.

이에 비추어 우리나라의 현행 수학과 교과서의 내용을 조사하여 보면, 중학교용 수학과 교과서에 서는 컴퓨터의 기본원리와 관련된 내용으로 「이진법」을 소개하고 있고, 고등학교 수학과 교과서인 '수 학 I'과 '수학 II'의 「수열」단원에서 알고리즘과 순서도를 프로그램없이 지도하고 있을 뿐으로, 컴퓨 터 프로그래밍이 수학과 교과서에서는 전혀 취급되지 않고 있다.

그러나, 우리나라에서도 최근에는 박임숙(1986), 김창동(1987), 강옥기(1991))등 많은 수학 교육 관

계자들이 수학교과에 컴퓨터 프로그래밍을 도입해야 한다고 주장하고 있다. 그런데 컴퓨터 프로그래밍의 내용과 활용방법 등을 수학과 교과서에 도입 하고자 한다면, 먼저 그러한 내용들을 포함하는 수학 학습자료를 개발하는 것이 선행 되어야 할 것이다.

따라서, 본 연구에서는 외국의 수학 교과서에서 컴퓨터 프로그래밍을 지도하고 활용하는 내용과 학습자료의 구성 형태 등을 분석하여, 이를 기반으로 우리나라 수학교육과정에 맞도록 컴퓨터 프로그래밍이 활용된 수학학습자료를 개발하여 제시함으로써 중학교 수학과 교육과정에 컴퓨터 프로그래밍을 도입하는데 대한 시사점을 얻는 것을 목적으로 하고 있다.

B. 연구 방법 및 절차

앞에서 제시한 본 연구의 목적을 달성하기 위하여 연구의 방법과 절차는 “문헌연구”와 “개발”의 두 가지 방법을 사용하였다. II장의 컴퓨터 프로그래밍에 대한 고찰은 컴퓨터 프로그래밍의 교육적 효과에 대해 연구한 국내외의 여러 선행 연구 논문을 조사하여 수학교육과정에 컴퓨터 프로그래밍의 도입 필요성에 대한 근거를 제시하였고, 수학과 교과서에 도입된 컴퓨터 프로그래밍에 대한 분석은 우리나라와 미국의 수학교과서를 분석하되, 미국의 수학과 교과서를 구체적으로 분석하여 본 연구에서 개발하고자 하는 수학학습자료의 구성 체계에 대한 이론적인 틀을 제공하였다. III장에서는 수학교육과정에서의 컴퓨터 프로그래밍의 특성과 활용 방법에 대해서 고찰하고 수학 학습내용에 관련된 컴퓨터 프로그래밍 연습문제(PERCs)의 구성형태 및 설계 방법 등을 알아봄으로써 컴퓨터 프로그래밍을 활용한 수학학습자료 개발을 위한 이론적 배경을 제시하였다. IV장에서는 우리나라 중학교 2학년의 수학교과 의 각 학습영역 즉, 수와 연산, 방정식과 부등식, 함수, 통계, 도형 등의 5개 영역에서 BASIC 프로그래밍 활동과 관련될 수 있는 학습문제 12개를 선정하여 수학학습자료를 개발하였다. 이 때, 모든 프로그램은 GW-BASIC 을 사용하여 작성하였다.

C. 연구의 제한점

본 연구를 수행하는 데 있어서 고려할 몇 가지 제한점을 제시하면 다음과 같다.

첫째, 각 영역별 학습내용중 컴퓨터 프로그래밍이 활용될 수 있는 학습문제 선정시 객관적인 자료에 입각한 것이 아니라 연구자의 현장경험과 문헌연구를 토대로 한 선정이므로 절대적인 것은 아니며 다른 학습문제도 선정 가능하다.

둘째, 중학교 수학과 교육 과정에 도입될 수 있는 BASIC의 내용과 수준은 1학년 기술 교과에서 지도되고 있는 BASIC 명령어들을 기준으로 하면서 연구자의 주관에 의해 분석하는 것이므로 실제 현장에서 프로그래밍 활용시에는 필요에 따라 더 수준 높은 BA-SIC 명령어 또는 내장함수의 사용도 가능하다.

셋째, 현행 학교 교육 과정에서는 중학교 1학년 기술 교과에서 프로그램 언어 및 컴퓨터 프로그래밍을 지도하고 있으므로 컴퓨터 프로그래밍을 활용한 수학학습자료는 중학교 2학년에 한하여 개발하였다.

넷째, 본 연구에서 사용한 컴퓨터 언어는 현재 가장 보편적으로 알려져 있고 가르쳐지고 있는 BASIC 언어를 사용하였는데, 학생들의 컴퓨터 프로그래밍의 경험 수준에 따라 다른 컴퓨터 언어의 사용도 가능하다.

