

고등학교 컴퓨터 교과 수준별 교육과정 설계

김애경*, 김태윤**
고려대학교 컴퓨터학과
e-mail:kyung605@orgio.net

Plan the Computer Science Curriculum of High school students based on their Computer literacy levels

Ae-kyung kim*, Tai-Yun Kim**,
Dept of Computer Science, Korea University

요 약

정보과학의 발전은 급속도로 발전하고 있다. 가장 기초적인 정보 과학의 교육이 이루어지는 현행 중등 컴퓨터 교육은 어떠한 교과보다도 창의적인 자기 주도적 학습 및 개별학습이 필요하다. 컴퓨터 활용을 위한 교육보다는 컴퓨터 과학으로서의 컴퓨터 교육이 필요할 뿐 아니라 학생의 학습환경, 학습동기 및 흥미, 선행학습정도, 교과 수업의 참여도 등에 따라 학생 개개인의 능력 차이를 고려한 학습자의 개인의 학습 능력에 맞는 수준별 교육이 필요하다. 고등학교 컴퓨터 교과 교육에서의 컴퓨터 과학으로서의 교육효과를 위해 학습자 학습 능력에 맞는 교육과정을 설계하였고 보충학습과 심화학습을 위한 자기 주도적 학습 방법인 웹에서의 학습 방법을 도입하여 학습의 효과를 높이고자 하였다.

1. 서론

현대사회의 모든 분야에 정보 통신 기술이 관련되어 있어서 정보과학의 발전을 위해서는 컴퓨터 교육이 중요하다. 컴퓨터 과학으로서의 교육보다는 컴퓨터 활용을 위한 교육으로 이루어지고 있다. 창의적이고 자기 주도적인 학습을 위해서는 학생의 학습환경, 학습동기 및 흥미, 선행학습정도, 교과수업의 참여도, 진로 등을 고려하여 학습자를 분류하여 학생능력에 맞는 컴퓨터과학교육을 위한 수준별 교육과정을 설계하였다. 또한 보충학습과 심화학습을 위하여 웹 상에서의 자기 주도적 학습방법을 채택하여 학습의 효과를 높여 정보산업의 발전을 가져오는 기초 교육으로서 교육의 효과를 기대해 본다.

2. 수준별 교육과정의 형태

수준별 교육과정은 학습자의 학습 능력 수준과 요구에 대응하는 차별적 선택적 교육을 제공하는 데 근본적인 의의를 두고 있다.

여기서 학습 속도의 수준을 고려한 교육과정의 편성·운영은 '단계형'이 적합하고 학습 깊이 수준을 고

려한 교육과정 편성 운영은 '심화 보충형'이 적합하다. 그리고 소질과 적성을 충분히 살릴 기회를 제공하기 위해서는 '과목 선택형'이 적합하다[1].

2.1 현행수준별 교육과정의 형태

서울시 5개 특목고와 95개 인문계고등학교의 교육계획서를 참고하여 조사해 본 결과 단계형은 15개교, 단계형과 심화 보충형은 4개교, 심화 보충형을 실시하고 있는 학교는 2개교였다. 상, 중, 하의 3단계로 실시하는 학교는 14개교이고 주로 영어, 수학과목만 실시하고 컴퓨터교과를 실시하는 학교는 없었다.

2.2 현행 수준별 판별기준

영어, 수학과목을 수준별 교육과정을 실시하고 있는 21개교의 수준별 판별기준은 정기고사로 실시하고 있었다. 현행 컴퓨터 교과 정기고사의 평가 방법은 지필고사(50%), 실기고사(40%), 교사 학습태도 평가(10%) 반영하는 비율이 가장 높았다.

2.3 현행 수준별 교육과정의 문제점

첫째, 운영구조의 문제점으로 정기고사로 결정하기 때문에 학생들의 흥미나 적성 희망사항 등이 고려되지 않고 학교 실정에 따라 수준반을 편성하므로 형식적이다. 둘째, 하위반의 학습기회가 점점 적어지게 되고 정서적으로 불건전한 정신상태를 갖게 한다.

셋째, 수업내용이 교과서 위주로 되어 있어 수업내용의 수준별 차이가 없고, 수준별 학급 간 수업방법의 차이가 없다.

넷째, 심층적 다양한 능력을 판별하는 기준의 평가가 이루어지지 않아 신뢰도에 문제점이 될 수 있다[2] 다섯째, 컴퓨터교과에서는 수준별 교육이 이루어지고 있지 않다.

여섯째, 학생들의 능력의 차이에 따른 보충학습, 심화학습이 이루어지고 있지 않다.

