

중등 수학 및 컴퓨터과학 교육에서 컴퓨터 교육의 연결성 및 통합성¹⁾

정 치 봉 (순천향대학교)

수학 교육에서 새로운 정보기술의 교육적 사용의 범위는 커지고 교육 모형은 다양해지고 있다. 특히 중등 수학교육에서의 변화에 대한 기대도 커지고 있다. 본 논문은 미국 및 한국의 수학 교육계의 폭넓은 지지를 받는 NCTM의 컴퓨터 및 정보기술 활용에 관한 기준들의 교육적 의미, 방법, 효용 등을 정리하였다. 이어서 컴퓨터과학 교과과정에 포함된 수학 분야에 대한 컴퓨터과학자 Hamming의 견해를 소개하였다. 본 논문은 한국의 중등 교육에도 컴퓨터과학 교과가 시행될 것을 예상하면서 수학 교사들이 미리 컴퓨터과학을 교과과정을 이해함으로써 통합 교과적 요소, 특성 그리고 역할 관계를 알아보았다.

I. 서론: 학교 수학 교육과 정보기술

정보기술 시대, 지식정보사회, 지식경제와 경영, 전자상거래 그리고 수 없이 다양한 사이버 세계들이 20세기를 뒤로하고 새로운 세기와 새 천년에 임하는 현재의 사회를 표현하는 용어들이다. 이 시점에서 결론이며 실천해야 할 명제는 “학교 수학 교육은 변해야 한다”이다.²⁾

7차 교육과정 개편 안이 나와 있지만 학교 수학의 변화를 수학 교육 과정 안에 모두 담을 수 없으며 그리고 담길 수 없다. 수학 교육도 화폐와 같이 유동적인 것이다. 수학 교육이 활력이 넘치는 역동적인 교과가 되어야 한다.

한국에서는 특히 대학입시 제도로 인하여 중등 수학 교육이 독립적이지 못하고 종속되어 왔다. 이러한 종속성을 탈피하기 위하여 수학이 유클리드 기하에서 비유클리드 기하로의 해방으로 시작하여 20세기에 와서는 수학의 거의 전 분야가 해방되어 발전하고 있는 현재의 다양한 수학의 모습을 상기할 필요가 있다.

본 논문은 컴퓨터 및 정보기술을 수학교육에 적극적으로 수용함으로써 한국 수학 교육의 발전, 변화 그리고 성장을 할 것이라는 기대로 수학과 컴퓨터과학 사이에 연결된 여러 의미를 찾아보았다. 우선 미국의 컴퓨터 및 정보기술 자원 및 초고속 정보 통신망 등이 학교, 교실 도서관에 보급되고 활용하는 국가적 추진 양상을 소개한다. 그리고 미국의 수학 교육에서 컴퓨터 및 정보기술의 교육적 활용 모습과 방향 등을 NCTM을 중심으로 이루어지고 있는 내용을 소개한다. 일리노이 대학에서 운

1) 이 연구는 1998학년도 순천향대학교 기초과학연구소 지원에 의한 결과임.

2) 구광조·강완 (1996). 학교 수학의 재구성, 한국수학교육학회 연구 자료 96-II.

영하는 웹을 통한 수학교사 재교육 프로그램을 통하여 수학 교사가 전문 직업인으로 가져야 할 정보 기술과 수학교육 소프트웨어를 소개한다.

양 교과와 속성으로 볼 때 수학교과와 컴퓨터 교과 사이의 구별이 학교 교육에서 점점 허물어지고 통합하여 가게될 것 같다. 이유는 교육 수요자인 학생, 학부모 그리고 산업체의 수학 교육에 대한 요구를 반영하지 않을 수 없다. 수학교육은 플라톤의 이상주의적 교육관에 고요히 있을 수 없어 보인다.

미국을 비롯한 선진국에서는 수학 교과 교육에 컴퓨터 활용을 통합해 가고 있다. 이러한 변화를 대비한다면 수학교사는 컴퓨터과학과 수학의 관계를 이해할 필요가 있다. 컴퓨터과학은 이론적 기초로서 순수수학 및 응용수학 분야 모두를 갖고 있다. 본 논문에서는 현시점에서조차 여전히 유효한 의미를 가지고 있는 Hamming의 1968년도 컴퓨터 과학 교과과정에 필요한 수학에 대한 견해를 소개하였다. 그리고 컴퓨터과학이 수학에서 기술과 공학적 성격을 갖는 차이를 소개하였다. 한편 현재 대학 수학 전공 교과과정의 양상 그리고 수리과학의 양 측으로서 수학과 컴퓨터과학에 대한 전망을 소개한다.

미국 일리노이대 수학과에서 운영하는 웹 기반 수학교사 재교육프로그램의 모듈들이 수학 수업에 사용 가능한 Mathematica, Logo, GSP, JAVA, HTML 작성과 같은 정보기술이다. ACM/IEEE-CS가 권고한 미국의 고등학교 컴퓨터교육 과정과 Logo를 중심의 교과과정 예시를 소개하였다. 수학적 사고와 이해를 위하여 컴퓨터 및 정보기술은 필요하다. 학생의 지적 성장에 따라 어떻게 자연스럽게 교육과정을 구현할 수 있는가에 대하여 관련 국가 정책, 수학 교육계의 주장과 기준, 강의안, 교사재교육 프로그램의 방향을 알 수 있는 자료를 제시하고자 하였다.

II. 미국의 학교 교육에서 컴퓨터 및 정보기술

미국은 언제, 어디서나 최상의 교육을 받을 수 있도록 미국의 모든 교실, 도서관, 박물관 사무실, 연구실을 초고속 정보통신망에 연결시키고 첨단 정보기술을 통한 교육 개혁을 추진하고 있다. 2000년까지 초고속 정보통신망에 의한 평생 학습 사회의 기반을 조성하고자 한다.³⁾

클린턴 대통령은 학부모, 교사, 정부, 지역사회, 사업자들이 미국의 모든 어린이들이 과학 기술적 탁월한 소양을 지닐 수 있도록 함께 노력할 것을 제안하고 과학기술 교육의 4가지 목표를 세웠다.⁴⁾

1. 미국의 모든 학교와 교실을 정보망에 연결한다. 모든 교실을 학교망과 인터넷에 연결한다. 학교망과 인터넷은 컴퓨터를 강력한 학습 도구로 만들 것이다.

3) USACNII. (1996). *KickStart initiative: Connecting America's communities to the information superhighway*. St. Paul, MN: West Publishing Company.

4) 강오한 (1998). 미국의 컴퓨터 교과교육 연구 실태조사 연구, 교원대 연구과제 RR 97-V-3.

2. 모든 학생과 교사들에게 현대적인 컴퓨터를 제공한다. 모든 학생들이 현대 컴퓨터와 학습 도구들을 접할 수 있도록 한다.
3. 효과적이고 믿을 수 있는 소프트웨어 및 온라인 학습 자료를 개발하며 교육용 소프트웨어가 교과 과정의 한 부분을 형성하도록 한다. 2만개 이상의 교육용 소프트웨어 타이틀이 개발되어 있고, 매년 백만 이상의 학생들이 원격 학습망을 통해 강좌를 듣고 있으며 인터넷의 WWW에는 매일 수백 개의 새로운 홈페이지가 생기고 있다. 이들 자원은 학습을 향상시키고 문제 해결, 연구, 자료 분석 등을 돕는다.
4. 모든 교사들이 컴퓨터와 정보망을 통해 학생들의 학습을 도울 수 있도록 교사 연수와 지원을 제공한다. 교사는 과학기술을 사용하고 가르칠 준비가 되어 있어야 한다. 미국 과학 기술 평가(The Office of Technology Assessment)에 의하면 과학기술이 학교에 효과적으로 융합되기 위해서는 5년이 소요될 것이라고 예상된다. 그 기간 동안 교사는 훈련 및 지원을 받아야 한다.

새로운 과학기술을 활용한 교육이 효과를 내기 위해서는 교사들이 과학기술을 교육에 이용하는 창조적인 새 방법을 발견하고 교육 부분에서 새로운 교육 도구 및 콘텐츠 개발을 촉진하여야 한다. The National School Boards Association(전미 학교 연합 위원회), The National PTA(전미학부모교사연합), The National Education Association(국제 교육연합), The American Federation of Teacher(미국교사연합)와 같은 학부모 및 교육단체 그리고 수많은 사업체들이 정보과학기술을 교육에 사용하여 이해를 증가시킬 수 있는 방법에 대해 교사들과 함께 연구하고 있다.⁵⁾

Hacker(1998a)는 과학기술 교육 활동의 기준, 과학기술 교육 활동의 설계 등에 대해 다음과 같이 제시하고 있다.

