

프로그래밍 학업성취도에 관한 사례연구: 하드웨어 교과과정 중심으로

이승우¹

¹서경대학교 전자공학과

접수 2014년 6월 12일, 수정 2014년 6월 30일, 게재확정 2014년 7월 7일

요약

본 논문에서는 H/W 전공자들의 프로그래밍 수준을 S/W 전공자들과의 상대적 비교 분석을 통하여 파악하고자 첫째, H/W와 S/W 분야 졸업예정자들에게 C언어와 C++언어에 관한 프로그래밍 학업성취도를 비교 측정하였다. 둘째, H/W와 S/W의 교과과정 편성을 비교 분석하여 프로그래밍 학업성취도 향상에 관련된 정보를 교과과정을 통하여 도출하였다. 셋째, 수학적 학습능력이 프로그래밍 학업성취도에 어떠한 영향을 주는지 파악하기 위해서 수학 교과목과 프로그래밍 교과목간의 학업성취도에 관한 회귀분석을 실시하여 그 관련성을 분석했다. 이를 통하여 향후 H/W 교과과정에서 학과차원의 특수성과 실정을 반영한 효율적인 프로그래밍 학업성취도 증대 방안에 관하여 제안하였다.

주요용어: 교과과정, 프로그래밍, 학업성취도, C언어.

1. 서론

H/W 및 S/W 분야의 경우 최근 20년 동안 실로 놀란 만한 발전을 이루어 왔고, 이러한 발전이 정치, 경제, 사회는 물론 개인의 삶의 방식까지도 영향을 줄 정도로 전 인류 생활 전반에 크나 큰 영향을 불러 일으키고 있다.

특히 H/W 및 S/W 분야에서 프로그래밍은 기본도구로 사용될 뿐만 아니라 필수불가결한 요소이다. 더욱이 다양한 신기술을 통한 새로운 시스템을 구현하는 실질적인 도구이자 기업의 경쟁력 제고에 기여하는 전략적 기본 요소이지만, H/W 및 S/W 분야에서 프로그래밍 개발능력은 현저하게 떨어지고 있는 것이 우리나라 현실이다 (Kim, 2003).

컴퓨터 활용능력의 근간이 되는 프로그래밍 언어 학습의 교육적 효과로서 첫째, 학습자에게 컴퓨터 프로그래밍 작성 과정을 통하여 분석력, 논리력, 창의력, 사고력 등을 바탕으로 학습효과를 극대화시켜서 문제해결력을 향상시켜줄 수 있는 효과적인 학습 환경을 제공해 준다 (Kim, 2002). 둘째, 학습자는 컴퓨터 프로그래밍의 작업 과정에서 발생하는 오류를 학습자 스스로 찾아내고 수정함으로써 문제해결력을 배양시킬 수 있는 효과를 습득할 수 있다 (Lee, 2008).

2010년과 2011년 2년간에 걸쳐 S대학교의 H/W와 S/W 졸업예정자들에게 C언어와 C++언어에 관한 프로그래밍 학업성취도를 측정하였다. C언어 프로그래밍 학업성취도에 있어서 H/W 졸업예정자들의 평균성적이 S/W 졸업예정자들의 평균성적보다 다소 높은 반면 C++언어 프로그래밍 학업성취도에 있어서 H/W 졸업예정자들의 평균성적이 S/W 졸업예정자들의 평균성적보다 다소 낮음으로써, C언어와 C++언어에 대하여 두 집단 간에 통계적으로 유의한 차이를 보이고 있었다.

¹ (136-704) 서울시 성북구 정릉동 16-1, 서경대학교 전자공학과, 교수. E-mail: swlee@skuniv.ac.kr

본 논문의 의의로서 H/W 졸업예정자들의 C언어 프로그래밍 학업성취도가 S/W 졸업예정자들보다 높은 이유를 H/W 교과과정에서 파악하고 그 원인을 분석하고자 한다.

프로그래밍의 실력과 자질을 함양하기 위해서는 논리적 사고력, 창의력 등의 기본 소양을 갖추고 충분한 실습이 요구되어진다. 논리적 사고력 및 창의력 증진은 수학 교육을 통해서 자연스럽게 배양되며, 충분한 실습기회는 전공 교과내용과 관련된 기술을 구현하기 위하여 C/C++언어 프로그래밍과 응용프로그램을 이용하여 설계 및 구현하는 교수-학습 과정을 통하여 제공받는다.

H/W 교과과정의 특징을 S/W 교과과정과의 상대적 비교 분석을 통하여 C언어 학업성취도가 왜 높았는지를 파악하고자, 수학 강좌 수, 수학을 필수도구로 활용하는 강좌 수, C/C++언어 강좌 수, C/C++언어 활용 강좌 수 등을 비교 분석해보았다.

본 논문에서는 연구 내용을 크게 세 가지 관점에서 수행하고자 한다.