3. 수준별 교육과정 설계

컴퓨터 과학교육으로서의 교육효과를 위해서는 학습자의 흥미와 적성을 고려하고 선행학습의 상태와 학습태도, 진로 등을 고려하여 학습자를 분류하고 교과시간외에 웹에서의 보충, 심화학습을 실시하는 수준별 교육과정을 설계하였다.

3.1 수준별 교육과정의 방향

첫째, 컴퓨터교육은 컴퓨터에 관한 교육(Learning about Computer), 컴퓨터를 통한 교육(Learning through Computer), 컴퓨터와 함께 하는 교육(문제해결 Learning with Computer), 컴퓨터를 이용한 교육경영(Learning supper system; CMI)등으로 구분해 볼 수 있다. 두 개념은 실질적으로 밀접한 관계가 있다[5].

둘째, 컴퓨터 교육과정의 방향은 컴퓨터 교과와 타 교과와 연계성이 강화되며, 컴퓨터 산업의 급속한 발전을 반영할 수 있는 교육과정이어야 한다.

셋째, 예체능 분야처럼 컴퓨터 분야의 영재들을 위한 이 분야의 인재양성을 위한 교육과정도 포함되어야 한다[3].

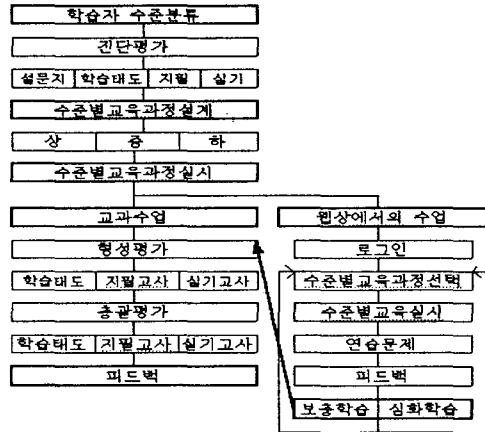
넷째, 컴퓨터 관련분야가 다양하고 복잡하기 때문에 획일화된 프로그램보다는 특성화된 프로그램이 요구되고 있다. 컴퓨터과학의 경우 예술분야까지도 포함된 종합학문이기 때문에 특성화하는 것이 바람직하다[3].

다섯째, 학교 컴퓨터 교육의 지향수준은 프로그래밍모형과 컴맹탈피의 모형, 도구적인 모형의 수준을 거친 Course로서 활용과 응용의 수준인 복합적인 문제의 처리 능력을 훈련한다.

여섯째, 정보교과 교육과정은 내용중심보다 능력중심의 목표 설정이 요구되며 교과내용도 능력중심의 항목

들로 학생들의 수준을 고려하여 구체적이어야 하며 타 교과와 체계적인 연계를 고려해야한다. 교수·학습 과정에서도 협동학습 등의 활동중심의 수업진행이 요구된다[4].

3.2 수준별 교육과정의 진행과정



(그림 1) 수준별 교육과정의 진행과정

3.3 학습자 수준별 분류

진단평가를 실시하여 수준별 학습자를 분류한다. 진단평가는 설문지를 통해 학습환경과 흥미, 적성, 진로, 희망 등을 조사하여 학습태도(10%), 지필고사(50%), 실기고사(40%)를 실시하여 그 결과와 개인별 상담을 통해 학습자를 수준별로 분류한다.

학습태도는 흥미도, 이해도, 참여도, 자신감, 성취도 등을 평가한다. 지필고사는 현행 자격시험에서 추출해 상위 자격과 구분하여 난이도를 상, 중, 하로 하고 문항의 수준을 상(20-25%), 중(50-60%), 하(20-25%)의 배점 비율로 배점 기준을 정한다. 실기고사는 실제 활용 사례에 대한 관찰과 실기 능력에 중점을 두고 과정을 중시하여 평가 해야한다[6]

지식단위 영역은 정보사회, 컴퓨터구조, 운영체제, 소프트웨어, 컴퓨터통신, 프로그래밍의 6개 영역으로 분류하여 추출하였다.

3.4 컴퓨터 과학으로서의 수준별 교육과정 설계

컴퓨터 과학으로서의 교육과정을 설계하기 위해서는 본 연구에서는 'ACM Model High School Computer Science Curriculum'을 기본으로 하여 <표 1>과 같이 지식단위를 63개로 추출하여 고등학교의 교과서 내용과 자격증시험과목 내용과 대학의 컴퓨터학과의

Curriculum과의 공통된 지식단위와 공통되지 않은 지식단위를 <표 2>와 같이 추출하였다. 고등학교 교과과정과 자격증 시험과목 중 중복되는 교과내용은 수준별 교육과정 중 하위반에 편성하고 'ACM Model High School Computer Science Curriculum'[6],의 'Recommendtopic' 과 대학의 Curriculum의 중복되는 지식단위는 중위반에 첨가 편성하고 'Optional topic'과 중복되지 않는 지식단위는 상위반에 첨가 편성하였다. 각 수준별 지식단위는 <표 3>과 같이 지식단위의 번호로 표시하였다.