II. 컴퓨터 프로그래밍

A. 교육과정에 컴퓨터 프로그래밍 도입의 강조

Moody(1987)가 “컴퓨터 기술이 대학 준비과정에 포함되어야 하고, 선택적인 컴퓨터 프로그래밍과정이 제시되어야 한다.”라고 제안하였고, Mccoy(1987)도 컴퓨터 프로그래밍 경험이 수학 문제 해결 능력을 향상시킨다는 사실을 실험연구를 통해 밝힘으로서 “수학 문제 해결 능력을 극대화 시키기 위해서는 학교교육과정에 수학과 컴퓨터 프로그래밍 경험을 모두 포함시켜야 한다”고 학교교육과정에 컴퓨터 프로그래밍의 통합을 강조하였다. 그리고 Shafto는 “수학에 관련된 컴퓨터 프로그래밍 작성은 수학의 학습을 촉진 시킨다. 프로그래밍은 가능한 한 어린 시절부터 수학과 과학 교과 내에 통합되어야 한다. 만일 프로그래밍이 어떤 교과목의 학습을 신장시키기 위한 수단으로서 가르쳐져야 한다면 그것은 그 교과목의 내용의 복잡성 수준과 일치 하여야 한다.”고 주장하면서 수학과 과학의 교육과정에 컴퓨터 프로그래밍의 조기 도입을 강조하였다(강욱기, 1991 : 재인용).

이에 대해, 최근에는 국내에서도 강욱기(1992)를 비롯하여 수학교육에 관련된 많은 학자들이 수학교육에 컴퓨터 프로그래밍을 도입하여 교육의 효과를 높일 것을 제안하고 있다. 그러면 컴퓨터 프로그래밍을 수학 교육에 도입하였을 때 효과를 올릴 수 있는 프로그래밍의 기능은 무엇인가? 이에 대해 류희찬(1992)은 다음 세 가지의 컴퓨터 프로그래밍의 기능을 제시하고 있다.

1) 오류수정(debugging) 활동을 통해 수학적 태도를 개선 시킬 수 있다 : 프로그래밍이 수학교육에 도입되면 학생들의 오류에 대한 부담이 없어지므로 수학에 대한 태도를 개선시킬 수 있다.

2) 새로운 교과 내용으로서의 프로그래밍 기능을 생각할 수 있다 : 마치 주어진 문제를 지필로 풀게 하듯이 컴퓨터 프로그래밍을 이용하여 문제를 풀 수 있게 하는 것은 장차 도래하게 될 새로운 컴퓨터 사회에 대비시키는 첩경이다.

3) 프로그래밍 활동을 통해 사고력을 신장시킬 수 있다 : 프로그래밍 과정 그 자체가 문제 해결 과정이다. 프로그래밍을 하기 위해서는 목표를 인식하고 설계를 하며 프로그래밍을 작성하고 실행을 해야한다. 그리고 프로그래밍의 결과를 오류수정해야 한다. 이러한 단계는 Polya가 말하는 문제해결의 단계 즉, 문제의 이해, 계획, 실행, 반성의 단계와 유사하다.

B. 우리나라 현행 컴퓨터 교육 내용 사항

5차 교육과정에서는 컴퓨터 교육 강화라는 차원에서 국민학교 4학년부터 컴퓨터 내용이 간접, 직접적으로 일단 교육과정에 삼입되어 있으나, 프로그래밍 학습에 대한 내용은 중학교 과정에서 부터 포함하고 있다. 여기에서는 중학교 교육과정에서의 컴퓨터 관련 교육과정 내용을 구체적으로 살펴 본다.

중학교 컴퓨터 관련 교육과정 내용

현재의 중학교 컴퓨터 교육 목표는 현대 생활과 컴퓨터의 역할, 활용 분야 및 컴퓨터의 주요 구조와 원리를 알게 하고, 간단한 프로그램을 작성할 수 있게 하는 데 있다. 이러한 컴퓨터 교육목표를 성취하기 위한 교육과정 내용은 컴퓨터와 생활, 컴퓨터의 구성과 원리, 컴퓨터의 사용 방법으로 구성되어 있다.

중학교 실업·가정과 과목중 기술 1, 기술 2, 가정 1, 기술·가정 1과 선택 과목인 상업에 들어 있는 컴퓨터 교육과정 내용과 교과서 내용을 살펴 보면, 중학교 1학년에서 순서도와 프로그램에 대한 이론적인 내용을 학습하고 직접 간단한 프로그램도 작성해보고 있다(교육개발원, 1992).

따라서 중학교 2학년 부터는 수학 문제를 해결하거나 수학 학습 강화를 위해 컴퓨터 프로그래밍을 수학 수업에 도입하여도 그 프로그램들을 이해하고 또한 학생들이 직접 간단한 프로그램 정도는 작성할 수 있을 것으로 보인다. 즉 수학교육에 컴퓨터 프로그래밍을 도입하고자 할 때, 중학교 2학년 부터 하는게 적절하리라 생각된다.