첫째, S대학교의 H/W와 S/W 분야의 교과과정 편성 중에서 수학 교과목 및 수학을 필수도구로 활용하는 전공교과목들 (이하 수학 관련 교과목들로 지칭함)과 C언어와 C++언어로 교수-학습한 프로그래밍 교과목들로만 제한하여 그 교과목들을 비교 분석해보고자 한다.

둘째, 2010년부터 2013년까지 4년간에 걸쳐 H/W와 S/W 졸업예정자들에게 C언어와 C++언어에 관한 프로그래밍 학업성취도를 비교 측정하여 H/W 전공자들의 프로그래밍 학업성취도 수준이 어느 정도인가를 분석하고자 한다. 이 분석결과를 토대로 H/W 전공자들의 프로그래밍 학업성취도 현황파악 및 H/W 교과과정 분석을 통하여 C언어와 C++언어 제고방안을 마련하고 대책을 강구하여서 프로그래밍 분야로 관심을 유도하고 프로그래밍 학업성취도 향상에 기여하고자 한다.

특히 2012년과 2013년 2년간에 걸쳐 C언어 프로그래밍 학업성취도 검사를 구문 에러, 논리적인 에러, 시간 제한 등으로 세분화해서 측정하여 본 실험연구의 신뢰도 향상을 도모하고자 한다. 이 분석결과를 통하여 H/W 졸업예정자의 C언어 프로그래밍 학업성취도가 S/W 졸업예정자의 C언어 프로그래밍 학업성취도보다 높으면, 그 이유를 파악하고 H/W 교과과정을 통하여 정보를 도출하고자 한다.

이 분석결과는 위 대학교의 H/W와 S/W의 교과과정 편성 비교 분석의 특징으로 파악하고 학업성취도가 높은 해당 교과과정을 통하여 학업성취도 향상에 관련된 정보를 도출하고자 한다.

셋째, H/W와 S/W 교과과정의 수학 관련 강좌 수를 비교 분석하고 수학 관련 교과목들에서 배운 수학적 학습효과로 인해 발생된 요인이 C언어 프로그래밍 학업성취도에 어떠한 영향을 주는지 파악하기 위해서 수학 교과목과 C언어 프로그래밍 교과목간의 학업성취도에 관한 회귀분석을 실시하여 그 관련성을 파악해보고자 한다.

마지막으로 본 연구의 분석결과에서 도출된 정보를 통하여 향후 H/W 교과과정에서 학과차원의 특성 성과 실정을 반영한 효율적 프로그래밍 학업성취도 제고 추진 방안에 관하여 제안하고자 한다.

2. H/W와 S/W 교과과정의 편성에 관한 비교 분석

S대학교의 H/W 분야의 E학과와 S/W 분야의 S학과의 교육과정은 전공교양과정, 전공기초과정, 전공심화 및 응용과정 그리고 전공실무 및 신기술적용과정으로 구성된다 (Lee, 2012).

본 연구에서는 위 대학교의 H/W와 S/W 교과과정 편성 중에서 수학 관련 교과목들과 C언어와 C++언어로 교수-학습한 프로그래밍 교과목들로만 제한하여 그 교과목들을 비교 분석하면 다음과 같다.

우선, 위 대학교의 H/W 프로그래밍 교과과정은 2013년에 일부 개편되었다. 2013년 이전 H/W 프로그래밍 교과과정 편성은 1학년 2학기에 C프로그래밍, 2학년 1학기에 기초프로그래밍 (고급 C언어 강의), 2학년 2학기에 고급프로그래밍 (C++언어인 객체지향언어 강의)으로 구성되었다. 단, 1학년 1학기에는 C언어 프로그래밍 강좌가 개설되지 않았다. 2013년 이후 H/W 프로그래밍 교과과정 편성은

1학년 1,2학기에 C프로그래밍1,2, 2학년 1,2학기에 고급프로그래밍1,2 (객체지향언어인 C++/JAVA 강의)로 구성되어있다.

위 대학교의 H/W와 S/W 교과과정의 편성에 관한 비교 분석은 다음과 같다.

첫째, 전공교양과정의 편성에 관한 비교 분석은 Table 2.1과 같다.

수학 교과목으로서, H/W와 S/W 교과과정에서는 대학수학1,2로 동일하게 구성되어있다. C언어 프로그래밍 교과목으로서, H/W 교과과정에서는 두 강좌, 즉 2013년 이전 교과과정 편성은 C프로그래밍과 기초프로그래밍이고 2013년 이후 C프로그래밍1,2로 구성되었으며, S/W 교과과정에서는 프로그래밍언어기초, 프로그래밍언어응용으로 동일하게 구성되어있다.

Table 2.1 The comparative analysis on major liberal curriculum of H/W and S/W

Courses	Departmental Courses of H/W		Departmental Courses of S/W
Mathematics Courses	Mathematics 1,2		Mathematics 1,2
C Programming Courses	Before 2013	C Programming	Fundamentals of Programming Language
		Basic Programming	
	After 2013	C Programming 1,2	

둘째, 전공기초과정의 편성에 관한 비교 분석은 Table 2.2와 같다.