- 1) 과학기술 교육 활동은 중요한 아이디어들을 포함해야 하며 2) 학생들과 관계 있는 흥미있는 내용이어야 하며 3) 하나의 모델 또는 생산물의 조립 등과 같이 생활과 관련이 있는 실제적 내용으로 구성하고 4) 적은 비용으로 학교에서 완성할 수 있어야 하며 5) 목표가 분명하게 진술되어야 하고 6) 학생들로 하여금 자신들의 연구 과정 및 결과를 문서화하도록 하고 7) 과학적인 질문과 수학적 분석을 위한 기회를 제공해야 하며 8) 다른 상황의 문제로도 해결 방법이 전이될 수 있어야 한다

컴퓨터 교육은 컴퓨터와 교육이 완전한 하나를 이루는 방향으로 가고 있다. 수학 또는 다른 교과 의 한 단원이 아니라 각 교과 학습 환경에 컴퓨터를 완전히 통합하는 것이다. 그러한 통합은 학생들에게 많은 이점을 줄 수 있다. 대화식의 사용자 환경을 가진 교육용 소프트웨어는 기존 방식의 수업을 도와주거나 대체할 수 있고 사무용과 같은 일반 응용소프트웨어들은 학생들이 연구, 작문, 발표, 보고 등의 프로젝트를 수행하는 것을 도울 수 있다.

정보기술은 발전하며 진화한다. 진화하는 정보기술을 지속적으로 사용하기 위하여 컴퓨터 및 정보 기술의 일상적인 사용은 중요한 의미를 갖는다. 정보기술에 대한 소프트웨어적 측면에 대한 이해는

5) (<http://www.ed.gov/updates/PresED Plan/part11.html>, 1997).

체계적인 컴퓨터 및 정보기술에 대한 학교 교육으로 실현될 수 있을 것이다.

Ⅲ. 수학 교육에서 컴퓨터 및 정보기술의 활용

[NCTM의 수학교육의 변화 방향과 정보기술]

미국 수학 교사 협의회(NCTM)의 여러 보고서는 다음과 같은 수학 교육에 관한 철학, 교육과정, 변화의 방향과 내용을 90년대부터 10여년 동안 일관된 견지에서 컴퓨터 및 정보기술을 사용하는 수학 교육의 입장을 유지하고 있다. 본 장은 NCTM의 여러 보고서에 기초하여 수학교육에 있어서 컴퓨터 및 정보기술에 관련된 미국 수학교사 협의회와 NCTM의 중요한 주장 및 기준들을 정리한 것이다. [NCTM 1980, 1989, 1991, 1995, 1998]⁶⁾

수학 교육에 컴퓨터 및 정보 기술을 사용하려는 노력과 변화의 양상은 매우 다양하다. 컴퓨터 및 정보기술이 수학을 사용하는 방법을 변화시키고 있다. 따라서 수학 교육 프로그램의 내용과 수학의 교수 학습 방법의 변화로 이어지고 있다. 학생과 수학교사가 받아들이고 적응하여야 하는 측면을 다음과 같이 보고 있다.

학생들은 졸업 후 사회에서 직업적인 활동을 할 때 요구되는 적합한 수학 내용을 학교 수학 학습을 통하여 습득하고 있어야 하고 지식정보 사회, 전자-교육, 전자-상거래 시대에 필요한 수학을 평생교육의 방식으로 계속해서 공부해야 한다고 보고 있다. 또한 학생은 수학을 공부할 때 언제 어떻게 컴퓨터와 정보기술을 효과적으로 그리고 실제적인 기술로서 사용해야 하는지를 알 수 있어야 한다.

교사들은 학생들의 수학에 대한 이해를 발전시키고 확장시키기 위해 컴퓨터 및 정보기술 도구를 교수-학습 방법을 증진시키기 위하여 언제 어떻게 적절히 사용해야 하는지를 알아야 한다.

컴퓨터 및 정보기술은 수학의 학습, 지도, 그리고 평가를 향상시키고 증대시키기 위해 다양한 방법으로 사용되어질 수 있는 적절한 도구라는 것이 NCTM의 공식적인 입장이다. 그러므로 NCTM은 수학과 교육과정, 교수방법, 컴퓨터 및 정보기술에 관한 하드웨어와 소프트웨어에 대한 접근 방법, 전자-교육에서 콘텐츠의 중요성에 대한 인식, 수학 교사의 수학 교육 전문가로서 평생교육 방식의 재교육, 그리고 교사의 평가 등에 있어서 변화와 발전을 추구하고 있다.

정보통신기술은 사회를 빠르게 변화시키고 자유 경쟁 원리로 움직이게 한다. 따라서 수학 교육은 사회의 실용적 가치와 요구를 반영함으로써 개방되고 교육적 다양성이 증가하는 쪽으로 변해야 한다고 보고 있다. 수학 교육과정에서 더 이상 유용하지 못한 단순히 수학 지식의 계층적 구조를 반영하는 내용을 삭제하고, 새로운 중요성이 부각되는 주제를 새로 첨가하며, 여전히 중요하다고 판단되는 주제는 존속 및 발전시키는 방향으로 개정되어 가야 한다. 중등수학 교과는 양과 질 그리고 주제의 다양함을 추구하는 것으로 보인다.

교육과정을 개정하는데 있어서, 교육자들은 각 주제에 할당되어지는 시간과 강조 정도가 정보기술

6) <http://www.nctm.org/>.

도구에 대한 접근이 증가되는 시기에 있어서의 중요성과 일치되도록 해야 한다. 컴퓨터를 활용한 교수 자료는 모든 단계의 학생들을 위해 계속해서 개발되어야 한다.

교사는 학생들이 개념을 탐구하고 발견하는데, 구체적 경험을 추상적인 수학적 아이디어로 변환하고, 기능을 연습하며, 문제를 해결하는 과정을 돕기 위한 도구로써 컴퓨터를 이용해야 한다. 수학교육에서 컴퓨터는 수업에 통합된 도구이다. 단순한 컴퓨터 사용 기술을 훈련하는 그 자체가 수업의 목적이 되어서는 안 된다. 마찬가지로, 수학수업에서 컴퓨터 프로그래밍이 수학 활동을 자극할 수 있는 그리고 수학의 활용과 이해를 증진하는 교육으로서 역할을 다 할 수 있어야 한다.

프로그래밍이 수학의 이해를 높일 수 있어야 한다. 특히 프로그래밍이 문제해결 학습과 연결되기 위하여 다양한 연습 문제, 자료 그리고 프로젝트형의 실제 과제를 제시해 줄 수 있어야 한다.

수학 교사는 프로그래밍 언어를 아는 수준을 넘어서 프로그래밍의 실제적인 기술을 가지고 학생들과 함께 프로젝트형의 과제를 도와 줄 수 있는 수준의 능력이 요구된다.

학생들은 프로그래밍을 통하여 자신의 프로그래밍 스타일을 형성해 가는 것이다. 학생 자신이 스스로 좋은 프로그래밍 스타일을 형성해 가는 것은 프로그래밍 수업의 소홀히 할 수 없는 질적인 측면으로서 프로그래밍 교육이 추구하는 창조성 개발 교육의 부분을 한 부분을 이룬다. 프로그래밍 수업이 단순한 구문 타이핑 연습 시간이 되어서는 안 된다. 프로그래밍은 자기 스타일을 형성해 가는 창조적 활동임을 보여 주어야 한다.

수학 교육에 필요한 프로그래밍 언어는 교사가 분별 있게 도입하여야 하고 수학교육의 측면을 충분히 고려하여야 한다. 수학을 학습하는 학생들이 소비하는 수업시간의 양은 컴퓨터 프로그래밍 활동을 통해서 수학을 이해하는데 도움이 되도록 계획한 결과와 일치되도록 할 수 있어야 한다.

컴퓨터 실험실에는 수학적 주제를 개인별이나 집단별로 탐구하려는 모든 학생들이 이용 가능하도록 해야 한다. 학교에는 수학 교과를 교수-학습을 하는데 있어서 일관성(호환성, 지속성, 일치성) 있게 사용되어야 할 양질의 풍부한 소프트웨어와 컴퓨터, 그리고 주변기기가 구비되어 있어야 한다. 학교는 모든 수업과 컴퓨터 실험실에서 사용되는 하드웨어와 소프트웨어를 구입, 유지, 그리고 새로운 것으로 교체할 수 예산을 마련해 두어야 한다.

모든 수학 예비교사와 현직 교사들은 수학 지도에 있어서, 컴퓨터의 사용에 대한 교육을 받아야 한다. 교사들은 학생과 컴퓨터 그리고 교사 사이의 원만한 상호작용을 위하여 컴퓨터를 활용한 교실 수업과 실험실 수업을 설계할 준비가 되어 있어야 한다. 수학교사는 시뮬레이션, 데이터의 생성과 분석, 문제해결, 그래프 해석, 그리고 연습과 같은 다양한 활동을 가능하도록 하는 코스웨어를 선별하여 사용할 수 있는 안목이 있어야 한다.

수학교사는 수학수업에서 프로그래밍 언어와 스프레드시트와 같은 컴퓨터를 활용할 수 있는 도구(tools)를 다양한 수업 및 교육 상황에 적절하게 사용할 수 있어야 한다. 예를 들면, 컴퓨터 프로그램은 학생의 통찰력을 더욱 증대시킬 수 있는 소재이므로 교사는 수업에 연관된 알고리즘(algorithm)을 소개하는 수학적 주제를 선정할 수 있어야 하며, 사용하는 학습용 프로그램을 학급이나 개인의 요구

에 맞도록 개발하거나 변경할 수 있어야 한다. 수학교사는 정보기술의 진보와 보조를 같이 하면서 가장 효율적이고 효과적인 방식으로 컴퓨터를 사용할 수 있어야 한다.