C. 수학과 교과서에 도입된 컴퓨터 프로그래밍

1. 우리나라

현재의 중학교 수학과 교육과정 중에서 컴퓨터의 기본원리와 관련되어 있는 내용은 중학교 1학년의 기수법 단원 중 이진법 단원 한군데 인데, 그것도 단지 일반적인 위치적 기수법의 원리를 이해시켜서 십진법을 올바르게 알도록 하기 위한 목적으로 지도되고 있는 내용이다. 그리고, 고등학교용 수학과 교과서인 '수학I'과 '수학II'의 수열 단원에서 순서도 (flow chart)만을 취급하고 있을 뿐, 컴퓨터 프로그래밍을 수록하고 있는 수학 교과서는 없다.

2. 미 국

미국에서 사용되고 있는 대부분의 중학교와 고등학교 수학과 교과서들은 BASIC 프로그래밍의 응용을 지도하고 있다. 그러한 교과서를 살펴 보면, 컴퓨터 응용 단원의 구성 형태에 따라 다음 세가지로 구분하여 그 특징들을 제시할 수 있다.

첫째, 가장 일반적인 형태로서, 각 학습 단원의 마지막에 부록의 형식으로 컴퓨터 활동(COMPUTER ACTIVITY) 또는 컴퓨터 응용(COMPUTER APPLICATION) 단원을 설정하여 평균 2쪽씩을 할애하여 실고 있다.(HBJ ALGEBRA 1, HOLT ALGEBRA 1, Heath ALGEBRA 1, etc.)

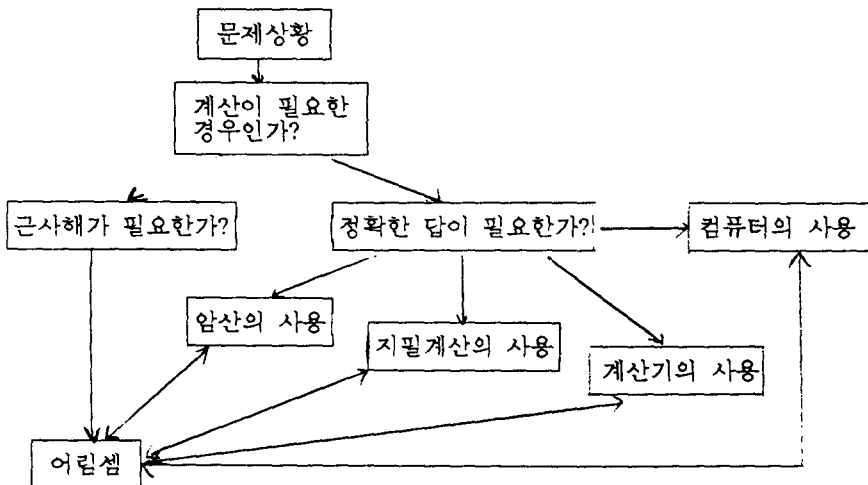
둘째, 교과서의 권말 부록으로 제시하고 있다.(Introductory ALGEBRA 1, ALGEBRA etc.)

셋째, 많은 연습 문제의 형태안에서 작은 공간을 할애하여 가벼운 보기형식 (COM-PUTER TIP)으로 제시하고 있다.(MATH is/2 etc.)

III. 수학교육과 컴퓨터 프로그래밍

A. 계산도구로서의 컴퓨터 프로그래밍

NCTM(1989)에서는 미국 수학 교육에 당면한 문제를 일곱가지로 요약하여 제시하고 있는데 그 중 한가지가 '컴퓨터와 계산기가 수학교육에 미칠 수 있는 잠재력에도 불구하고 수학 교수법에 큰 영향을 미치지 못하고 있다.'라는 점이다. 더불어 이러한 문제점을 극복하기 위한 교육 내용과 방법에서의 획기적인 방향 전환의 하나로서 '수학 수업에서 계산기와 컴퓨터의 사용이 대폭 허용되어야 한다.'라고 요청하고 있다.그러면, 어떠한 계산(numerical)문제에 컴퓨터 프로그래밍을 사용해야 하는가? 답을 구하기 위해서 계산을 해야 할 필요를 느꼈을 때, 먼저 방법을 선택하여야 하는데, 그 선택 과정을 그림으로 간단히 나타내면 다음과 같다(NCTM, 1989).