<표 1> ACM의 지식단위

영역	번호	지식단위
1 Algorithms	1	Algorithms in the world
	2	Techniques used to design and represent algorithms
	3	Examples of important algorithms
	4	Basic problem solving concepts, e.g. selection, iteration, recursion
	5	Methods to test algorithms
	6	Basic data structures
	7	Boolean algebra
	8	Characteristics of an algorithm, e.g. correctness, finiteness
	9	Complexity of algorithms
	10	Limits of computability
2 Programming Languages	11	Introduction to a specific computer language
	12	Concept of sequence, selection, and repetition
	13	Levels of computer languages
	14	Competence in high level language
	15	Compilers and interpreters
	16	Program modularization, e.g. functions, procedures
	17	Reading and analyzing existing programs
	18	Comparison of languages: procedural, structured, functional, object-oriented, parallel
	19	Modifying existing programs
	20	Program verification, e.g., pre- and post- conditions, loop invariants
	21	Theoretical machines and formal languages
3 Operating Systems and User Support	22	Command language and its use
	23	Files and disk management
	24	Telecommunications, local and wide area networks
	25	Human-computer interaction
	26	Working with large complex systems
	27	Communication network implementation, e.g., graphs, protocols
	28	Memory management and virtual memory
	29	Operating system functions, e.g., task scheduling, interrupts, buffered I/O
	30	Basic computer model, e.g., CPU, memory, I/O
4 Computer Architecture	31	Basic data representation: numbers vs. characters, ascii vs. non-ascii
	32	Logic, gates, and circuits
	33	Data representation, e.g., bits and bytes, binary numbers, real numbers
	34	Accuracy of numerical computation, von Neumann stored program model, opcodes, registers, clock, fetch-execute cycle
	35	Physical disk organization
	36	Sequential and parallel processing
	37	Special data representation, e.g., graphics, sound
	38	
	39	Data compression

5. Social, Ethical, and Professional Context	39	Impact of technology on today's society
	40	Ethics in an electronic community
	41	Team solution of problems
	42	Future of computer technology
	43	Risks and liability in computing, viruses
	44	Computer support of the disabled
	45	Software, public domain and private
	46	Privacy, reliability and system security
	47	Uses, misuses, and limits of computer technology
6. Computer Applications *	48	Electronic crime: stealing and spying, Intellectual property, infringements and penalties
	49	Legal issues
	50	Computer aided design/computer aided manufacture
	51	Computer speech, music synthesis, and art
	52	Database systems
	53	Electronic mail and bulletin boards
	54	Multimedia
	55	Presentation graphics
	56	Scientific analysis, e.g., Mathematica, Matlab
7. Additional Topics *	57	Spreadsheets and data analysis
	58	Word processing and desktop publishing
	59	Artificial Intelligence, e.g., games, expert systems, robotics, knowledge representations
	60	Computational science, e.g., scientific visualization, modeling
	61	Graphics, e.g., image generation, two- and three-dimensional animation
	62	Simulation and virtual reality
	63	Software engineering, e.g., system development, software development cycle, modeling and diagramming

<표 2> ACM과 중복되지 않는 단위

영역	번호	지식단위
정보사회	64	전자상거래
운영체제	65	마이크로 프로세서 구조 및 프로그래밍
컴퓨터 구조	66	디지털 시스템 설계
	67	디지털 시스템 실험
	68	분산처리
	69	컴퓨터 공학 소개(s/u)
	70	일반전자공학
	71	기계학수
	72	VLSI 설계이론
소프트웨어	73	CAM
	74	경영과학S/W응용
	75	시스템소프트웨어 설계 및 실험
	76	소프트웨어 프로젝트
데이터통신	77	무선통신
	78	인터넷 응용
프로그래밍	79	문제해결시범
	80	심플렉프로그래밍
	81	생신시스템 설계

고등학교 교과과정과 자격증 시험과목 중 중복되는 교과내용은 수준별 교육과정 중 하위반에 편성하고 'ACM Model High School Computer Science Curriculum'[6],의 'Recommendtopic' 과 대학의

Curriculum의 중복되는 지식단위는 중위반에 첨가 편성하고 'Optional topic'과 중복되지 않는 지식단위는 상위반에 첨가 편성하였다.