IV. 일리노이대 수학과와 웹 기반 수학 교사 재교육 프로그램

컴퓨터 및 정보기술을 수학 교육에 도입하려는 NCTM의 다양한 견해들은 수학 교육을 강화하고 활기차게 하려는 노력임에 틀림없다. 이러한 주장 하나 하나는 전통적으로 수학을 가르쳐 왔던 방법과 커다란 차이가 있다. 수학을 이론으로서 뿐만 아니라 컴퓨터라는 도구를 사용하여 탐구하는 수학 교육이므로 교사의 전문성 향상과 정보기술 시설투자, 효율적 운영 등 매우 복잡한 교육자원 운영 기술이 요구된다. 우선 수학 교사의 전문성을 높이기 위한 미국에서의 교사 재교육의 모습을 구체적인 사례를 통하여 알아볼 필요가 있다.

일리노이 대학 수학과에서 운영하는 웹을 통한 수학교사 재교육 프로그램 Math Teacher Link⁷⁾을 보면 수학 교사가 전문 직업인으로 가져야 할 지식의 범위가 수학에 머물지 않음을 볼 수 있다. 수학교사는 수학지식과 함께 수학을 하게 하는 소프트웨어 및 필요한 정보기술에 대하여 상당한 지식과 기술을 가져야 한다. 따라서 재교육 프로그램은 이러한 요구사항을 충족시키기 위하여 새로운 정보 기술을 실제로 수업을 통한 교육에 사용할 수 있는 지식과 기술(arts)을 중심으로 운영된다. 교사 재교육 모듈을 그대로 소개하면 다음과 같다.⁸⁾

- Module#1 Calculus & Mathematica for mathematics teachers
- Module#2 Using Internet Resources for High School Mathematics Instruction
- Module#3 Using Mathematica in the Mathematics Classroom
- Module#4 Using the Geometer's Sketchpad
- Module#5 Algebra Through Modeling with the TI-82,83 Graphing Calculators
- Module#6 Teaching Statistics in High School
- Module#7 HTML Programming for Teachers
- Module#8 JavaScripting for Teachers
- Module#9 Logo Programming for the Mathematics Classroom
- Module#10 Dynamic Geometry with Geometer's Sketchpad
- Module#11 Discrete Dynamic Systems for Mathematics Teachers
- Module#12 Computers and Connections

이러한 수학 교사 재교육 프로그램의 주제를 보면 한국의 수학 교사 재교육 프로그램과 차이를

7) (<http://www-cm.math.uiuc.edu/MathLink>)

8) (<http://mtl.math.uiuc.edu/index.htm>)

느낄 수 있고 수학 교사들이 새로운 정보기술을 받아들여야 한다는 적극성과 수준 높은 수학 교육을 향한 열정 같은 무엇을 느낄 수 있다. 새로운 관점에서 수학 교과 교육의 영역을 수학 교사들이 확대하고 있는 느낌을 갖는다.

Taylor(1980)는 수업에서 컴퓨터의 활용 방법을 교사(tutor), 수업대상(tutee), 도구(tool)의 세 가지로 분류해서 제시하였다. 위의 수학 교사 재교육에 나타난 소프트웨어인 Mathematica, Geometer's Sketchpad(GSP), LOGO, TI Graphing Calculators는 수업대상(tutee) 또는 수업 도구(tools)에 해당한다. 오늘날의 고급 교육용 소프트웨어는 프로그램을 작성하여 컴퓨터에게 수학적 작업을 수행하게 할 수 있으므로 도구(tool)와 학습 대상(tutee)이 함께 있는 소프트웨어이다. 특히 웹 기반 교육에서 작동하는 여러 정보 기술 자원은 1980년의 Taylor의 컴퓨터의 활용 특성에 대한 구분을 무의미하게 한다.

컴퓨터를 수업대상(Tutee)으로 사용하는 방식으로 프로그래밍 교육을 들 수 있다. 컴퓨터 프로그램은 컴퓨터가 프로그램에 쓰여진 지시어를 수행하게 하는 것이다. 따라서 컴퓨터와 관련된 단순 기능 교육보다도 학생이 작성한(창조한) 컴퓨터 프로그래밍에 의하여 가능한 세계, 원리 그리고 현상을 경험하는 교육이 중요한 것이다. Knuth는 ACM Turing 시상 강연에서 컴퓨터 프로그래밍을 단순 코딩 작업이 아닌 예술가의 'Art'로 비유하였다.⁹⁾ 컴퓨터 프로그래밍은 창조성 교육과 관련되며 'Art'이며 수학적 힘을 필요로 하는 독특한 영역이다. 컴퓨터 프로그래밍 교육의 의의는 다음과 같이 정리해 볼 수 있다.

- 1) 프로그래밍을 배우는 것은 컴퓨터 사용에 자신감을 줌으로 매우 중요하다.
- 2) 프로그래밍은 학교 교육과정에 포함되어 학생 자신의 효과적인 프로그래밍 스타일을 형성할 수 있어야 한다.
- 3) 프로그래밍을 배우는 과정에서 컴퓨터의 가능성, 신뢰성에 대한 통찰력을 얻는다
- 4) 다른 문제 해결 학습 상황에 응용할 수 있는 지식을 얻게 된다
- 5) 구체적인 프로그래밍 작성 과정은 문제의 이해와 분석, 오류 수정 등을 통하여 효과적이고 유연한 프로그램을 작성하는 기술을 익힐 수 있다.
- 6) 구체적인 수학적 절차를 구성하는 과정에서 수학적 원리, 컴퓨터의 작동 원리, 컴퓨터 자원을 더 깊이 이해하게 된다.

앞에서 수학교육과 컴퓨터 프로그래밍에 관한 NCTM의 여러 주장을 살펴보았다. 컴퓨터 프로그래밍은 학생, 교사, 컴퓨터, 교육 그리고 정보기술의 관점에서 다양한 의미를 갖으며 각 교과를 초월한 현시대의 복잡한 교육적 논쟁의 대상이다.

9) Knuth, Donald E. (1987). *ACM Turing Award Lectures, Computer Programming as an Art: 1974 Award Lecture*, ACM Press, New York.

수학 교육에서 Mathematica, Geometer's Skechpad, LOGO, TI Graphing Calculators는 프로그래밍이 가능하고 프로그래밍 환경이 학생 수준에 적절한 도구이다. 이들 도구를 수학 수업에 도입한 중심 목표는 이론적인 학습에서 이해한 수학 내용을 프로그래밍을 통하여 실험하거나 문제 해결의 도구로 사용하는 것이다. 학생의 수준, 소프트웨어의 이해, 수학 교과과정에 대한 이해, 학생의 반응, 교육 효과에 대한 예측, 수업 계획, 수업 준비, 컴퓨터 자원 준비와 관리 등등 성공적인 수업을 위하여 교사가 고려해야 할 커다란 요소들이 적어도 10 가지가 넘는다.

교육과정을 변화시키고 하드웨어와 소프트웨어의 활용을 늘여 나간다고 해서 교사가 컴퓨터를 수업에 적절하게 교육적 효과를 높이는 방식으로 사용하리라고 보장할 수 없다. 학생들이 컴퓨터 힘과 수학적 힘을 충분히 통찰할 수 있고 학생들도 그러한 힘을 문제 해결에 활용할 수 있도록 교육하는 것이 교사들의 해야 할 일이다.

Mathematica, Geometer's Skechpad, LOGO, TI Graphing Calculators 등이 수학 교사 재교육의 대상으로 나타난 점은 수학 교육에서 학습자의 관점에서 볼 때, 정보기술은 학습 도구(tool)로서 활용하는 측면과 학습 대상(tutee)으로서의 측면이 점점 더 중요해 지고 있음을 알 수 있다. 뿐만 아니라 교사의 역할을 하는 CAI나 Web-based 교육도 이들 교육 시스템 속에서 학생들이 학습 활동할 때 시스템 속에 포함된 학습 도구를 사용 할 수 있어야 하고 또한 학습시스템이 학생을 지도하는 시스템인 경우도 있게 될 것이다.

[7차 수학 교육과정에서 컴퓨터 활용]

현재 한국 수학 교육과정의 철학과 기준 간단히 살펴보자. 한국의 학교 수학교육에도 컴퓨터 활용에 관하여 7차 교육과정 안에서 다음과 같이 제시하고 있다.¹⁰⁾

계산기와 컴퓨터는 수학을 학습하고 지도하는 데 있어서 연필과 종이 이상의 중요한 도구로 사용될 수 있음은 국내외의 여러 연구에서 밝혀진 바 있으며, 미국을 비롯한 선진국에서는 각급 학교의 수학교육에서 계산기와 컴퓨터를 적극적으로 도입하고 있다. 계산기와 컴퓨터는 수학적 개념의 이해, 수학적 사고력, 문제해결력, 창의적 사고력을 기르기 위해 사용될 수 있으나, 교육적 효과를 극대화하기 위해서는 적절한 시기에 수학의 기초 기능을 저해하지 않는 범위에서 조심스럽게 도입되어야 한다.

