

미국 영국 독일 컴퓨터 교육과정에서 한국 컴퓨터 교육의 시사점

김갑수

서울교육대학교 컴퓨터교육과

요 약

21세기 지식정보 사회에서 미래 세대에 대한 컴퓨터 교육은 매우 중요하다. 따라서 선진국인 미국, 영국, 독일은 컴퓨터 교육을 어떻게 하고 있는지 알아보는 것은 매우 중요하다. 미국의 경우에는 NETS 교육과정과 ACM 교육과정을 알아보고, 영국의 국가 수준의 컴퓨터 교육과정을 알아보고 또한 독일에서 국가수준의 컴퓨터 교육과정을 알아본다. 미국, 영국, 독일의 교육과정에서는 컴퓨터 과학, 정보 기술, 디지털 리터러시가 포함되어 있고, 컴퓨터 교과가 독립적으로 되어 있고, 최소한의 시수를 초등학교에서는 17시간 이상 한다는 것을 알 수 있다. 2015년 개정된 우리나라 초등학교 컴퓨터 교육과정은 독립교과로 운영되지 않기 때문에 정체성이 없고. 내용으로는 소프트웨어 교육만 포함되어 있고, 시간수도 17시간이라는 제한이 매우 문제이다.

키워드 : 미국 컴퓨터 교육, 영국 컴퓨터 교육, 독일 컴퓨터 교육, 초등학교, 소프트웨어 교육, 교육과정

An Implications of Computer Education in Korea from the U.S., U.K. and Germany Computer Curriculums

Kapsu Kim

Dept. of Computer Education, Seoul National University of Education

ABSTRACT

The 21st century computer education is very important for the future generations in the knowledge and information society. Therefore, developed countries that are the US, UK, and Germany, it is very important to find out how the computer education. In the US, the NETS and ACM K12 curriculum are investigated, In the UK and Germany national level computer curriculums are looked out. USA, UK, and Germany's curriculum includes computer science, information technology, digital literacy, and computer curriculum is independent, at least in the elementary schools computer education can be seen that more than 17 hours. After the revised national curriculum at 2015 is amended in korea. compared to the developed countries to derive the problems of the Korean revised national curriculum 2015. The revised 2015 Korean elementary school computer education, has no identity because it is not operated as an independent subject. Software will include education only. A very limited time of 17 hours may be a problem.

Keywords : U.K. Computer Education, U.S Computer Education, Germany Computer Education, Elementary School, Software Education, Curriculum

이 논문은 2015년도 서울교육대학교 연구교수 연구비 지원을 받아 수행된 연구임.

논문투고 : 2016-08-22

논문심사 : 2016-08-22

심사완료 : 2016-08-30

1. 서론

교육부는 2015년에 문·이과 통합 교육과정을 발표하면서 중요한 요소 중 하나로 사회 이슈를 반영한 소프트웨어 교육 안을 마련하였고, 초등학교 5-6학년을 대상으로 실과 시간에 17시간을 이용하여 소프트웨어 교육을 하는 안을 만들어 성취기준을 발표하였지만 교사들인 인지 수준은 매우 낮다[13].

미국, 독일, 영국 등은 초등학교부터 컴퓨터 교육을 적극적으로 실시하여 고급 인재를 양성하고 있다. 전 분야에서 컴퓨터 과학자와 같은 사고력[21][22]이 매우 중요하기 때문에, 초등학교 때부터 컴퓨터 교육을 적극적으로 실시하고 있다[1][2][8][15]. 또한, 컴퓨터 교육을 적극적으로 실시함으로써 초등학생들의 논리와 창의력을 향상[12]시키기 때문에 컴퓨터 교육을 중요시 하고 있다. 우리나라에서는 2000년 12월에 초등학교 1학년부 터 주 1시간씩 ICT 교육을 실시하여 컴퓨터 교육을 세계에서 선도적으로 실시하였고[17], 2006년 12월에 개선안을 만들어 알고리즘 개념이 들어간 ICT 교육 지침 개선안[18]을 만들었지만, 2008년 말에 ICT 교육 지침을 폐지한 후 컴퓨터 교육이 매우 축소되어 학교 간 격차가 심화되었다.

21세기 인공지능 시대에 컴퓨터 교육의 중요성은 아무리 강조해도 지나치지 않을 정도이지만 우리나라 컴퓨터 교육은 교과 이기주의로 자리 잡지 못하고 있다. 필요성은 인정하지만 교과들 간의 이기적인 관점에서 컴퓨터 교육을 수용하지 못하고 있는 것이다. 또한, PISA의 컴퓨터 능력 평가에서 우리나라 학생들의 능력이 떨어지기 때문에 초등학교 때부터 컴퓨터 교육을 실시해야 한다[9][11][19].

본 연구에서는 선진국인 미국, 영국, 독일에서 초등학교 컴퓨터 교육이 구체적으로 어떤 교육과정으로 구성되어 있는지 알아보고 우리나라 컴퓨터 교육과 비교하여 분석한다.

구체적으로 미국의 경우에는 ACM에서 초등학교부터 고등학교까지 어떤 내용으로 가르치고 있는지에 대한 성취 기준을 제안하였는데, 2011년에 제안한 초등학교 컴퓨터 교육과정을 구체적으로 살펴 볼 필요가 있다 [5][6][20]. ISTE에서는 1998년도에 학생들이 알아야 할 컴퓨터 교육 내용에 대한 표준을 정했고, 이후 2007년도

에 컴퓨터 교육에 대한 교육과정을 개정하였으며, 최근 2016년도에 최신 표준안을 발표한 것을 설명한다. 2014년부터 영국에서 운영하고 있는 새로운 초등학교 컴퓨터 교육과정을 상세히 살펴보고[3][4]. 독일에서 2008년도[7]에 발표한 컴퓨터 교육과정 중에서 초등학교 고학년에 해당하는 교육과정을 설명한다.