각 수준별 지식단위는 <표 3>과 같이 지식단위의 번호로 표시하였다.

<표 3> 수준별 지식단위 편성

영역	하	중	상
정보사회	39,40,41	42,43,44,45,46,47,48	49,64
운영체제	22,23,	25,26	28,29
컴퓨터구조	30,31,32,1,2,34,	33,34	35,36,37,38,65, 66,67,68,69,70, 71,72
소프트웨어	50,51,52,53,54	55,56,57,58,59,60	73,74,75,76
데이터통신	24,	27	77,78
프로그래밍	11,12,13	14,15,16,17	18,19,20,21,79, 80,81

수준별 교육과정의 영역별 교육 목표, 교육 내용, 교육 방법은 전문 교육인 대학교육과의 연계성을 고려하여 설정하였다.

교육목표와 교육내용은 <표 3>의 지식단위는 기본 개념을 이해하고 상위단위에서는 응용력과 적용력에 목표를 두었다.

교육방법은 상위단위에는 토의 학습, 프로젝트학습, 재택학습, 원격학습 등에 중점을 둔다.

평가 방법은 [그림 1]과 같이 형성평가와 총괄평가를 실시하여 수준별 학습분류 방안의 방법과 같이 실시하여 피드백을 통하여 새롭게 학습자를 수준별로 분류하여 교육과정이 계속 진행되게 한다.

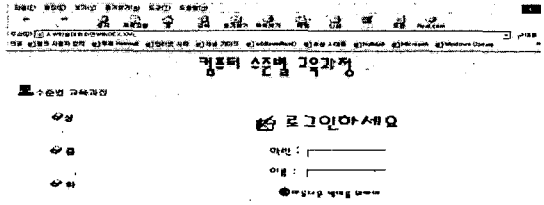
4. 웹 상에서의 교육과정

웹 상의 교육과정은 정기교과 과정에서 실시되는 내용을 웹 상에 탑재해 보충, 심화 학습을 할 수 있게 설계하였다.

(그림 1)과 같이 본인이 선택하여 피드백으로 반복하여 수업할 수 있으며 연습문제를 풀어 자신의 학습효과를 확인할 수 있다.

화면구성은 학습자 로그인 화면과 교수자 수준별 교안 작성 화면으로 나누어진다. (그림 2)는 학습자 로그인 화면이다. 스타일시트는 XSL을 사용하고 XML로 설계하였다.

(그림 3)은 수준별 교육과정의 학습지도안 문서를 XML DTD로 표현한 것이다.



(그림 2) 화면구성

```
<?xml version="1.0" encoding="euc-kr"?>
<!ELEMENT DOCUMENT (수준별학습지도안)*>
<!ELEMENT 수준별학습지도안
(수준반, 단원명, 지도일시, 교육내용, 연습문제)>
<!ELEMENT 수준반 (#PCDATA)>
<!ELEMENT 단원명 (#PCDATA)>
<!ELEMENT 지도일시 (#PCDATA)>
<!ELEMENT 교육내용 (#PCDATA)>
<!ELEMENT 연습문제 (#PCDATA)>
```

(그림 3) 수준별교육 학습지도안DTD

5. 결론

본 연구는 컴퓨터 과학으로서의 교육을 위해서는 학습자의 개인의 능력과 흥미, 적성, 진로 등을 고려한 수준별 교육과정으로서 고등학교 교과 과정, 자격증 시험과목, 현재 대학의 컴퓨터학과 교육과정, ACM의 고등학교 교육과정 등을 고려하여 컴퓨터 과학의 교육이 대학까지의 연계 교육을 위한 교육과정을 편성할 수 있었다. 또한 웹 상에서의 보충, 심화 학습은 학습의 효과를 높일 수 있다.

향후 전 교육과정을 시간과 공간을 초월한 원격교육 과정으로 실시하면 높은 교육효과를 기대할 수 있다.

참고문헌

[1] 박용철, 수준별 학습을 위한 학습자 평가 시스템의 설계 및 구현, 대구효성카톨릭대학교 교육대학원 석사학위논문, 1999.
 [2] 정미향, 수준별 교육과정 운영 현상에 대한 문화기술적 연구, 고려대학교 대학원 교육학과 박사학위논문, 1999.
 [3] 백두권, 정보통신기술의 발전과 컴퓨터 교육, 정보과학회지, 제 14권, 제 9호, 1996.
 [4] 홍지영, ICT 활용 교육활성화에 따른 정보교과 교육과 고찰, 컴퓨터교육학회지, 제 4권, 제 2호, 2001.
 [5] 이태욱, 컴퓨터교육론. 좋은소프트사, 1999.
 [6] http://www.acm.org/education/hscur