현행 제 6차 교육과정에서도 계산기와 컴퓨터의 사용을 권장하고 대부분의 교사들은 계산기와 컴퓨터의 필요성을 인식하고 있지만, 실제로는 거의 활용되지 않고 있는 실정이다. 제 7차 교육과정에서는 수학의 전 영역에서 계산기나 컴퓨터를 사용함으로써 복잡한 계산을 수행하기 위한 불필요한 시간의 낭비를 줄이고, 수학적 사고력, 문제해결력, 수학에 대한 흥미와 호기심을 길러 줄 수 있도록 장려하고 있다.

인용된 두 문단의 주장은 수학 교육에서 정보기술을 사용하는 측면에서 보면 수학 교육의 변화는 미국과 상대적으로 대단히 보수적으로 보인다. 그리고 '조심스럽게 도입되어야'라든가 '장려하고'라는 등 애매 모호한 수준에 머물고 있다. NCTM에서 제시하고 있는 21세기 새 천년에 대비한 수학 교육

10) Edunet 교육관련자료실 교육정책자료 7차교육과정(수학) <http://www.edunet4u.net/top.html>

의 방법과 방향에서 보면 대단히 아쉬운 그리고 우려할 만한 수학 교육에 대한 전망으로 보인다.

7차 수학 교육과정을 구성하는 각 주제에 대한 세부 내용과 교육 지도 방법을 들여다보면 특히 중등수학교육과정은 학생들로부터 외면 당할 수밖에 없는 실패가 예견된다. 교사들의 설문 조사 결과도 믿기 어려울 만큼 수학교육에 대한 직업적 열정이 없는 것 같다. 컴퓨터 및 정보통신기술을 사용할 여지를 수학 교육계 스스로 닫아버린 느낌마저 든다.

21세기의 수학 교과 교육이 축소화되는 세계적인 심각한 추세를 우리는 목격하고 있다. 우리 나라의 7차 교육과정 개편 안에도 분명하게 명시되어 있다. 본 논문에서 21세기에 수학을 포함한 컴퓨터 과학 물리학 등 수리과학 분야의 교육은 학생들의 기피 현상이 심화되어 가고 있다는 보고를 자주 접하게 될 것이다. 이와 관련된 여러 문제점들이 체계적으로 연구되어진 바는 없다. 어떻게 대처해야 되는지에 대한 대안 제시도 충분하지 못하다. 교실붕괴와 같은 사건이 수학교과에서 일어나지 않기를 바랄 뿐이다. 수학 교사는 수학을 학생들에게 매혹적이도록 만들 열정이 있어야 될 것 같다.

학습자의 측면에서 20세기에는 듣고 보고 노트 필기를 하고 연습 복습하는 인간의 오감과 신체를 사용한 학습이었다. 현재와 미래는 지식과 정보의 기록과 전달의 수단이 정보기술 도구로 바뀔에 따라 전통적인 학습 방식은 전자-학습(e-education) 시대로 변환하고 있다. 일상생활도 전자상거래(e-commerce)를 넘어서 전자생활(e-life)로 전환되어 가는 시대이다.

전통적으로 사회는 수학에 대하여 지적 사고 방식으로 의미와 가치를 인정해 왔다. 그러나 현재는 수학도 시민이 사회 생활을 하는데 중요한 도구이며 방식일 것을 요구한다. 증명이나 이해보다도 어떤 의미에서 필요한 결과를 획득할 수 있는 (정보)처리 능력이 필요하다고 보는 것이다.

학교 수학 교육이 사회화된 새로운 정보기술의 사용에 열려 있지 않으면 수학 교과와 장래는 어둡게 보인다. 정보기술을 적극적으로 사용함으로써 그 속에서 수학이 절대적으로 필요한 사례와 개념을 발견하고 이러한 것들이 수학 교과 내용으로 구성될 수 있다. 수학 교육자와 교육계는 지금 전통적인 수학 지식의 완전성에 안주하고 고집스럽게 변하지 않으려 하고 있다.

수학 교과와 가치를 높이는 유일한 방법은 수학이 새로운 정보 기술과 가장 실제적인 의미에서 연결되어 있음을 수학 교육자들이 학생과 사회에 증명해 보일 수 있어야 한다. 수학 교과가 정보기술과 강하게 연결되어 있는 부분은 프로그래밍에서 스프레드시트, 그래픽스 그리고 데이터베이스 등 사무용 소프트웨어 활용에 이르기까지 매우 다양하다. 수학적 사고 또는 수학하는 힘과 정보기술의 활용 측면을 잘 조화하고 교사들이 이러한 측면의 수학 교육에 지금보다 적극적인 관심을 갖고 교육 행정에서 적절한 지원을 하는 등등의 주변 여건의 교육 환경의 변화를 유도하여야 한다. 누가 이러한 문제의 주체인가? 왜 실천이 어려운가?

V. 컴퓨터과학 교과과정 속의 수학

앞으로의 학교 수학 교육의 변화를 고려하고 준비한다면 컴퓨터 과학(계산 과학, computer

science)과 수학 사이의 관계를 정리해 볼 필요가 있다. 수학 교과로 볼 때 수학과 유사한 그리고 한 때는 응용 수학의 분야로 분류되었던 컴퓨터과학 분야를 수리과학의 가장 가까운 인접 학문과 교과 분야로 인정해야 하는 시점에 와있다. 10-20년 전부터 대학의 미적분학이 이공계통의 학문 분야에서 필수과목으로 가르쳐 졌던 것 같이 컴퓨터 프로그래밍을 포함한 컴퓨터 관련 교과목 2-3 개 정도를 가르쳐왔다. 현재 대학의 수학 교과과정에 컴퓨터 프로그래밍만 2-3학기 정도 가르쳐지고 있다. 조합론, 이산수학, 그래프론, 확률론 통계학, 수치해석, 확률과정, 알고리즘 등등의 컴퓨터 활용과 밀접한 관련이 있는 교과목의 등장은 수학 교육의 1/3-1/4 이상을 차지한다. 대학의 수학 교과과정은 10년 전과 비교해 볼 때 대수학 기하학 해석학 위상수학 중심의 표준적인 교과과정 형태를 더 이상 유지하고 있지도 못하고 있다. 그만큼 컴퓨터의 실용적인 측면이 대학의 고답적인 순수 수학적인 전통적인 교과과정을 빠르게 변화시키고 있다.

[Hamming의 1968년도 컴퓨터 과학 교과과정에 필요한 수학에 대한 견해]¹¹⁾

이러한 컴퓨터의 발전과 영향에 의한 사회 변화를 이미 컴퓨터 과학자와 교육자들은 오래 전부터 예견해왔고 60년대 말부터 최근까지 컴퓨터과학이 필요로 하는 실제적 의미를 갖는 수학에 대하여 논의하고 표준적인 교과과정을 구성하는 노력을 하여왔다. 그 결과 오늘날의 대부분의 대학의 수학 교과과정을 구성하는 교과목들은 놀랍게도 30년 전 벨연구소의 R. W. Hamming이 1968년 Turing Award Lecture에서 행한 컴퓨터 과학 발전을 위하여 그리고 우수한 컴퓨터과학자를 교육으로 양성하기 위하여 필요성을 인정하여 나열한 수학 교과목들이다.

Hamming은 컴퓨터의 발전 과정에서 훌륭한 컴퓨터 과학자는 정규 교육 과정으로 수학을 배웠기 보다는 선천적으로 수학적 재능가지고 있었다고 생각하였다. 그 당시 컴퓨터 과학의 발전이 사회 경제적 성장에 따르는 요구라는 것을 확신으로부터 수많은 컴퓨터 과학자와 전문 직업인을 필요로 한다는 생각에서 직업적으로 적응력이 높은 컴퓨터 교육 과정을 제안하게 되었다. Hamming은 컴퓨터 과학이 대학에서 독립된 학과로서 교과과정을 구성해야 한다면, 컴퓨터과학 전공자는 모두 수학적 재능을 교육으로 갖추어, 천재적인 과거의 컴퓨터 과학자들과 마찬가지로 강한 수학 지식 기반을 갖추어 컴퓨터를 사용할 수밖에 없다고 생각하였다. 아래와 같은 수학 교과목들이 컴퓨터 과학자의 입장에서 선택이 불가피하다고 하였다.

abstract algebra, Queuing theory, statistics, design of experiments, probability theory,
Markov chains, information and coding theory, graph theory

이러한 수학 교과목을 필요로 하는 이유로 컴퓨터 과학 전공자들이 단순한 컴퓨터라는 기계를 다루는 그리고 알려진 프로그램을 거의 복사하는 수준의 프로그래밍 기술자의 수준을 넘는 전문가로서

11) Hamming, R.W. (1987). *ACM Turing Award Lectures, One Man's View of Computer Science: 1968 Award lecture*, ACM Press, New York.