미국 독일, 영국의 교육과정을 분석한 것을 기반으로 우리나라 교육부가 개정한 2015 개정교육과정[16]을 문제점을 논한다.

2장에서는 미국, 영국, 독일 및 우리나라 초등학교에서의 컴퓨터 교육과정들을 설명하고, 3장에서는 선진국인 3개 국가의 컴퓨터 교육과정을 기반으로 우리나라 컴퓨터 교육과정의 문제점들을 알아본다. 4장에서는 연구의 결과이다.

2. 관련 연구

2.1 미국 컴퓨터 교육과정

2.1.1 NETS 초·중등 컴퓨터 교육과정

1998년에 초·중 등 컴퓨터 교육과정의 기본 철학은 테크놀로지 사용법을 배우는 것(Learning to use technology)에서 2007년도에는 테크놀로지를 이용하여 학습하는 것(Using technology to learn)으로, 2016년에는 테크놀로지와 함께 변형 학습하는 것(Transformative learning with technology)으로 컴퓨터 교육이 변화하고 있다.

2007년도의 NETS 교육과정은[14] 다음과 같은 6개의 기본 철학은 창의성과 개혁, 통신과 협력, 연구와 정보 유창성, 비판적 사고, 문제 해결 및 의사결정, 디지털 시민정신, 테크놀로지 운영 및 개념을 익히게 하는 것이다.

2016년도에 개정된 교육과정 틀[14]은 다음과 같다. 첫 번째는 능력 있는 학습자(Empowered Learner)를 만드는 것이다. 학생들은 과학적으로 학습을 하기 위해서 학습 목표에 적합한 능력을 선택하여 성취하고 표현할 수 있는 능동적인 역할을 해야 한다.

학생들은 개인의 학습 목표를 명확하게 설정하고, 목표를 성취하기 위해서 전략적으로 컴퓨터 기술을 이용하며, 좋은 결과물을 만들기 위해서 컴퓨터 기술을 학습 과정에 반영하고 네트워크를 구축하고 학습 과정을 지원하는 방식으로 자신의 학습 환경을 직접 구축하고 컴퓨터 기술을 사용하여 실습을 알리고 개선하는 피드백을 하며, 다양한 방법으로 학습을 입증하고 기술 운영의 기본 개념을 이해하고, 현재의 기술을 선택하여 사용하고 문제점들을 알 수 있고, 새로운 기술을 탐구하기 위해 자신의 지식을 적용한다.

두 번째는 디지털 시민정신(Digital Citizen)을 갖는 학생을 만드는 것이다. 학생들은 권리, 책임, 생활 학습 및 상호 연결된 디지털 세계에서 작업의 기회를 인식하고 안전, 법적, 윤리적 방식으로 행동하고 모범 보이고 다음과 같은 것을 교육하고 있다.

학생들은 디지털 ID와 명성을 구축 및 관리하고, 디지털 세계에서 자신의 행동의 영속성을 알고, 기술을 사용하는 경우 긍정, 안전, 법적이고 윤리적인 태도로 참여하고, 사회적 상호작용을 온라인이나 네트워크 장치를 사용하고 권리와 지적 재산권을 사용하거나 공유하는 것을 이해하고 지적 재산권을 존중해야 하고 디지털 개인 정보 보호 및 보안을 유지하기 위해 자신의 개인 정보를 관리하고 온라인으로 탐색을 추적하는데 사용하는 데이터 수집 기술을 알다.

세 번째는 지식 구성자(Knowledge Constructor)이다. 이것은 학생들이 자신과 다른 사람을 위해 다양한 자원과 디지털 도구를 사용하여 지식을 구성, 창의적인 이슈를 생성, 의미 있는 학습경험을 만들어 비판적으로 선별하는 것이다.

학생들은 지적이고 창의적인 활동을 추구하기 위해서 정보나 다른 자원들을 이용하여 효과적인 연구 전략을 계획하고 전개하고, 정확하고 전망 있고, 신뢰성 있는 정보, 미디어, 데이터 또는 기타 자원의 관련성을 평가 하고, 의미 있는 연관성이나 결론을 보여주기 위한 사례들을 수집하는 것을 만들기 위하여 다양한 도구와 방법을 이용

하여 디지털 자원으로부터 정보를 선별한다. 학생들은 적극적으로 현실 세계의 이슈와 문제를 탐구하고 아이디어와 이론을 개발하고 해법을 추구하는 지식을 구축 한다.

네 번째는 혁신적인 디자이너(Innovative Designer)이다. 학생들은 디자인 프로세스에서 다양한 기술을 사용하여 새로운 유용한 상상력 또는 솔루션을 생성하여 문제를 해결할 수 있고 다음과 같은 것을 교육한다.

학생들은 아이디어를 생성하고 이론을 실험하며, 창의적인 인공물을 만들거나 확실한 문제를 해결하기 위하여 의도적인 설계 프로세스를 알고 사용하고, 디지털 도구를 선택하고 사용해서 디자인의 제약이나 계산된 리스크를 고려하여 설계, 프로세스를 계획하고 관리 하고, 순환 설계 프로세스의 일환으로 프로토타입을 개발하고 시험하고 구체화할 수 있다. 학생들은 개방형 문제와 함께 모호함에 대한 관용과 능력을 보여준다.