<그림> 계산(numerical) 문제의 계산 절차에 대한 판단

따라서, 수학 수업에서 계산이 필요할 때, 학생들은 근사해를 구할 것인지, 정확한 해를 구할 것인지 결정해야 하며, 또한 가장 적절한 방법을 선택하여 사용해야 한다. 학생들은 계산기와 컴퓨터를 포함하여 다양한 방법으로 계산을 하고, 적절한 과정을 선택할 수 있어야 하며 답을 찾고, 그 답의 타당성을 판단할 수 있어야 한다

이러한 내용을 기초로 하여 Cangelosi(1992)는 컴퓨터를 직접 수학 수업시간에 활용하여 본 결과를 통해 컴퓨터 프로그래밍에 대한 의견을 다음 두가지로 제시하고 있다. 첫째, 알고리즘을 실행하기 위해서 프로그래밍을 함으로써 학생들은 알고리즘에 대한 이해를 높일 수 있다. 둘째, 계산기와 같이 컴퓨터가 알고리즘을 해결하는 짐을 덜어 줌으로써 학생들은 그 시간에 그들의 정력과 시간을 좀 더 복잡한 인지적 과정에 쏟을 수 있다. 물론, 알고리즘 기능 숙달이 목표인 수업시간에는 학생들에게 손으로 알고리즘을 실행하도록 해야한다. 그러나, 응용 수준(application-level)의 수업에서는 컴퓨터를 사용해 알고리즘을 실행할 수 있도록 함으로써 알고리즘적 단계를 반복하는 것 보다는 오히려 문제를 푸는 방법에 학생들의 시간과 정력을 집중할 수 있도록 할 수 있게 한다.

B. 컴퓨터 프로그래밍 연습문제 구성 형태 및 설계

미국에서는 학습내용에 초점을 맞추고, 프로그래밍과 관계된 어려움을 줄이기 위하여 프로그래밍 연습문제가 개발되었는데, Camp & Marchionini(1984)는 이것을 PERCs(Programming Exercise Related to Contents)라고 제시하였다.

1. PERCs의 특징

- (1) 특정한 컴퓨터 언어나 설비에 제한되지 않는다.
- (2) 학생들의 수준에 따라 다양하게 적용할 수 있다.
- (3) 교사가 다양하게 변형할 수 있다.
- (4) 내용에 있어서 학생의 흥미를 끌어야 한다.
- (5) 프로그래밍보다는 수학 자체에 근본적인 초점을 맞추어야 한다 : 결국에는 프로그래밍 기술도 향상이 될지라도 수학적 개념과 기능에 대한 의미와 상세함을 강화하도록 구성되어야 한다.
- (6) 학생들이 프로그래밍에 대한 일반적인 개념을 탐구하고 발견하도록 한 다음, 표준이며 비교적 쉽게 숙달될 수 있는 일련의 명령문으로 기술을 발달시키고, 수학 학습 내용의 연구를 향상시키기 위해 적합하다고 생각이 들 때에, 고급 언어를 사용해야 한다.

2. PERCs의 구성 형태

일반적으로 사용될 수 있는 PERCs의 구성 형태는 다음 10가지로 제시할 수 있다.

- (1) 프로그램을 실행하고 결과를 논의하기.
- (2) 프로그램의 실행을 모의실험하기.
- (3) 프로그램의 특성을 설명하기 위하여 주석프로그램을 작성하기.
- (4) 주요한 명령문을 삽입함으로써 프로그램을 완성하기.
- (5) 결과를 예측하기.
- (6) 관련된 과제를 실행하도록 프로그램을 변형하기.
- (7) 오류 전달문으로부터 추론하기.
- (8) 프로그램의 오류를 발견하여 수정하기.
- (9) 프로그램등의 과정, 효율성, 결과에 관하여 비교하기.
- (10) 프로그램을 작성하고 실행하기.

PERCs의 구성 형태는 학습을 향상시키기 위하여 프로그래밍을 사용하고자 하는 교사와 학생들에게 하나의 지침서를 제공한다. 이러한 프로그래밍의 각 단계를 수학 학습 내용에 적절히 활용함으로써

써, 프로그래밍을 통한 수학적 사고를 개발할 수 있으며, 문제 해결의 도구로 활용할 수 있다. 또 각 단계를 학습함으로써 학생들은 컴퓨터 언어에 익숙해진다(박임숙, 1986).

3. PERCs의 설계(Designing)

PERCs를 만드는 세가지의 일반적인 단계를 제시하면 다음과 같다.

1단계 : 프로그램할 학습 내용의 선정

수학적 학습 내용과 컴퓨터 프로그래밍을 연결지을 수 있는 프로그램 내용을 선정한다. 학생들의 프로그래밍 경험과 언어 선택의 수준이 충분히 고려되어야 한다.

2단계 : 프로그램을 작성하기

비록 프로그램의 일부뿐만이 결과적으로 그 활동에 사용된다 할지라도, 완성된 프로그램을 작성하는 것은 PERCs의 설계에서는 필수적이다. 왜냐하면, 주어진 하나의 학습과제(task)나 문제(remark), 그리고 다른 학생들보다 몇몇 학생에게는 변형된 프로그램이 더 적절할 수도 있기 때문에 프로그램이 만들어지는 과정에서 생기는 변화 등에 따라 다양한 프로그램이 작성될 수 있기 때문이다.