컴퓨터를 사용하여 중요하고 의미 있는 독창적인 일을 하기 위하여 꼭 필요하다고 하였다. 그리고 수학 교육과 수학적 훈련을 받지 않은 컴퓨터 전공 교육은 막다른 길(dead end)로 가게될 것이라고 하였다.

여기에 컴퓨터과학 교과로 당연하게 생각하는 discrete mathematical structures, algorithms, numerical analysis, combinatorics 등을 포함하면 현재 대학 수학 전공으로서 완벽한 하나의 수학과 교과과정으로 손색이 없어 보인다. Hamming의 강연이 있고 20년 후에도 필요한 수학 교과들은 변하지 않고 대학 컴퓨터과학 및 공학의 교과과정에 대한 ACM/IEEE-CS 의 1991년의 권고안에 다음과 같이 포함되었다.¹²⁾

Discrete Mathematics, Calculus, Probability, Linear Algebra, Mathematical Logic, combinatorics, statistics, recursion, graph theory, algorithms, Numerical computation, Symbolic computation, computation theory

컴퓨터과학은 이러한 수학의 기초 위에 소프트웨어, 정보기술 시스템에 관한 기술 및 공학적 접근을 위한 교육과정이 덧붙여지는 교육이다. 한편으로 현재 대학의 수학 교육은 과거와 다르게 기술 및 공학과 관련된 수학 응용 교과를 포함하고 있다. 컴퓨터 프로그래밍은 적어도 2학기는 필수이다.

앞으로 특히 중등 수학교과 과정은 대학의 수학 교과과정의 변화만큼이나 변화할 것이다. 특히 중등 수학 교과 과정에서의 이산수학, 확률, 통계, 수리모델, 게임이론 등의 과목은 내용이 풍부해지는 변화를 할 것이다. 현재처럼 수학 1, 2 공통수학 등과 같이 책의 목차에 모든 수학적 함의가 있는 수학교과목은 분화되고 심화되어 양과 질 양쪽으로 충실하게 보강되어 변할 것이다.

Hamming의 앨런 튜링상 수상 강연의 내용이 시사하는 바가 크다. Hamming은 ‘What is mathematics’에 대한 자신의 견해를 그 당시 훌륭한 수학자가 언급했던 수학에 관한 정의를 인용하였다. “mathematics is what mathematicians do” 그리고 강연의 주제에 맞추어 ‘what is computer science’에 대한 컴퓨터 과학자인 자신의 견해도 수학자의 견해와 같이 컴퓨터과학은 컴퓨터 과학자가 행하는 것’이라고 말했다. 수학이 다양한 것과 동등하게 컴퓨터과학도 다양하다. 교육은 다양함도 반영하면서 교육의 결과에 대한 측면으로 효율성과 효과를 고려하지 않을 수 없다. 다양함을 지원할 수 있는 컴퓨터 과학이 가져야 할 교육적 아이디어와 개념들이 Hamming은 수학에 있다고 생각한 것이다.

이 시점에서 중등 수학 교사는 가까운 미래에 계속 수학 교사로 있기 위하여 어떤 준비를 하여야 하는가? 미래의 수학 교사가 될 현재 대학의 수학 교과과정의 모습과 수학자들의 수학관은 어떠한가?

20세기 말에 페르마의 문제가 풀리는 등 순수수학의 흥분된 열기가 있기는 하지만 많은 수학자들은 10-20년 전과 같은 순수수학을 game-playing 하듯 연구할 수 있는 시대는 지나갔다고 본다. 수학 전공 졸업생들도 직업적 측면을 고려하지 않으면 안 되는 시대이고 대학의 교과과정도 직업과 관련

12) ACM Computing Curricula 1991: Report of the ACM/IEEE-CS Curriculum Task Force

된 사회 현실의 요구를 수용하기 시작했다. 대학의 수학 교과과정은 현재컴퓨터 및 정보기술 분야와 밀접한 관련을 갖는 수학 교과목들로 구성되어 있다. 10년 전의 수학 전공 졸업자와 다르게 현재의 수학 전공자는 10년 전 컴퓨터과학 전공자만큼이나 컴퓨터프로그래밍, 자료구조, 알고리즘, 소프트웨어 패키지 관련 실습 교과 등등의 컴퓨터 관련 교과목을 실습과 더불어 공부하고 졸업한다.

이러한 상황은 Hamming의 수학과 컴퓨터과학에 대한 교과과정 구성을 위한 견해가 역전되어 있음을 보여 준다. 컴퓨터 과학과 관련된 교과 교육 없이는 대부분의 수학 전공자는 특히 직업적으로 막다른 길로 가게된다. 지금은 수학 교과과정이 필요로 하는 핵심적인 컴퓨터 교과목을 선택해서 구성해야하는 시대가 되었다. 수학이 필요로 하는 컴퓨터 교과목을 선택하려 한다면 Hamming이 컴퓨터 과학 교육에서 중요한 핵심은 컴퓨터 실습이 수반되는 실제적인 컴퓨터 사용과 관련된 교육을 강조한 점을 충분히 고려해야 한다. Hamming은 컴퓨터 교과목이 실제적이어야 한다고 다음과 같이 말하였다.

We need to avoid making computer science look like pure mathematics: our primary (computer science) standard for acceptance should be practical experience in the real computing world, not aesthetics.

Were I setting up a computer science program, I would give relatively more emphasis to laboratory work.

컴퓨터과학의 창조적 영역은 결과적으로 컴퓨터가 새로운 일을 할 수 있는 것을 입증하는 일이고 이는 프로그래밍으로 실현되는 것이다. 컴퓨터과학 분야의 실험실은 컴퓨터와 컴퓨터에게 새로운 일을 명령할 수 있는 프로그램 제작실 또는 설계실인 것이다. 컴퓨터과학자의 궁극적인 창조 활동은 프로그래밍에 있다.

창조적인 프로그래밍 영역은 수학자의 관점에서 보면 컴퓨터과학 전공자들만의 몫이라기보다는 수학전공자 또는 의미를 확대하여 수리과학 전공자들의 영역이기도 하다.

여기서 언급하지는 않았지만 역사적으로 컴퓨터의 발전에 부울, 아다, 앨런 튜링, 존 폰 노이만, 알 폰소 처치 등의 여러 수학자들의 역할이 컸었다. 특히 Hamming이 30년 전인 1968년 컴퓨터과학이 필요로 하는 수학교과목을 열거한 후 미국의 컴퓨터과학의 표준적인 교육과정은 이를 대부분 반영되어 지금도 이러한 전통은 강하게 내려오고 있다. 이와 같이 대학의 교육과정을 통하여 수학과 컴퓨터과학은 수리과학의 중심교과로 한 울타리에 속한 학문적 전통과 발전의 역사를 갖고 있다.

[수학교사는 컴퓨터프로그래밍을 왜 가르쳐야만 하는가?]

전세계적으로 10여전부터 국가 경쟁력 강화의 교육적 전략으로서 컴퓨터과학 교과의 학교교육에서 도입되고 강화되고 있는 현상이다. 특히 컴퓨터교과가 중등교과과정에 들어오고 수학 교과과정이 크게 변화하려고 하는 한국의 상황을 수학 교육계는 주목해야 한다. 현 시점에서 중등교과에서 컴퓨

터 교과를 담당하는 교사는 컴퓨터과학 전공자만으로 충원되어 운영되어서는 안 될 것으로 보인다.

컴퓨터 교과의 본질적인 특성을 고려하면 수학교과와 적절하게 조화되어야 하고 수학과 컴퓨터 과학의 미래의 발전과 경쟁력을 위하여 균형을 이루어야 한다. 그리고 수학교과 교사와 컴퓨터 교과 교사는 수리과학의 틀 안에서 협동하고 공존해야한다. 수학교과와 컴퓨터과학 교사들은 자신이 가르치는 교과에 대하여 의사소통이 될 수 있어야 한다.

수학교과에서 볼 때 학교 교육에 컴퓨터 교과의 도입이 현재 한국의 실정에서 볼 때 상당히 위협스런 측면을 가지고 있다. 한국의 학교 교육의 구조조정을 볼 때 현실적으로 기타 교과 교사의 재교육을 통하여 컴퓨터 교과 교사로 활용하려는 측면이 강하다. 이 점은 학교 컴퓨터 교과 교육을 한국의 컴퓨터 과학자들도 원하지 않는 컴퓨터의 단순 기능 훈련 중심 교육으로 학생들의 흥미와 창의력을 유발시킬 수 없는 교과로 전락시키고 컴퓨터에 진정한 자질이 있는 학생을 탈락시키거나 희생시킬 수 있는 위험이 존재한다. 학교의 컴퓨터 교과 교육은 컴퓨터와 더불어 학생들이 창의력을 자유롭게 발현할 수 있고 자신의 프로그래밍 스타일을 발휘할 수 있는 교육으로 가야할 것이다. 이러한 방향으로 교육이 진행되기 위하여 학교 컴퓨터 교과의 중심에 있는 프로그래밍을 어떻게 교과과정으로 구성할 것인가가 핵심적인 논의 사항일 것이다.