다섯 번째는 컴퓨테이션 사교자(Computational Thinker)가 되어야 한다. 학생들은 개발 및 개발 기술 방법의 힘을 활용하는 방법 및 테스트 솔루션의 이해와 문제 해결을 위한 전략을 사용한다. 이를 위해서 학생들에게 다음과 같은 것을 교육한다.

학생들은 해법을 탐구하고 찾기 위해서 데이터 분석, 추상 모델과 알고리즘 사고 등의 기술을 이용하고, 데이터를 수집하거나 관련 있는 데이터 집합을 찾아내고 디지털 도구를 이용하여 데이터를 분석하고, 여러 가지 방법으로 데이터를 표현해서 문제 해결 및 의사 결정을 용이하게 하고, 문제를 작은 부분으로 나누어서 중요한 요소들을 뽑아 낼 수 있고, 복잡한 시스템을 이해하거나 문제 해결을 하고, 자동화를 이해하고 알고리즘 사고를 이용하여 자동화된 해법을 만들고 시험한다.

여섯 번째는 창의적인 소통자(Creative Communicator)이다. 학생들은 명확하게 의사소통하고 자신의 목표에 플랫폼, 도구, 스타일, 형식 및 디지털 미디어를 적절하게 사용하여 다양한 목적을 창의적으로 자신을 표현하고 다음과 같은 것을 교육한다.

학생들은 목표 달성을 위해서 목표를 설정하거나 통신을 위해서 적절한 플랫폼과 도구를 선택하고, 오리지널 작품을 만들거나 책임의 용도를 변경하거나 새로운 작품으로 디지털 자원을 재구성하고, 시각화, 모델이나 시뮬레이션 등의 디지털 다양한 개체를 만들거나 사용함으로써 복잡한 아이디어를 명확하고 효과적으로 전달하고, 의도된 청중을 위해 메시지나 미디어를 게시하거나 표현한다.

일곱 번째 글로벌 협력자(Global Collaborator)이다. 학생들은 자신의 관점을 확대하고 다른 사람들과 협력하며, 로컬 및 글로벌 팀에 효과적으로 협력하여 자신의 학습을 풍부하게 디지털 도구를 사용할 수 있고 다음과 같은 것을 교육한다.

학생들은 상호 이해와 학습을 확장하는 방법으로 다른 배경과 문화를 가진 학습자와 연결하는 디지털 도구를 사용하고, 여러 관점에서 문제를 검토하고 동료, 전문가 또는 지역 사회 구성원을 포함하여 다른 사람과 함께 일을 하는 협업 기술을 사용하고, 공통의 목표를 위해 효과적이고 다양한 역할과 책임을 지닌 팀 프로젝트에 건설적으로 기여하고, 학생들은 지역 및 글로벌 이슈를 탐구하고 해결책을 조사하기 위해 다른 사람들과 협업 기술을 이용한다.

2.1.2 ACM 초중등 컴퓨터 교육과정

2ACM 교육과정에서 컴퓨터 과학이 핵심교육과정이 되어야 하는 이유는 컴퓨터 과학이 지적으로 중요하고, 컴퓨터 과학이 다양한 직업의 필수요인이며, 컴퓨터 과학이 문제 해결 능력을 길러준다는 점이다. 또한, 컴퓨터 과학이 다른 과학을 지원하거나 연결되어 있기 때문이고, 컴퓨터 과학은 모든 학생이 재미있어하기 때문이다.

2011년 ACM 교육과정의 영역은 계산 사고(Computational Thinking), 협력(Collaboration), 컴퓨팅 실습 및 프로그래밍(Computing Practice and Programming), 컴퓨터와 통신 장비(Computer and Communications Devices), 공동체, 국제화 및 윤리적 영향(Community, Global and Ethical impacts)[5]이다.

2016년 교육과정 개편에서는 컴퓨팅 시스템(Computing Systems), 네트워크와 인터넷(Networks and the Internet), 알고리즘과 프로그래밍(Algorithms and Programming), 데이터와 분석(Data and Analysis) 그리고 컴퓨팅의 영향(Impacts of Computing) 영역[6]으로 구성된다. 다음은 2016 CSTA 표안을 그대로 번역한 것[6]이다.

초등학교 2학년까지 알고리즘과 프로그램 영역에서는 다음과 같은 것을 학습하게 한다.

다른 사람이 만든 코드, 음악, 그림을 사용할 때에 신용도를 준다. 블록기반 비주얼 언어를 이용하여 순차, 이벤트, 단순 반복을 포함하는 프로그램을 만들 수 있다. 생각, 아이디어, 스토리를 순차적인 방법으로 계획할 수 있고 문서화 한다. 숫자나 다른 심볼을 사용하여 데이터를 표현한다.(선생님 안내 또는 스스로) 큰 문제를 더 작은 문제로 나눌 수 있다. 컴퓨터 장비를 사용하거나 사용하지 않고 어떤 속성을 정하여 항목들을 분류하여 정리한다. 어떤 일을 수행하는데 순차와 단순 반복문을 포함하는 알고리즘을 만들고 수행한다. 순차와 단순 반복문을 포함하는 알고리즘을 분석하고 수정 또는 틀린 것을 찾을 수 있다.

컴퓨팅 시스템 분야는 다음과 같다.