3단계 : 하나의 PERC 구성 형태를 선택하고 작성하기

학생들의 프로그래밍의 경험과 PERC의 방법은 어떠한 형태를 선정할 것인가를 결정하는데 도움을 줄 것이다. 예를 들면, 프로그래밍 경험을 갖지 못한 학생들에게는 복잡한 프로그램을 작성하거나 프로그램을 광범위하게 수정한다는 등은 기대할 수 없다.

IV. 컴퓨터 프로그래밍을 활용한 수학학습자료의 실제

A. 학습문제 선정

컴퓨터의 특성때문에 중학교 수학 교과서의 모든 학습 내용에 대하여 BASIC 프로그래밍을 활용한 학습자료를 만들 수는 없다. 따라서, 먼저 중학교 2학년의 수학 교과 내용을 제 5차 교육과정에 제시된 수와 연산, 방정식과 부등식, 함수, 통계, 도형등 5개 영역으로 분류하여 영역별 학습목표를 제시한 뒤, 그러한 학습목표를 달성하는데 도움을 줄 수 있으면서 컴퓨터 프로그래밍을 활용한 학습자료를 만들기에 적절한 학습문제를 선정한다.

본 연구에서는 수와 연산 영역에서 2문제, 방정식과 부등식에서 3문제, 함수에서 2문제, 통계에서 2문제, 도형에서 3문제를 선정하여 총 12개의 학습문제를 선정하였다.

B. 컴퓨터 프로그래밍을 활용한 수학학습자료의 구성체계

여기에 제시된 수학 학습 자료의 구성체계는 미국 수학 교과서에 도입되어 있는 단원부록 제시 형태를 참조하였다.

1. 학습문제 제시

학습문제를 선정하여 제시할 때, 고려할 점은 다음과 같다.

첫째, 학습목표에 비추어 가능한 한 하나의 학습단원을 대표할 수 있는 학습문제를 선정한다 : 여기에서는 우리나라 5종 교과서에 실려 있는 보기문제와 예제문제 중 공통적인 문제나 그 단원의 학습시 중요하다고 생각되는 문제 등을 선정하여 제시하였다.

둘째, 선정된 학습문제는 컴퓨터 프로그램을 작성할 수 있는 문제이어야 한다 : 비록 학습목표에 적합하고 학습단원을 대표할 수 있는 문제라 할지라도 BASIC 프로그램으로 작성할 수 없는 문제라면 이 문제로는 컴퓨터 프로그래밍 활동을 할 수 없다. 그리고 여기에서는 중학교 2학년 과정을 위한 것이므로 학생의 BASIC 프로그래밍 경험을 고려하여 될 수 있는 한 간단한 프로그램으로도 해결될 수 있는 문제이어야 한다.

2. 예제 프로그램

주어진 학습문제에 대한 BASIC프로그램을 작성하여 제시한다. 이 경우 학생들의 프로그래밍 경험 수준이나 새로 학습되어야 할 BASIC 명령어의 종류등을 충분히 고려하여 학생들이 이해할 수 있는 프로그램을 작성하여야 한다. 따라서, 여기에서는 기본적인 BASIC 명령어 부터 내장함수의 사용까지 차례로 도입하였다(예 : ^, *, /, +, -, REM, LET, PRINT, END, INPUT, READ/DATA, GOTO, IF/THEN, FOR/NEXT, RND(X), INT(X)). 그리고, 학생들은 예제 프로그램의 각 단계를 전개해 가면서 그 속에 담긴 수학적 알고리즘을 이해하도록 한다. 이 예제 프로그램은 선정된 학습문제의 성격에 따라 완성된 프로그램을 제시하거나 불완전한 프로그램을 제시할 수 있는 데, 수학적 개념이나 원리, 알고리즘의 흐름의 이해, 논리적인 판단 등을 강화하고자 할 때는 불완전한 프로그램을 제시하였다.

3. 예제 프로그램 내용 설명

예제 프로그램에서 제시되는 각 단계의 명령문들이 학생들에게 어떻게 하도록 지시하는 것인가를 간략히 주석을 달아 준다. 이는 프로그래밍의 경험 수준이 높은 학생에게는 효용 가치가 작으나, 아직 예제 프로그램을 이해하는데 어려움을 겪는 학생에게는 도움을 줄 것이다. 그리고 새로 나오는 BASIC 명령어에 대한 이해를 하는 데도 도움을 준다.

4. 연습문제

주어진 예제 프로그램을 활용하여 다양한 수학 문제를 풀 수 있도록 구성한다. 이 연습문제의 특징을 다음 네 가지로 제시할 수 있다.

(1) 문제의 구성형태는 앞 장에서 제시한 PERCs의 10가지의 구성형태를 학습문제에 맞게 적절히 사용하였다. 그 외에 수학적 개념이나 알고리즘의 이해를 묻는 질문도 다양하게 제시된다.

(2) 수학 학습 효과를 높이는 데 주안점을 두고 구성하였다 : 복잡한 프로그램을 작성하는 능력보다는 수학 학습 문제를 해결하는 데 있어서 컴퓨터를 이용할 수 있도록, 간단한 컴퓨터 프로그래밍을 활용할 수 있는 능력이 향상되도록 한다.