그리고 컴퓨터 교과 과정에 대한 제안의 배경으로 미국 교육의 현재 상황은 수학 컴퓨터 과학 그리고 과학과 같은 수리과학 관련 교과들에 대한 학생들의 관심이 현저히 줄어들 것이라는 학력저하 그리고 국가의 산업 경쟁력 하락이라는 우려에서 출발한다. 이러한 우려를 반영해서 수학 컴퓨터 과학 그리고 물리와 같은 수리 과학 교과의 운영이 학생의 자유로운 선택보다는 엄격한 교과과정 운영과 선수과목 등으로 연계를 강화하고 있다. 이것은 7차 교육과정에서 수리과학 교과목들이 선택 과목화 되어 학생 지도에 혼란이 우려되는 우리의 현실과는 큰 차이가 있다.

국가 경쟁력과 학생의 미래의 복지를 위해서 수리과학 교과를 무제한으로 학생들 자유 선택에 맡길 수 없다. 지식정보사회의 중요성과 컴퓨터와 통신 등등의 정보기술의 중요성과 경쟁력을 외치면서 10-20년 후에 결실이 맺어지는 인적 자원 양성을 교육계는 방치하고 있다는 생각이다. 능력을 갖춘 자에게는 상응하는 충분한 자율을 줄 수 있지만 부족한 학생에게는 자율보다는 엄격한 교육과정의 적용과 운영을 필요로 한다.

VI. 중등 컴퓨터과학 교과 교육 속의 수학 - 미국의 사례

[ACM/IEEE-CS 가 권고한 미국 고등학교 컴퓨터과학 교과 과정]¹³⁾

[소개]

컴퓨터 기술은 사회와 세계에 심대한 영향을 주고 있다. 모든 시민은 가정, 학교, 직장 그리고 사

13) <http://www.acm.org/education/hscur/index.html>

회에서 정보기술과 그 중요성을 적절하게 숙지할 필요가 있다. 매일 매일 정보기술의 세세한 사항들이 변하고 있기 때문에 새로운 사항들을 항상 파악하려는 것은 어려우며 대개는 비생산적인 일이다. 따라서 정보기술에 대한 학습은 주어진 분야의 근본적인 과학적 원리와 개념에 집중해야 한다.

미국 계산 기계 학회(ACM)은 1991년 고등학교 컴퓨터과학 교과과정에 대한 권장안을 IEEE-CS (국제전기전자공학회-컴퓨터과학) 과 공동으로 마련하였다. 1989년에 ACM은 새로운 고등학교 컴퓨터과학 교과과정 권장안에 대한 대책반을 구성하여 보고서를 작성하고 권장안을 폭넓게 검토하였으며 수많은 확장된 권고 사항을 수용하게 되었다.

[동기]

컴퓨터과학 교육에 대한 필요는 자연과학 분야의 교육의 필요성과 같다. 자연과학은 모든 고등학교에서 물리적 세계를 이해하기 위하여 가르쳐지고 있다.

컴퓨터는 여러 가지 방식으로 세계를 근본적으로 변화시키고 있다. 즉

- 1) 세계의 많은 부분이 산업 중심에서 정보 기반 분야로 변화하고
- 2) 일상의 의사 결정이 컴퓨터의 적절한 사용에 기초하여 이루어지며
- 3) 컴퓨터는 전통적인 도구들을 대체하고 있으며
- 4) 컴퓨터는 복잡한 상황을 모델링하는 데 쓰이며
- 5) 컴퓨터는 통신에 널리 사용되고 있다.

고등학생이 자연세계를 이해하기 위하여 자연과학을 공부하는 것과 같이 컴퓨터과학을 정보사회의 사회, 경제 그리고 문화 환경을 이해하기 위하여 공부할 필요가 있다.

현재 고등학교의 컴퓨터교육은 크게 2가지 측면을 가지고 있다;

- 1) 컴퓨터를 다른 교과목의 도구로서 사용한다.
(워드프로세싱, 스프레드시트, 데이터베이스, CAD/CAM, Mathematica)
- 2) 프로그래밍
(Pascal, BASIC, C, Logo, Scheme)

이들 두 가지 형태의 컴퓨터 교육은 컴퓨터과학 교육을 구성하는 요소이지만 컴퓨터과학은 아니다.

컴퓨터과학 교육은 정보기술을 승화시키고 고등학교교육의 핵심을 이루는 기본적인 보편적인 개념들로 구성한다. 학생들이 이해할 수 있고 현재의 세계에 효과적으로 관여할 수 있도록 하는 개념들이다.

권장안 마련을 위한 대책반은 고등학교를 위한 컴퓨터과학 교과과정을 개발하여 왔다. 교과과정은 폭넓은 이해를 위한 1년짜리 컴퓨터 교과를 구성하였다. 개발된 컴퓨터 교과는 교육의 범위, 깊이, 넓이 그리고 방법에 있어서 전형적인 고등학교 과학 교과와 유사하다. 모든 고등학생은 생물, 화학

그리고 물리 교과와 같은 방식으로 컴퓨터교과를 배운다. 교과는 컴퓨터과학의 배경을 제시하고 주요한 문제를 토의하며, 문제 해결을 하고, 문제해결에 수학을 응용한다.

컴퓨터교과의 수준은 10학년도에 도입하도록 의도되었다. 교과는 자체로 최소 필요사항이 되도록 하였으며 고급 컴퓨터 교과에 대한 선수 교과가 되도록 하였다. 교과는 상급 컴퓨터교과 이수 자격 시험(advanced placement computer science course) 준비에 충분하도록 하였다. 컴퓨터 교과에 앞서서 이수해야 할 교과로 고교 1학년의 대수와 약간의 컴퓨터 사용 경험이 요구된다.

교과는 컴퓨터과학의 기초 개념들에 초점을 맞추고 있다. 몇 가지 제안된 교과 과정 모델은 이러한 교과과정의 권고안에 따라 어떻게 다양하게 구현될 수 있는가를 보여준다. 가능한 학생들은 추상 개념들을 실현해 보이기 위하여 프로그램을 작성하고 컴퓨터로 실행해 볼 수 있어서 이론을 확인하고 컴퓨터의 가능성 보일 수 있어야 한다.

컴퓨터 프로그래밍 과정, 응용과정 그리고 소양 과정(programming, applications and literacy courses)은 컴퓨터과학으로 확대되고 진화할 것이다. 이러한 다양한 컴퓨터 교과 과정 모델은 부록에 포함되어 있다.

[배경]

미국 교육의 현재 상황에서 고교 컴퓨터교과과정 권장안을 만들게 된 배경은 다음과 같다;

- 1) 1983년 국가교육위원회는 보고서 “위기에 처한 국가(A Nation at Risk)”를 발간 (보고서는 필요한 컴퓨터과학 교과와 성취시험을 권고)
- 2) ACM/IEEE-CS는 대학 컴퓨터과학 교과과정을 개정
- 3) 고교 교육과정에서 교육적으로 보장할 기초 부분과 노력해야 될 부분에 대한 강조
- 4) 정보기술에 대한 크게 국가적으로 강조하는 시점
- 5) 수학과 과학에 대한 고등학생의 관심이 줄어드는 상황
- 6) 고등학교에서 심지어 대학에서까지 모든 학생들의 특히 소수민족 학생과 여학생의 컴퓨터과학에 대한 관심의 퇴조
- 7) 컴퓨터과학에 대한 명확한 정의 및 정체성에 대한 결여

[교과 주제와 모형(topics and models)]

고교 컴퓨터교과에 대한 대학의 기대를 충족하기 위하여 대학의 컴퓨터 교과과정과 일관성을 가져야한다. 또한 교과과정은 고등학생의 능력, 흥미 그리고 학교 상황을 반영하여야한다. 교과과정을 구성하는 주제와 교육시간의 양은 권고안은 다음과 같이 제시한다.

Introduction to a Programming Language	12 hours
Algorithms and data structures	47 hours

Programming languages	46 hours
Architecture	59 hours
Operating systems	51 hours
Software methodology and engineering	44 hours
Artificial intelligence and robotics	9 hours
Database and information retrieval	9 hours
Human-computer communication	8 hours
Numerical and symbolic computing	7 hours
Social, ethical, and professional issues	11 hours

교과과정 대책팀(Task Force)은 고등학교의 컴퓨터과학을 소개하기 위한 7 분야로 구성된 교과 과정을 구성하였다.

1. Algorithms
2. Programming Languages
3. Operating Systems and User Support
4. Computer Architecture
5. Social, Ethical, and Professional Context
6. Computer Applications
7. Additional Topics

컴퓨터응용은 컴퓨터 사용(활용) 방법을 반영하도록 하였다. 그리고 교과과정은 컴퓨터 교육에서 프로그래밍과 컴퓨터 활용이 균형과 조화를 이루고 통합되도록 하였다.

실습실 운영과 문제해결

실습과 문제 해결 연습은 주제를 실제로 구성하고 컴퓨터로 실험해 보이는 기회를 갖을 뿐 아니라 사용하는 소프트웨어의 세부 사항을 익힌다. 학생은 교육기간 동안 완성해야하는 긴 과제를 수행할 수 있다. 실습은 독립적인 과제를 수행할 수 있는 시간을 제공해 주고 물리나 화학 같은 교과외의 실험 보고서처럼 교과에서 제시된 과제를 수행할 시간을 주어야한다.