컴퓨터이셔널 디바이스를 제어하는 소프트웨어를 찾을 수 있고 사용한다. 예를 들어 앱을 사용하여 스크린을 그리거나 소프트웨어를 사용하여 로봇을 제어한다. 컴퓨터 장비들에 대해 적절한 용어들을 사용할 수 있고 기능을 정의 한다. 사용 중에 하드웨어 및 소프트웨어 문제를 정확하게 인식한다.

데이터 및 분석 분야는 다음과 같다.

시간 변화에 따른 데이터를 수집하고 어떤 예측을 위해서 차트나 그래프를 구성한다. 컴퓨팅 장비를 이용하여 정보를 저장, 탐색, 검색, 수정 및 삭제할 수 있고, 정보를 정의한다. 패턴과 기본적인 요소를 찾아내기 위해서 객체나 프로세스를 생성한다.

컴퓨팅의 영향 부분에서는 컴퓨터 기술이 사람들의 생활, 일, 상호작용을 어떻게 변화하게 하고 개선하는지를 비교하여 설명한다.

네트워크와 인터넷 부분은 다음과 같다.

어떤 클래스의 정보를 통신하고 접근하고 공유하기 위하여 네트워크를 사용하여 사람과 연결하게 한다는 것을 안다. 비밀번호를 사용하여 개인정보를 보호하고 비밀번호를 사용하지 않을 때의 영향을 토론하게 한다.

초등학교 3학년에서 5학년까지는 각 영역별로 다음과 같은 학습을 한다.

알고리즘과 프로그램 영역은 다음과 같다.

어떤 문제를 해결하기 위해서 협동적인 전략을 수행한다. 아이디어를 가져오고 변경할 때에 적절하게 인용하여 작성한다. 스토리보드, 프로차트, 슈도코드 및 스토리 맵을 사용하여 설계하고 계획서를 만들 수 있다. 블록기반 비주얼 프로그램 또는 텍스트 기반 언어를 사용하여 순차, 이벤트, 반복, 조건, 병행 및 변수를 사용하여 프로그램을 만들 수 있다. 수학적 연산을 이용하여 변수에 저장된 값을 변경한다. 큰 문제를 작은 여러 개의 문제로 나눌 수 있다. 순차, 반복, 조건이 들어가는 알고리즘을 만들고 수행한다. 순차, 이벤트, 반복, 조건, 병행 및 변수가 들어가는 알고리즘을 분석하고 수정한다.

컴퓨팅 시스템 분야는 다음과 같다.

컴퓨터 시스템이 어떻게 동작하는지 모델링을 한다. 컴퓨터 장비에 적당한 이름을 붙일 수 있고 그들의 관계를 설명한다. 문제를 풀기 위해서 발생할 수 있는 하드웨어 및 소프트웨어 문제들을 정확하게 확인한다.

데이터 및 분석 분야는 다음과 같다.

솔라 시스템과 같은 어떤 개념에 연관된 속성과 행위들을 모델링하기 위해 계산하는 인공물을 만들 수 있다. 학생이나 학급에서 수집한 데이터

를 다루고 분석하기 위해서 컴퓨터를 사용한다. 컴퓨터에서 비수치형 데이터를 표현하기 위해서 수치 데이터를 사용한다.

컴퓨팅의 영향 분야는 다음과 같다.

일상생활에 컴퓨터와 컴퓨팅의 영향으로 긍정적인 측면과 부정적인 측면을 평가하고 설명한다. 컴퓨팅이 사회에 어떻게 영향을 미치는지 설명할 수 있고 또한 사회적인 가치가 컴퓨팅의 선택에 어떻게 영향을 미치는지 설명한다. 어떤 프로젝트를 개선하기 위해서 다양한 관점으로 탐구하고 비교한다. 브레인스토밍 방법으로 컴퓨팅 장비가 모든 사람들에게 접근할 수 있는지 설명한다. 컴퓨팅 장비와 네트워크를 사용함으로써 문제점을 설명한다.

마지막으로 네트워크와 인터넷이다.

강력한 비밀번호가 왜 사용되어야 하고 만들어 있어야 하며, 개인 비밀번호를 사용하고 보호할 수 있다. 네트워크상에서 장비가 메시지를 어떤 규칙에 의해서 다른 장비로 보내어 지는 것을 모델링 할 수 있다.

2.2 영국의 컴퓨터 교육과정

영국은 1992년 국가 교육과정을 설정하면서 ‘정보기술’을 11개 필수 교과로 규정하여 ICT 교육을 주로 실시해 오다가 2014년부터 교과목명을 ‘computing’이라는 과목으로 변경하여 초등학교 1학년부터 컴퓨터 교육을 실시하고 있다[3][4].

컴퓨터 교육과정은 2014년 이전까지는 ICT 활용에 중점을 두었다. 이후 다른 사람이 만든 것을 사용하는 것도 중요하지만 직접 만들 수 있는 교육과정에 역점을 두었고, 프로그래밍을 작성하는 데 중점을 두었다.

영국의 컴퓨팅 과목의 구체적인 교육 목표는 추상화, 논리, 알고리즘, 데이터 표현이 포함된 컴퓨터 과학의 기본적인 개념과 원리를 이해하고 적용할 수 있으며, 컴퓨테이션 용어를 이용하여 문제들을 분석, 해결하기

위하여 컴퓨터 프로그램을 작성한다. 또한, 문제를 분석적으로 해결하기 위하여 새로운 정보 기술을 평가하여 활용할 수 있고, 정보 및 통신 기술을 책임감 있고, 능숙하고, 창의적인 사용자가 되는 것이다. 다음은 CAS[3][4]의 교육과정을 설명한다.