(3) 예제프로그램에 제시되어 있는 학습문제와 유사한 많은 문제를 제시하여 주었다 : 예제프로그램의 변수 값에 다른 여러가지 수 값을 대입하여 유사한 많은 문제를 풀어 보도록 함으로써 적은 시간 내에 풍부한 문제 경험을 쌓을 수 있다.

(4) 컴퓨터가 없어도 사용 가능하다 : 컴퓨터가 없어도 주어진 프로그램의 알고리즘을 따라 지필계 산이나 암산으로 주어진 학습문제들을 대부분 해결할 수 있다. 그러나 만약, 컴퓨터를 사용할 수 있는 환경이 구축되어 있다면 직접 컴퓨터로 실행하여 봄으로써, 즉각적으로 결과를 컴퓨터 화면을 통하여 확인하여 볼 수 있다. 이는 이 학습자료의 효과를 더 높게 할 것이다.

C. 컴퓨터 프로그래밍을 활용한 수학학습자료의 실제

여기에서는 본 연구자가 만든 12개의 학습자료 중에서 1가지만 제시한다.

단원 : 통계

확률에 대하여 모의실험 하기

주머니 안에 9개의 구슬이 들어 있는데, 붉은 구슬이 4개, 흰 구슬이 3개, 푸른 구슬이 2개가 들어 있다. 이 주머니에서 1개의 구슬을 꺼낼 때, 붉은 구슬이 나올 확률, 흰 구슬이 나올 확률, 푸른 구슬이 나올 확률을 각각 구하라.

(확률은 $P(\text{붉은 구슬}) = 4/9 = 0.44$, $P(\text{흰 구슬}) = 3/9 = 0.33$, $P(\text{푸른 구슬}) = 2/9 = 0.22$ 이다. 이 모의실험에서는 30번을 반복해서 시행한다.)

프로그램

프로그램 설명

```

10 REM 이론적 확률에 대한 모의실험
20 R=0:W=0:B=0           R,W,B의 초기값을 0으로 설정한다.
30 FOR I=1 TO 30         루프의 초기값을 1로 놓고 시작한다.
40 LET X=RND(1)         X에 0과 1사이의 난수를 할당한다.
50 IF X<=4/9 THEN R=R+1:GOTO 80   참이면 R의 값에 1을 더하고 80문 실행.
60 IF X<=7/9 THEN W=W+1:GOTO 80   참이면 W의 값에 1을 더하고 80문 실행.
70 IF X>7/9 THEN B=B+1         참이면 B의 값에 1을 더하고 다음문 실행.
80 NEXT I               I의 값이 30이 될 때까지 루프를 반복 실행
90 PRINT "구슬 종류   뽑힌 횟수   확률(%)"   행한다.
100 PRINT "붉은 구슬   ";R;"   ";R/0.3;"%"   구슬 종류, 뽑힌 횟수, 확
110 PRINT "흰 구슬     ";W;"   ";W/0.3;"%"   률을 각각의 경우에 맞게
120 PRINT "푸른 구슬   ";B;"   ";B/0.3;"%"   출력한다.
130 END

```

연습문제

1. 위의 프로그램은 주머니 안에서 구슬을 한번에 하나씩 반복적으로 30회를 추출하는 모의실험을 하기 위한 프로그램이다. 위의 프로그램을 수행하여라. 그 결과들은 확률과 비교하여 어떠한가?
2. 위의 프로그램을 구슬이 들어 있는 주머니에서 300회를 추출하는 경우로 바꾸어라(30,100,110,120문을 변경). 그리고 이 프로그램을 수행하라. 표본의 수가 300개인 경우의 결과와 표본의 수가 30개인 경우의 결과를 확률과 비교하여 보면 어떠한가?
3. 표본의 수가 900개인 경우의 결과를 예측해 보아라. 프로그램을 이 문제의 경우에 맞게 바꾸어서 (30,100,110,120문을 변경) 수행해 보고 자기의 예측이 맞는지 확인하여라.
4. 문제 1에서 문제 3까지의 실행 결과를 통해서 어떠한 사실을 알 수 있는가? 자신의 생각을 기술하여라.

BASIC 기호 : RND(1)

검토 및 의의 : 어떤 시행에서 일어날 수 있는 모든 경우의 수가 n 이고, 각각의 경우가 일어날 가능성이 같다고 할 때, 어떤 사건 A 가 일어날 수 있는 경우의 수가 a 이면, 그 사건 A 가 일어날 수 있는 가능성을 나타내는 수 값 a/n 을 사건 A 가 일어날 확률이라고 한다. 시간적인 제한 때문에 이러한 확률을 교실에서 직접 실험을 통하여 확인해 보는 것은 어렵다. 따라서 실제 교육 현장에서는 확률에 대한 정의를 이론적으로만 설명하여 학생들에게 이해시키는 경향이 있다. 그러나 컴퓨터를 통하여 모의실험을 함으로써, 실험의 회수를 얼마든지 크게 할 수 있으므로 적은 시간 내에 학생들에게 이론적인 확률의 정당성을 실험을 통하여 이해, 강화시킬 수 있다.