[우수 과학 고등학교의 컴퓨터과학 교과과정 사례]¹⁴⁾

LBJ 고등학교 (LBJ High School-Science Academy, Austin, Texas)의 컴퓨터과학 교육 과정

LBJ 고등학교는 우수한 컴퓨터과학 및 과학기술 교육 프로그램을 갖추고 있다. 고등학교 컴퓨터 과학 경진대회를 대비한 컴퓨터 클럽활동이 있고 컴퓨터 교사나 일반 교과 교사를 위한 교내 연수도

14) LBJ high school ap test. <http://www.collegeboard.org/ap/html/index001.html>

실시되며 전담 컴퓨터 기술자가 있어서 컴퓨터나 다른 과학 기술 기재를 관리·정비하고 있다. 노후된 기재의 최신화, 일반 교과에 과학 기술의 통합, 새로운 교육 환경에 대처하기 위한 교사 교육, 선형대수(Linear Algebra), 불리언대수(Boolean Algebra), 논리수학(Logic Math), 수론(Number Theory), 통계학(Statistics) 등 컴퓨터과학의 이론적 기초가 되는 수학교육의 강화에 주력하고 있다.

다음은 LBJ 고등학교의 컴퓨터과학 및 과학기술 과목의 내용을 기술한 것이다.

- 가. 컴퓨터과학 I (Computer Science I) - 프로그래밍 기초(필수교과)
- 나. 컴퓨터과학II(Computer ScienceII-Advanced Placement Course)
자료구조(스택, 큐, 트리)와 프로그래밍 프로젝트로 구성되며 AP프로그래밍 시험에 대비한다.
- 다. 컴퓨터과학 III(Research Methods / Guided Research-CS)
게임이나 C++ 에뮬레이터를 개발하는 등 각자 선택한 프로젝트를 수행하는 C++ 프로그래밍 연구 과제 중심 과목이다. 학생들은 날짜, 이름, 그 주의 활동, 사항, 문제점들을 기록한 프로젝트 진행 보고서를 매주 제출하고 학기말에 완성된 프로젝트를 제출한다.
- 라. 컴퓨터 과학 응용(Computer Science Applications)

컴퓨터 과학 응용은 컴퓨터 응용에 관한 1년 과정으로 10개 핵심모듈과 5개의 선택모듈로 구성되어 있으며 핵심모듈 내용을 다음과 같다.

- 1) 컴퓨터 구조(Computer Architecture) 2) 인터넷 이용(Internet Utilities) 3) 그래픽(Graphics) 4) 통계적 분석(Statistical Analysis) 5) 수학적 모델링(Mathematical Modelling) 6) 멀티미디어(Multimedia-Hypercard/Hyperstudio) 7) HTML 저작(HTML Authoring) 8) 컴퓨터 윤리(Ethics) 9) 전자출판(Desktop Publishing) 10) 소프트웨어 탐구(Software Exploration)

선택모듈 내용은 다음과 같다.

- 1) HTML과 Hypercard 기반 저작(HTML and Hypercard Advanced Authoring) 2) 자바 스크립트(Java Scripting) 3) Novell 네트워크 관리(Novell Netware Management) 4) 소프트웨어에 관한 세미나(Seminars over Software) 5) 유닉스 관리(Unix Administration)

[ACM/IEEE-CS 고등학교 1학기용 컴퓨터과학 교과과정]-Carol E. Wolf, Pace Universit, NY-15)

[개요]

본 교육과정은 컴퓨터에 대한 경험이 거의 없는 고등학생을 대상으로 구성한 것이다. 학생은 대수에 대한 지식을 가지고 있어야 한다. 본 교육과정은 ACM/IEEE-CS가 제시한 권장안의 모든 주제를 포함하기보다는 거의 핵심 주제(core topics)를 포함한다.

본 과정은 주당 2시간의 강의와 2시간의 실습으로 구성된다. 고등학교에서 강의와 실습은 적절히 결합되어 진다. 그러나 토론 중심의 수업과 컴퓨터 작업 수업이 분리되는 것이 바람직하다.

15) <http://www.acm.org/education/hscur/appendixij.html#i>

본인의 수업 경험에서는 학생들이 컴퓨터 작업에 적극 참여하였고 설명이나 해설을 듣기 위하여 실습 작업을 중단하려하지 않았다. 컴퓨터 실습과 강의와 토론을 분리할 때 가장 좋은 교육 과정인 것처럼 보인다. 본인이 수행한 수업은 2학생이 한 대의 컴퓨터를 사용하였다. 이러한 컴퓨터를 2명의 학생이 공동으로 사용하는 방식은 서로 가르쳐주는 협력 교육이 가능하여 좋았다.

컴퓨터 경험이 없는 학생들의 첫 번째 컴퓨터과학 1학기용 교과 수업이기 때문에 실습에서 해야 할 것과 행해야하는 방법을 상세히 기술한 실습실 워크북이 필요하다. 제한 없이 할 수 있는 폭넓은 학기형 과제(open-ended term project)를 제시해 주었다. 대부분의 학생들은 일일이 단계별로 지도 받음이 없이도 어떻게 하는지를 이해할 수 있게 되었다.

Logo와 Pascal은 본 교육에서 사용한 프로그래밍 언어이다. 이들 언어는 프로그래밍언어에 대한 내가 선택한 하나의 제안일 뿐이다. 여러 학교에서는 Scheme과 BASIC을 사용하여 성공하였다. 프로그래밍 언어의 기초적인 사항만을 소개하였기 때문에 프로그래밍의 특별한 주제는 크게 중요하지 않다.

[1주 강의] 컴퓨터와 교과 소개

- 컴퓨터가 할 수 있는 것과 수업과 교과의 요구 사항들을 소개
- 실습: 컴퓨터와 실습실 자원, 파일, Logo
- 학교 컴퓨터 실습실에 사용 가능한 자원에 대하여 충분한 정보를 줌으로서 학생들이 컴퓨터를 사용할 수 있도록 한다. 디스켓 포맷, 프로그램과 소프트웨어의 실행 할 수 있도록 한다. Logo 실습을 통하여 도형 그리기 등의 logo의 여러 모습을 소개하고 작업한 것을 디스크에 저장하도록 한다.

[2주 강의] 컴퓨터의 요소, 알고리즘

- 컴퓨터의 주요 부분 CPU, 기억장치, 입출력장치를 소개. 컴퓨터에 작업을 시키는 명령서(처방서)로서 알고리즘을 소개
- 실습: 문서작성
- 문서를 작성하고 수정하고 철자를 검사하고 인쇄하는 문서작성의 여러 기능을 실습

[3주 강의] 알고리즘, Logo 프로그램

- 일상적인 보통의 예와 Logo 프로그램을 통하여 알고리즘이 무엇인가를 소개
- 실습: Logo
- 로고 프로그램을 타자하여 작성하고 실행. 프로그램 연습문제를 주고 스스로 작성하고 실행하도록 함

[4주 강의] 컴퓨터 발전 역사과, 사회적 영향

- 컴퓨터가 발전해온 역사와 일상 생활에 영향을 미친 요소를 소개하고 토의
- 실습: 스프레드시트소개
- 스프레드시트 실습을 위한 매뉴얼 워크북 사용. 수식을 계산하고 수식을 변경하고 작업 결과를 디스크에 저장하고 인쇄하는 기능을 훈련

[5주 강의] 수 체계, ASCII 코드

- 이진수, 정수의 보수, 수의 과학기술 표기법, 문자에 대한 ASCII 코드
 - 실습: 스프레드시트에서 식
- 은행이자 표를 작성하는 것과 같은 스프레드시트에서 복잡한 식의 사용 방법.
- 식을 복사하고 이동하여 사용하는 방법

[6주 강의] 스프레드시트와 logo에서 식의 표현

- 스프레드시트와 Logo의 산술연산자와 내장된 함수 소개, 연산자 우선 순위 소개
 - 실습: 스프레드시트에서 그래픽
- 스프레드시트에 기록된 자료에 대한 그래프, 도표, 발표용 도표 그리기

[7주 강의] 문제해결, Logo 변수, 조건문

- Logo 프로그램에서 변수를 통한 컴퓨터의 정보의 저장 방식 소개. 로고프로그램에서 변수를 사용하여 조건문을 사용하는 방법을 소개
 - 실습: Logo
- 적절한 수준의 몇 개의 Logo 예제 프로그램을 제시. 파라미터 또는 글로벌 변수 갖는 프로그램 소개 하고 연습문제를 주어 변수를 사용하는 프로그램을 작성하는 실습

[8주 강의] 부울변수, 진리표, 데이터 베이스와 논리연산 and or not, 논리 데이터형

- 데이터베이스에서 파일 레코드, 필드 그리고 데이터형을 소개
 - 실습: 데이터베이스 생성과 활용
- 데이터 베이스를 구성하고 사용하도록 한다. 조건식을 작성하여 레코드 탐색을 하고 조건에 맞는 탐색 자료를 보고서로 작성