이를 위해서 초등학교 1학년과 2학년에는 다음과 같은 성취를 할 수 있다.

알고리즘을 이해해야 한다. 디지털 장비에서 프로그램이 어떻게 구현되어 있고, 정확한 명령어로 실행된다는 것을 알아야 한다. 간단한 프로그램을 만들고 수정할 수 있다. 논리적인 추론을 이용하여 간단한 프로그램의 동작을 예측한다. 디지털 콘텐츠를 정보기술을 이용하여 목적에 적합하게 만들고, 구성하여 저장하고, 다루고 검색한다. 학교 이외에도 정보 기술이 많이 이용되고 있다는 것을 인식한다. 개인정보를 침해하지 않게 기술을 안전하게 이용한다.

단계 2인 초등학교 3학년에서 5학년까지는 다음과 같은 것을 한다.

특정목표를 수행하기 위한 프로그램을 설계하고, 코딩하고 수정한다. 이때에서 피지컬 시스템을 제어하고 흉내도 낼 수 있어야 하며, 큰 문제를 더 작은 문제로 분해한다. 순차, 선택, 반복문을 사용할 수 있어야 하고 다양한 입·출력 문을 처리할 수 있어야 하며, 변수들을 사용한다. 단순 알고리즘의 동작을 설명하기 위하여 논리적 추론을 할 수 있어야 하고, 알고리즘이나 프로그램에서 에러를 검출하거나 수정 한다. 인터넷을 포함하여 컴퓨터 네트워크를 이해해야 하고, 월드 와이드 웹처럼 다양한 서비스를 어떻게 제공하는지 이해해야 하고, 통신과 협력을 위한 기회를 가져야 한다. 검색기술을 효과적으로 사용할 수 있어야 하고, 검색 결과가 어떻게 선택되고 순위가 부여 되는지 알아야 하고, 디지털 콘텐츠를 평가한다. 인터넷 서비스를 포함한 다양한 소프트웨어를 자신의 목적에 적합하게 선택하여 여러 가지를 조합하여 사용한다. 또한 데이터와 정보를 수집, 분석, 평가, 표현해서 어떤 목표를 달성하기 위해 생성하는 디바이스에서 운영되는 소프트

웨어를 활용해 프로그램, 시스템, 콘텐츠를 설계하고 제작한다. 기술을 안전하고 책임감 있게 사용하는 행동을 하게 한다.

2.3 독일의 컴퓨터 교육과정

독일의 컴퓨터 교육에 대한 기본적인 구조는 콘텐츠 차원과 프로세스 차원으로 나누어진다. 각 차원마다 5개의 영역을 다음과 같이 구성하였다.

콘텐츠 영역은 정보와 데이터(Information and Data), 알고리즘(Algorithms), 언어와 오토마타(Languages and Automata), 정보과학 시스템(Informatics Systems), 정보-사람-사회(Informatics, Man, and Society)의 5개의 영역이고, 프로세스 차원은 모델링과 구현(Modelling and implementing), 추론과 평가(Reasoning and evaluating), 구조화와 상호연관(Structuring and interrelating), 의사소통과 협력(Communicating and cooperating) 및 표현과 해석(Representing and interpreting)[7]으로 나눈다.

데이터와 정보에 대해서 모든 학년의 학생들은 정보와 데이터의 내용을 이해할 뿐만 아니라 다양한 데이터의 표시 형식을 알아야 한다. 이를 위해서 초등학교 5학년부터 7학년까지 배우야 할 내용은 다음과 같다.

정보를 표현할 때에 다른 의미와 품으로 표현한다. 그래픽을 표현할 때에 픽셀 그래픽과 벡터 그래픽을 구별한다. 폴더 구조와 같은 트리 구조를 안다. 그래프를 사용하여 연결된 문서 구조를 안다. 클래스, 객체, 속성, 속성 값이라는 용어를 안다. 또한, 데이터에 대한 동작(또는 작업)을 이해하고 표시한 정보를 해석할 수 있고, 데이터에 적당한 동작을 수행한다.

알고리즘에 대해서 모든 학년의 학생들은 다양한 분야의 문제해결을 위해서 알고리즘을 알고 주어진 알고리즘을 읽고 해석한다. 또한 알고리즘을 설계하고 구현한다.

언어와 오토마타에 대해서 모든 학년의 학생들은 공식 언어를 사용하여 컴퓨터 과학 시스템과 상호 작용할 수 있고 문제 해결한다.

컴퓨터 과학 시스템에 대해서 모든 학년의 학생들이 컴퓨터 과학 및 시스템의 구조와 기본적인 동작을 이해하고, 목적에 적합하게 컴퓨터 과학 시스템을 사용할 수 있고, 더 많은 컴퓨터 과학 시스템을 알 수 있다.

컴퓨터 과학, 인간과 사회에는 모든 학년의 학생들이 컴퓨터 과학 시스템과 사회 사이의 상호 작용을 알 수 있어야 하고, 컴퓨터 과학 시스템을 다루는 의사 결정을 할 때 사회적 규범에 따라 행동할 수 있고 컴퓨터 과학 시스템의 위험한 사용에 적절하게 대응한다.

모델링 및 구현에서는 모든 학년의 학생들이 주어진 상황에 정확한 정보 시스템 모델을 만들고, 적절한 도구를 사용하여 모델한 것을 구현한다.