1. 주머니에서 구슬을 꺼내는 모의실험을 컴퓨터를 통하여 30회를 실시한다. 이 때, 학생들은 이 결과가 확률과 같은가를 비교하여 본다.
2. 주머니에서 구슬을 꺼내는 모의실험을 컴퓨터를 통하여 300회를 실시한다. 이 때, 학생들은 확률과도 비교하여 보고, 문제 1에서의 실험 결과와도 비교하여 본다.
3. 문제 1과 문제 2의 경험을 통하여 모의실험을 900회 실시할 경우의 결과를 미리 예측하여 본 뒤, 컴퓨터를 통하여 모의실험을 900회 실시한다. 그리고 출력된 결과를 자신의 예측 결과와 비교하여 본다.
4. 문제 3의 결과를 통하여 실험의 횟수가 커지면 커질수록 컴퓨터에 의해 계산된 확률의 값은 이론적인 확률에 가까와 짐을 확실히 알 수 있다.

V. 요약 및 제언

컴퓨터가 만들어진 이후로 교육계의 많은 연구자들은 컴퓨터를 교수-학습에 활용하여 학습 효과를 높이기 위해 많은 노력을 하여 왔다. 최근에는 많은 교육자들이 컴퓨터 프로그래밍 교수에 대하여 다시 강조하기 시작하고 있고, 국내·외의 여러 수학 교육자들이 수학 문제 해결력이나 수학적 사고력을 향상시키기 위하여 수학교육에서도 컴퓨터 프로그래밍을 도입할 것을 강조하고 있다. 이미 미국과 일본등은 수학과 교과서에 컴퓨터 프로그래밍을 도입하여 수학문제를 해결하기 위하여 컴퓨터 프로그래밍을 활용할 수 있도록 지도하고 있다.

우리나라에서도 앞으로 실시될 제 6 차 교육과정에서 수학의 교수-학습에 있어서 컴퓨터와 계산기의 활용을 더욱 강화하고 있다. 특히, 몇몇 수학 교육 연구자들은 수학교육과정에 컴퓨터를 도입할 수 있는 여러가지 방법 중에서 컴퓨터 프로그래밍을 도입해야 한다고 주장하고 있다. 그러나, 이러한 많은 요구에 반하여 국내에서는 이 분야에 대한 연구내용이 빈약하고 어떤 학생들에게 어떤 수준의 컴퓨터 프로그래밍을 지도하고 활용할 것인가를 제시하는 자료가 전무한 실정이다.

따라서, 본 연구에서는 수학교육과정에 컴퓨터 프로그래밍의 도입을 강조하는 관점에서 미국의 수학 교과서에 도입되어 있는 컴퓨터 프로그래밍의 내용 및 수준과 구성 형태등을 분석하였다. 그리고 그것을 토대로 우리나라 중학교 2학년 수학교육과정의 학습 내용에 알맞게 컴퓨터 프로그래밍을 활용한 수학학습자료를 개발하여 제시하였다. 그러나 여기에 제시한 학습자료는 단지 몇가지의 자료만을 토대로 개발한 것이기에, 교육 효과에 대한 충분한 근거 자료를 제시할 수 없었음을 밝히며, 실용화를 위해서는 보다 많은 자료의 분석과 여론 수렴 연구가 있어야 할 것이다.

마지막으로 본 연구자가 지지하고 있는 학교 수학교육과정에서의 컴퓨터 프로그래밍 활용이 실제 학교 현장에서 그 기대하는 효과를 나타내기 위해서 다음과 같은 몇 가지 제언을 하고자 한다.

1. 컴퓨터 프로그래밍 활동을 도울 수 있는 내용이 수학 교과서에 도입되어야 한다.

수학 교육에서 특정한 수학 문제를 해결하기 위해 컴퓨터 프로그래밍의 활용을 강조하더라도 이에 대한 내용이 직접적으로 수학 교과서에 쓰여지지 않는다면 그 효과는 기대하기가 어렵다. 따라서 컴퓨터 프로그래밍을 활용하고자 하는 교사나 학생들을 위하여 컴퓨터 프로그래밍이 활용된 수학학습자료가 수학 교수-학습의 직접적인 매체인 수학 교과서에 도입되어야 한다고 생각된다. 단, 도입하는 방법에 있어서는 많은 연구가 필요하리라 생각된다.

2. 수학 교육에서 컴퓨터 프로그래밍 활용의 근본적인 목적은 컴퓨터와 관련된 경험을 통하여 수학적 개념과 기능을 개발하는 데 두어야 한다.