[9주 강의] 데이터베이스 질의어, 탐색과 정렬, 인덱싱, 정보보호

- 탐색과 정렬 알고리즘 소개. 인덱싱이 작동하는 방법을 설명, 데이터베이스 다양한 응용 영역 그리고 유용성, 오용성, 남용성 토의
 - 실습: 데이터베이스 질의어
- 데이터베이스작성, 정렬, 인덱싱, 복잡한 질의어로 표현된 조건문을 사용한 탐색, 결과 보고서로 작성

[10주 강의] 컴퓨터 칩, 게이트, 논리회로 그리고 가산기

- AND, OR, NOT, NAND, NOR 게이트 소개, 게이트 조합 논리회로 설계, 가산기의 논리회로
 - 실습: 데이터베이스 질의어 보다 복잡한 데이터베이스에 대한 실습

[11주 강의] 문제해결과 알고리즘

- 탐색과 정렬 알고리즘 토의. 순환 개념 소개
 - 실습: Logo에서 순환 프로그램
- 간단한 순환개념이 사용된 Logo 프로그램 제시하고 실험, 스스로 순환으로 수행하는 logo 프로그램 작성. 눈송이 곡선과 같은 프랙탈그래픽을 통한 보다 복잡한 순환이 포함된 Logo 프로그램제시

[12주 강의] 컴퓨터 구조, 계산 기계

- 간단한 계산기계, 폰노이만형 컴퓨터의 예 소개. 간단한 프로그램의 수행하는 과정을 추적하는 설명
- 실습: 컴퓨터 시뮬레이션
- 강의에 제시된 기계장치를 흉내내는 프로그램을 작성하여 실행. 몇 개의 기계장치를 흉내내는 예제 프로그램 제시해주고 기계장치에 대한 창의적 프로그램을 작성

[13주 강의] 프로그래밍언어, 파스칼 언어

- 여러 유형의 프로그래밍 언어와 사용 그리고 특징을 소개. Logo 프로그램과 비교하여 파스칼 프로그램 스타일과 구조를 소개
- 실습: 파스칼 프로그래밍
- 간단한 예시 파스칼 프로그램 제시하고 이를 타이핑하고 실행, 예시 프로그램을 수정하고 자신의 독창적 프로그램 작성

[14주 강의] 파스칼언어

- 같은 일을 하는 Logo 프로그램과 파스칼 프로그램 비교, 컴파일러, 인터프리터, 어셈블리의 차이점 설명
- 실습: 파스칼 프로그래밍
- 보다 수준 높은 복잡한 예시 파스칼 프로그램 실행 후 유사한 복잡한 독창적 파스칼 프로그램 작성

[15주 강의] 컴퓨터의 사회에 대한 영향

- 컴퓨터가 한편으로는 우리의 일상 생활을 편리하게 하고 반대로 사회적 문제를 만드는 점에 대하여 토의
- 실습: 학기말 과제 발표
- 학기 동안 팀으로 수행하는 학기말 과제를 학생들이 구두로 보고 발표하고 구현한 프로젝트 결과를 시험해 보인다.

VII. 맺음말

일리노이대 수학과는 웹을 통한 수학 교사 재교육 모듈이 Mathematica, GSP, Logo, JAVA와 같은 프로그래밍을 필요로 하는 정보기술 교육임을 확인하였다. 컴퓨터과학자 Hamming의 견해와 ACM의 표준 컴퓨터과학 교과과정에서 필수 수학 분야를 알아보았다. 그리고 ACM/IEEE-CS에서 제시한 고교 컴퓨터과학 교과과정과 Logo 및 파스칼을 사용하는 1학기용의 강의 안을 보았다. 컴퓨터과학이 수학의 분야이며 앞으로 중등 수학 교육에서 Logo, C, JAVA, GSP, Mathematica를 사용하는 교육에 프로그래밍에 대한 교육적 관점을 보다 명확히 할 필요가 있다. 그리고 수학 교사들은 수학과 프로그래밍이 매우 중요한 관계를 갖고 있음을 이해하고 스스로 프로그래밍에 관심을 갖고 능숙해져야 한다.

참 고 문 헌

- 강오한 (1998). 미국의 컴퓨터 교과교육 연구 실태조사 연구, 교원대 연구과제 RR97-V-3.
- 교육부 (1997.1). 96~2000 교육정보화 촉진 시행 계획.
- 교육부 (1997). 수학과 교육과정, 교육부 고시 제1997-15호 [별책 8], 교육부.
- 구광조·강완 (1996). 모두가 중요하다, 학교 수학의 재구성, 한국수학교육학회 연구 자료, 96-I, II.
- 구광조·진평국·강완 (1996). 수학 교육 개혁 방안에 관한 연구, 한국교원대학교 부설 373-398, 교과 교육 공동연구소, 연구보고 RR 94-I-2.
- 구광조·오병승·류희찬(공역) (1992). 수학교육과정과 평가의 새로운 방향, 서울: 경문사.
- Backus, John W. (1978). Can Programming be Liberated From the von Neumann Style? A Functional Style and Its Algebra of Programs, *Comm. ACM.*, Vol. 21, pp.613-614.
- Educational Software Evaluation Consortium. (1997). The 1997 educational software preview guide. Eugene, OR: ISTE.
- Hamming, R.W. (1987). *ACM Turing Award Lectures, One Man's View of Computer Science: 1968 Award lecture*, ACM Press, New York.
- Harel, D. (1992). *Algorithms: The Spirit of Computing*, 2nd Ed, Reading, Mass., Addison-Wesley.
- Harel, I. (1988). *Software Design for Learning: Children's Learning Fractions and Logo Programming Through Instructional Software Design*. Unpublished Ph.D. Dissertation. Cambridge, MA: Media Laboratory, MIT.
- Knuth, Donald E. (1987). *ACM Turing Award Lectures, Computer Programming as an Art: 1974 Award Lecture*, ACM Press, New York.
- NCTM (1980). *Agenda for Action*. Reston: NCTM.
- NCTM (1989). *Curriculum and Evaluation Standards for School Mathematics*. Reston, Va: NCTM.
- NCTM (1991). *Professional Standards for Teaching Mathematics*. Reston, Va: NCTM.
- NCTM (1995). *Assessment standards for school mathematics*. Reston, VA: NCTM.
- NCTM (1998). *Principles and Standards for School Mathematics: Discussion Draft*.
- NRC (1989a). *Everybody Counts: A Report to the Nation on the Future of Mathematics Education*, National Academy Press.
- NRC (1989b). *Reshaping School Mathematics: A Philosophy and Framework for Curriculum*, National Academy Press, MSEB.
- Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas*. New York: Basic Books.
- Swan, K. & Black, J. (1990). Results of Four Studies on Logo Programming, Problem Solving, and Knowledge Based Instructional Design, ICTE, Brussell, Belgium, March 20-22.

- Taylor, R. (1980). *The Computer in The School: Tutor, tool, tutee*, New York: Teachers College Press.
- U.S.A.CNII. (1996). *KickStart initiative: Connecting America's communities to the information superhighway*. St. Paul, MN: West Publishing Company.
- U.S. Congress, Office of Technology Assessment. (1995). *Teachers & technology: Making the connection (OTA-EHR-616)*. Washington, DC: U.S. Government Printing Office.
- USDOE. (1994). *Why is Technology Planning a Part of Goals 2000*. US Department of Education, Washington, DC.

부 록

본 논문에 사용된 웹 사이트, 문서 그리고 파일에 대한 제목과 URL 주소를 정리하였다.

- EduNet 교육관련 자료실 교육정책자료 7차교육과정(수학) <http://www.edunet4u.net/top.html>
- 한국교원대학교부설 교과교육공동연구소, 1997년도결과보고서 <http://knuecc-sun.knue.ac.kr/~rcsme/>
- ACM - Association for Computing Machinery, <http://www.acm.org/>
- ACM Curricula Recommendations Volume I: Computing Curricula 1991: Report of the ACM/IEEE-CS Joint Curriculum Task Force
- ACM - Association for Computing Machinery, Model High School Computer Science Curriculum <http://www.acm.org/education/hscur/index.html>
- ACM/High School Computer Science Curriculum Appendix I Appendix J. Last Update: 11/11/97 <http://www.acm.org/education/hscur/appendixij.html#i>
- Computers As Teacher And Tutor. [http://www.cincenttic.net/~skiplac/computer.html\(1995\)](http://www.cincenttic.net/~skiplac/computer.html(1995)).
- Computers in Education. <http://www.edcom.org/tstheis/thesisCh3.htm>
- Computer in Schools—a framework for development <http://www.clare.tased.edu.au>
- Brian Harvey-Logo Programming Education <http://www.cs.berkeley.edu/~bh/papers.html>
- Logo Programming for the Mathematics Classroom MTL <http://mtl.math.uiuc.edu/index.htm>
- Math Teacher Link <http://mtl.math.uiuc.edu/index.htm>
- NCTM(전미수학교사협의회) <http://www.nctm.org/>.