정당화 및 평가에 대해 모든 학년의 학생들은 정보 이슈들을 질문할 수 있어야 하고, 정보 시스템의 문제에 대한 가정을 할 수 있다. 또한, 컴퓨터 과학 시스템의 사용 결정을 정당화할 수 있고, 정보학 문제를 평가하기 위한 기준을 적용한다.

구조화 및 네트워킹에 대한 모든 학년의 학생들은 이슈를 재배치 또는 정렬을 통해 구조화하고, 컴퓨터 과학의 내외적으로 연결한다.

의사소통 및 협력에 대한 모든 학년의 학생들은 정보 문제에 대해 전문적으로 의사소통할 수 있고, 정보학 문제를 해결하기 위해서 협력할 수 있으며, 소통과 협력을 위한 적절한 도구를 사용한다.

표현 및 해석에 대한 모든 학년의 학생들은 사실의 다른 표현을 해석할 수 있어야 하고, 정보 문제를 설명할 수 있으며, 정보를 표현할 때에 적절한 형태를 선택한다.

3. 우리나라 컴퓨터 교육과정과 비교 설명

3.1. 우리나라 컴퓨터 교육과정

3.1.1 2015 국가 교육과정[8]

우리나라 컴퓨터 교육과정은 한국정보교육학회 등에서 초등학교 1학년부터 실시하고, 독립교과로 구성하자는 주장이 많았지만[10], 2015 개정교육과정에서는 초등학교 5학년부터 6학년까지 컴퓨터 교육을 실시하도록 되어 있고[16], 각 내용은 2019년부터 실질적으로 소프

트웨어 교육을 실시하게 되어 있다. 교육부 교육과정의 기본 틀은 다음 <표 1>과 같이 되어 있다. 소프트웨어 교육이 기술 시스템 영역으로 되어 있고, 일반화된 지식으로는 ‘통신 기술은 정보를 생산, 가공하여 다양한 수단과 장치를 통하여 송수신하여 공유한다.’로 되어 있고, 내용 체계는 소프트웨어 이해, 절차적 문제 해결, 프로그래밍 요소와 구조로 되어 있다. 이는 아무리 보아도 소프트웨어 개발 또는 코딩 교육과정에 맞추어져 있지 소프트웨어를 담고 있는 컴퓨터 시스템 교육과정은 찾아 볼 수 없다.

2015 개정교육과정의 구체적인 성취 기준은 다음과 같다[16].

- 1) 컴퓨터에 사용된 소프트웨어 이외에도 휴대폰, 가전제품, 사물인터넷 제품까지 여러 상황에서 사용되는 소프트웨어를 탐색해 보고 우리 생활에 미치는 영향을 이해한다.
- 2) 절차적 사고란 문제를 효율적으로 해결하기 위해 문제를 작은 단위로 나누고, 각각의 문제를 단계별로 처리하는 사고 과정으로, 일상생활 속의 사례들을 찾아보고 절차적 사고 과정을 문제 해결에 적용한다.
- 3) 블록 기반의 교육용 프로그래밍 도구를 활용하여 기초적인 프로그래밍 과정을 체험 하고 자신만의 간단한 프로그램을 만들어 본다.
- 4) 수치 값을 입력하여 덧셈이나 뺄셈의 결과를 출력하거나, 복수의 문자열을 입력 하여 두 문자열을 서로 연결한 결과를 출력하는 프로그램을 만들어 봄으로써, 소프트웨어의 입력, 처리, 출력 과정을 이해한다.
- 5) ‘순차’는 명령문을 위에서 아래로 하나씩 순차적으로 수행하는 과정이며, ‘선택’은 주어진 조건에 따라 명령문을 선택적으로 수행하는 과정이다. ‘반복’은 명령문을 특정 횟수만큼 반복하거나, 주어진 조건이 만족할 때까지 반복하는 과정이다. 일상의 문제를 해결하는 프로그램을 만드는 기초 과정을 통해 위 프로그램의 3가지 구조를 이해한다.

3.2. 비교 분석

3.2.1 학습 시기

21세기 지식 정보 사회에서 지식 정보를 구성하는 원

리를 일찍 아는 것은 매우 중요하고, 학생들의 생활 속에서 교육을 강조하는 것 또한 매우 중요하다. 요즘 학생들은 매우 어린 나이에 디지털 정보를 접하고 디지털 정보를 이용하여 자신의 목적에 맞게 사용하고 있다. 이런 관점에서 컴퓨터 교육은 일찍 하는 것이 매우 중요하다.

미국, 영국, 독일에서의 컴퓨터 교육의 시기는 다음과 같다.

미국의 교육과정을 정의하는 기관인 ACM의 CSTA에서는 유치원부터 초등학교 2학년까지의 학습 요소와 초등학교 3학년부터 5학년까지의 학습 요소들을 정의한 것을 보면 유치원부터 컴퓨터 교육을 실시하고 있다. 또한, ISTE의 NETS에서도 유치원부터 학습 요소들을 정의하는 것을 보면 유치원부터 컴퓨터 교육을 권고하고 있다.

영국에서도 국가 수준의 필수 교과로 Computing이라는 과목을 정의하고 초등학교 1학년부터 컴퓨터 교육을 실시하고 있다.

독일의 경우에는 2008년도 자료를 살펴보면 초등학교 5학년에서 7학년까지의 학습 요소들을 정의한 것을 보면 초등학교 5학년부터 컴퓨터 교육을 실시하고 있다.

우리나라의 경우에 2015 개정교육과정에서 실과 교과에 구성되어 있기 때문에 초등학교 5학년부터 이론적으로 컴퓨터 교육이 가능하다.

각 국가별로 컴퓨터 교육 실시 시점은 다음 <Table 1>과 같다.