물론, 수학 교육에서 컴퓨터 프로그래밍 활용을 강조하는 주요한 목적 중의 하나는 현대의 정보화 사회의 요구에 부응하여, 교육 과정에서 직면하는 여러 수학 문제를 컴퓨터를 이용하여 해결할 수 있는 능력을 향상시키는 데 있다. 그러나 주의할 점은, 수학과에서의 컴퓨터 프로그래밍은 수학적 능력 신장을 돕기 위한 한 수단일 뿐, 그 자체가 수학 교육의 목표가 되어서는 안된다.

3. 가능한한, 수학 교육 과정에서 도입할 수 있는 컴퓨터 프로그래밍은 특정한 컴퓨터 언어나 하드웨어(Hardware)와는 독립되어야 한다.

여기에서는 BASIC 프로그래밍을 활용하였다. 그러나 최근에 학생들의 인지능력이나 문제해결력 신장에 많은 도움을 줄 수 있는 언어로써, LOGO에 대한 연구가 활발하게 일어나고 있다. 따라서, BASIC에만 국한 되지 않고 수학적 효과를 올릴 수 있는 다양한 컴퓨터 언어를 활용해야 할 것이다.

4. 수학 교육에 컴퓨터 프로그래밍을 활용하는 데 있어서 안내서가 될 컴퓨터 프로그래밍을 활용한 수학학습자료가 다양한 수학 학습 주제에 초점을 맞추어 개발되어야 한다.

진실로 이러한 컴퓨터 프로그래밍의 활용을 강화하고자 한다면, 교사와 학생들에게 수학 학습에 초점을 맞춘 이러한 컴퓨터 프로그래밍 활동을 하도록 하는 것 뿐만 아니라, 필요한 경우에 적절히 사용할 수 있는 보충교재로서 컴퓨터 프로그래밍을 활용한 수학학습자료를 다양하게 개발하여야 할 것이다. 더불어 이러한 자료를 적절히 교수-학습에 활용할 수 있도록 활용방법에 대한 안내서 및 지침서에 대한 개발도 뒤따라야 할 것이다.

참 고 문 헌

- 강옥기.(1991). 수학 교과서와 BASIC 프로그래밍. 제 8회 수학교육학 세미나. 인천 교육대학 부속 과학교육연구소 수학교육학 세미나 그룹.
- 김창동.(1987). 마이크로 컴퓨터를 활용한 수학 교수 학습법 개발에 관한 연구. 한국교원대학교 수학교육과 석사학위 논문.
- 류희찬.(1992). 수학과 교육과정의 「내용」 개정 방향에 대한 소고. 靑藍數學教育 第 2輯. 한국교원대학교 수학교육연구소.
- 박임숙.(1986). Personal Computer 를 통한 수학적 사고 교육에 관한 연구. 서울대학교 수학교육과 석사학위 논문.
- 이종락 외.(1991). 고등학교 수학과 교육과정 개선 방향에 관한 연구. 제 8회 수학교육학 세미나. 인천교육대학 부속 과학교육연구소 수학교육학 세미나 그룹.
- 이태욱.(1991). 컴퓨터 교육과 관련된 교원 양성 대학의 교육과정 실태 분석과 초·중등학교 컴퓨터 교육 환경 조사 분석에 관한 연구. 21세기에 대비한 초·중등학교 컴퓨터 교사 교육의 방향. 한국교원대학교.
- Bright, G.W., & Cliffliao Y.K.(1991). Effects of Computer Programming on Cognitive Outcomes: A Meta-Analysis. In R.H.Seldman(Ed.), *Journal of Educational Computing Research*, p.251-268.
- Camp J.S., & Marchionini G.(1984). Programming and Learning: Implication for Mathematics Education. In M.J.Zweng(Ed.), *Computers in Mathematics Education*. NCTM 1984 Yearbook, Reston, Virginia, NCTM.
- Cangelosi, J.S.(1992). *Teaching Mathematics in Secondary and Middle School : Research-Based Approaches*. Macmillan, Inc. p.47.
- Hatfield, L.L. & Kieren, T.E.(1972). Computer Assisted Problem Solving in School Mathematics, *Journal for Research in Mathematics Education*, pp.99-112.
- Mccoy, L.P.(1987). *The Effect of Computer Programming Experience on Mathematical Problem Solving Ability*. Unpublished Doctoral Dissertation : University of Virginia
- Moody, M.(1987). *Topics Which Should Be Included in College Preparatory Mathematics Courses*. Unpublished Doctoral Dissertation: University of Auburn.
- NCTM.(1989). *Curriculum and Evaluation Standards for School Mathematics*. Reston, VA : The National Council of Teachers of Mathematics, INC.
- Shumway, R.J.(1984). Young Children, Programming and Mathematics. In V.P. Hansen & M.J.Zweng(Eds.), *Computer in Mathematics Education*. NCTM 1984 Yearbook, Reston, Virginia, NCTM. p.127-134.