<Table 1> Learning Startup time

Nation	Startup
USA(ACM)	kindergarden
USA(ISTE)	kindergarden
U.K	Grade 1
Germany	Grade 5
Korea	Grade 5

3.2.2 학습 내용

21세기 미국과 영국의 컴퓨터 교육은 지식 정보를 이용하는 ICT 활용 위주의 교육에서 벗어나 지식 정보를 구성하는 소프트웨어를 직접 만드는 교육을 강조하고 있다.

ACM의 CSTA 교육과정을 분류하면 컴퓨팅 시스템, 네트워크와 인터넷, 알고리즘과 프로그래밍, 데이터와 분석 그리고 컴퓨팅의 영향 영역으로 되어 있다.

ISTE의 NETS에서 학생들의 교육 목표를 능력 있는 학습자, 디지털 시민정신, 지식 구성자, 혁신적인 설계자, 컴퓨테이셔널 사고자, 창의적인 소통자, 글로벌 협력자를 컴퓨터 교육을 통해서 양성하는 것을 목표로 한다는 것을 알 수 있다.

영국의 국가 수준 교육과정 내용은 알고리즘, 프로그램, 데이터, 컴퓨터 시스템, 통신과 인터넷 등의 내용으로 분류한다.

영국의 computing 교과목에는 큰 영역으로 컴퓨터 과학 영역, 정보 기술 영역, 디지털 문해영역으로 구분한다.

독일의 경우에는 컴퓨터 교육의 분야를 정보와 데이터, 알고리즘, 언어와 오토마타, 정보과학 시스템, 정보-사람-사회의 5개의 영역으로 구성된다.

이들을 영국의 교과 분류표에 표로 정리 하면 다음 <표 3>과 같다. 영국의 기준으로 CS, IT, DI 분야에는 독일의 교육과정들이 해당 영역에 다 포함되어 있다고 볼 수 있고, 미국의 ACM과 ISTE의 교육과정들을 분석하여 보면 영국의 기준인 CS, IT, DL 분야를 다 포함하고 있다. 그러나 우리나라 교육과정에서는 CS 분야에서 컴퓨터 프로그램 부분과 알고리즘 부분만 해당 된다고 볼 수 있다.

따라서 우리나라 컴퓨터 교육과정은 영국의 교육과정의 일부분만 포함하고 있다.

<Table 2> U.K. viewpoints

Nation	CS	IT	DL
USA(ACM)	0	0	0
USA(ISTE)	0	0	0
U.K	0	0	0
Germany	0	0	0
Korea	0		

미국의 ACM 교육과정의 관점에서 분석하면 다음 <표 4>와 같다. 우리나라 교육과정은 ACM의 교육과정에서 컴퓨팅 시스템, 네트워크와 인터넷 부분과 데이터와 분석은 포함되지 않고, 알고리즘과 프로그래밍, 그리고 컴퓨팅의 활용 부분만 수업 내용에 포함된다.

<Table 3> ACM. viewpoints

Nation	computer systems	Networks and the Internet	Algorithms and Programming	Data and Analysis	Impacts of Computing
USA(ACM)	0	0	0	0	0
USA(ISTE)	0	0	0	0	0
U.K	0	0	0	0	0
Germany	0	0	0	0	0
Korea	0				0

우리나라 컴퓨터 교육과정을 독일의 교육과정 측면에서 분석하여 보면 우리나라 교육과정은 독일의 정보와 데이터(), 정보과학 시스템(), 정보-사람-사회() 영역은 포함하지 않고 부분적으로 알고리즘(), 언어와 오토마타() 영역을 포함한다.

<Table 4> Germany viewpoints

Nation	Information and Data	Algorithms	Languages and Automata	Informatics Systems	Informatics Man, and Society
USA(ACM)	0	0	0	0	0
USA(ISTE)	0	0	0	0	0
U.K	0	0	0	0	0
Germany	0	0	0	0	0
Korea	0	0			0

우리나라 컴퓨터 교육과정에서는 학생들을 양성하는 목표는 ISTE의 NETS 교과 목표 중에 디지털 시민정신, 컴퓨테이션 사교자 양성에는 포함되고 있지만 능력있는 학습자, 지식 구성자, 혁신적인 설계자, 창의적인 소통자, 글로벌 협력자 양성에는 미흡하다.

4. 결론 및 제언

영국의 경우 2014년부터 실시하고 있는 교육과정, 미국의 경우에 2016년도에 ACM과 ISTE의 양 기관의 컴퓨터 교육과정 및 독일의 경우 2008년도 발표한 교육과정을 보면 초등학교부터 컴퓨터 과학의 교육, 정보기술의 교육 및 디지털 리터러시 교육을 실시하고 있고, 초등학교 교육 내용의 구성을 보면 미국의 경우 유치원부터 주1시간 이상은 해야 하고 영국의 경우 영국은 초등

학교 1학년부터 주 한 시간은 해야 하고 독일의 경우도 초등학교 5학년부터 주 한 시간 이상은 해야 한다.

첫 번째는 우리나라 초등학교 컴퓨터 교육과정은 실과교과 초등학교 5-6학년 군에서 실시하게 되어 있기 때문에 미국과 영국의 경우에는 초등학교 1학년 또는 유치원에서 실시하고 있다는 점과 비교하면 매우 늦은 나이에 컴퓨터 교육을 실시하고 있다. 이는 학생들이 디지털을 접근하는 시기에 비해서 교육 실시가 매우 늦다는 것이다. 따라서 빠른 시기에 컴퓨터 교육을 실시할 필요가 있다.

두 번째는 교육 시수가 적다는 점이다. 우리나라는 2015 개정교육과정에 의하면 17시간을 실시하게 되어 있기 때문에 초등학교 교육과정상의 총 시수에 비해 너무 적은 시수이고, 미국, 영국, 독일에 비해서도 적은 시수가 배정되어 있다. 더욱이 2000년부터 우리나라에서 실시하였던 정보통신교육지침에 의해 초등학교 1학년부 터 매주 한 시간씩 교육하게 되어 있는 것에 비해서 12분의 1정도 배정되어 있다. 선진국 수준 또는 2000년도 부터 실시한 정보통신교육지침에 준하는 매주 2시간씩 컴퓨터 교육을 실시해야 한다.

마지막으로 우리나라 컴퓨터 교육이 소프트웨어 교육과 정보 윤리의 일부분만 하게 되어 있다는 점이다. 컴퓨터 시스템과 소프트웨어를 모르고 소프트웨어를 개발하는 교육을 하는 것은 매우 잘못된 교육이다. 컴퓨터 과학의 원리와 개념, 네트워크의 원리와 개념, 정보의 개념과 원리 등을 컴퓨터 교육에 추가해야 하는 일이 시급하다.

참고문헌

[1] Barr, V. & Stephenson, S. (2011, March). Bringing computational thinking to K-12: What is involved and what is the role of the computer science community? ACM Inroads, (2)1, 48-45.
 [2] Bers, M. U., Flannery, L., & Kazakoff, E. R. (2014, March). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. Computers and Education, 72, 145-157.
 [3] CAS (2013A). Computing in the national curriculum:

- A guide for primary teachers. Computing At School.
- [4] CAS (2013B). Computing in the national curriculum: A guide for secondary teachers. Computing At School Computing At School.
- [5] Deborah Seehorn, Stephen Carey, Daniel Moix, Brian Fuschetto, Irene Lee, Dianne O'Grady-Cuniff, Chris Stephenson, Anita Verno (2011). CSTA K-12 Computer Science Standards Revised 2011.
- [6] Deborah Seehorn, Stephen Carey, Daniel Moix, Brian Fuschetto, Irene Lee, Dianne O'Grady-Cuniff, Chris Stephenson, Anita Verno (2016). CSTA K - 12 COMPUTER SCIENCE STANDARDS REVISED 2016 CSTA STANDARDS TASK FORCE.
- [7] German Informatics Society (GI) (2008). Grundsätze und Standards für die Informatik in der Schule. Bildungsstandards Informatik für die Sekundarstufe I (2008) (in German).
- [8] Grover, S. & Pea, R. (2013, January/February). Computational thinking in K-12: A review of the state of the field. *Educational Researcher*, 42(1), 38-43.
- [9] Hong, M. H. (2014). A Study on ICT Competences of Turkey and Korea Focus on PISA 2009 and PISA 2012. *Journal of The Korean Association of Information Education*, 18(1), 151-160.
- [10] Kim Kapsu, etc (2014). A Study on Contents of Information Science Curriculum. *Journal of The Korean Association of Information Education*, 18(1), 161-171.
- [11] Kim, Kapsu (2015). A Study on ICT Competences of Korean Students Focus on PISA 2009 and 2012. *Journal of The Korean Association of Information Education*, 19(2), 233-242.
- [12] Kim, Kapsu (2010). A Study on Programming Language Instruction Strategies of Improving the creative and logical thinking for Elementary Students. *JOURNAL OF The Korean Association of information Education*, 8(2), 63-70.
- [13] Kim, Kapsu (2016). A Recognition Analysis of Elementary Teachers for Software Education of 2015 Revised Korea Curriculum. *Journal of The Korean Association of Information Education*, 20(1), 47-56.
- [14] ISTE standards (2016). Redefining learning in a technology-driven world A report to support adoption of the ISTE Standards for Students June 2016.
- [15] Lye, S. Z., & Koh, J. H. L. (2014, December). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior*, 41, 51-61.
- [16] Ministry of Education (2015). The revised national curriculum 2015 for Primary and Secondary Schools, Vol.2015-75, No. 10.
- [17] Ministry of Education (2000). Information and Communication Technology Education Guidelines.
- [18] Ministry of Education (2005). Information and Communication Technology Education Guidelines.
- [19] OECD (2013). PISA2012 Results. www.oecd.org/pisa.
- [20] Tucker, A., Deek, F., Jones, J., McCowan, D., Stephenson, C., and Verno, A. (2002). A model curriculum for K-12 computer science, Report of the ACM K-12 Education Task Force Computer Science Curriculum Committee.
- [21] Twining, P., Raffaghelli, J., Albion, P., & Knezek, D. (2013, August 5). Moving education into the digital age: the contribution of teachers' professional development. *Journal of Computer Assisted Learning*, 29(5), 426-437.
- [22] Wing, J. M. (2008, July 31). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society*, 366, 3717-3725.

저자소개



김 갑 수

1985 서울대학교계산통계학과(학사)

1987 서울대학교 계산통계학과 전산학전공(석사)

1996 서울대학교 계산통계학과 전산학전공(박사)

1987~1992 삼성전자 사원-과장

1995~1998 서경대학교 전임강사-조교수

1998~현재 서울교육대학교 컴퓨터교육과 조교수-교수

관심분야: 컴퓨터 교육, SW 공학, 정보 영재, 기능성 게임

e-mail: kskim@snue.ac.kr