수학 교과목으로서, H/W 교과과정에서는 공학수학1,2로 구성된 반면, S/W 교과과정에서는 이산수학, 확률 및 통계로 구성되어있다. 특히 수학을 필수도구로 활용하는 전공 교과목으로서, H/W 교과과정에서는 신호 및 시스템 등 5개 교과목들로 구성된 반면, S/W 교과과정에서는 해당 교과목이 없다. 이로써, H/W 교과과정에서는 수학을 체계적이고 집중적으로 교육하고 있다고 파악된다.

C++언어 프로그래밍 교과목으로서, H/W 프로그래밍 교과과정 편성은 2013년 이전에는 고급프로그래밍, 2013년 이후에는 고급프로그래밍1,2로 구성되어있으며, S/W 교과과정에서는 객체지향 프로그래밍1,2로 구성되어있다. 이로써, H/W 교과과정에서는 C++언어를 2013년 이후 체계적으로 교육하기 시작한 반면, S/W 교과과정에서는 C++언어를 지속적이고 체계적인 교육으로 진행해 왔다고 파악된다.

Table 2.2 The comparative analysis on major foundation curriculum of H/W and S/W

Courses	Departmental Courses of H/W		Departmental Courses of S/W
Mathematics Courses	Engineering Mathematics 1,2		Discrete Mathematics, Probability & Statistics
Major Courses	Electromagnetics, Electromagnetic Wave Theory, Electric & Electronic Laboratory, Signals & Systems, Circuit Theory		N/A
	C++ Programming Courses	Before 2013	Advanced Programming
After 2013		Advanced Programming 1,2	

셋째, 전공심화 및 응용과정의 편성에 관한 비교 분석은 Table 2.3과 같다.

수학 교과목으로서, H/W 교과과정에서는 확률 및 랜덤신호로 구성된 반면, S/W 교과과정에서는 해당 교과목이 없다. 특히 수학을 필수도구로 활용하는 전공 교과목으로서, H/W 교과과정에서는 제어공학 등 12개 교과목들로 구성된 반면, S/W 교과과정에서는 데이터베이스 등 6개 교과목들로 구성되어있다. 이로써, H/W 교과과정은 S/W 교과과정보다 더 많은 수학 관련 교과목들을 교수-학습한다고 판단된다.

C/C++언어 프로그래밍 활용 교과목으로서, H/W와 S/W 교과목들은 전공 교과내용과 관련된 기술을 구현하기 위해 C/C++언어 프로그래밍과 응용프로그램을 이용하여, 이론을 통해 추상화했던 전공 지식을 실습을 통해서 구체화하여 전공이론에 대한 이해도를 높인다.

특히 H/W 교과과정의 교과내용들은 C와 C를 기반으로 한 응용프로그램을 활용하고 있고 디지털 신호처리 및 실험, 디지털 영상처리 교과목에서만 Visual C++을 활용한다. 반면, S/W 교과과정의 교과내용들은 C, C++을 비롯한 다양한 객체지향언어 (Java, C#)와 웹프로그래밍 언어 (ASP, JSP, XML) 등을 활용하고 있다. 이로써 H/W 교과과정에서는 S/W 교과과정보다 C언어 활용도가 높은 반면, C++언어 활용도는 낮다고 파악된다. 즉, 전공 수업도구로서 C/C++언어 활용도의 차이점으로는 H/W 교과과정에서 C언어를 집중적으로 실습을 통해 활용하고 있는 반면, S/W 교과과정에서는 학과 특성상 다양한 프로그래밍 실습을 통하여 설계 및 구현토록 교수-학습함으로써 각 언어의 주요 특성을 파악토록 지도한다.

Table 2.3 The comparative analysis on major intensification and application curriculum of H/W and S/W

Courses	Departmental Courses of H/W	Departmental Courses of S/W
Mathematics Courses	Probability & Random Signal	N/A
Major Courses	Electronic Circuit 1,2, Electronic Circuit Experiments, Semiconductor Engineering, Communication Theory, Control Engineering, Digital Signal Processing & Experiment, Microcontroller, Data Communications, Radio Engineering, Digital Communication, Digital Image Processing	Database, Multimedia Systems, Algorithm Analysis & Application, Information Security, Computer Graphics, Digital Image Processing

넷째, 전공실무 및 신기술적용과정의 편성에 관한 비교 분석은 Table 2.4와 같다.

수학을 필수도구로 활용하는 전공 교과목으로서, H/W 교과과정에서는 지능시스템 등 9개 교과목들로 구성된 반면, S/W 교과과정에서는 인공지능 등 5개 교과목들로 구성되어 있다. 이로써, H/W 교과과정은 S/W 교과과정보다 더 많은 수학 관련 교과목들을 교수-학습한다고 판단된다. C/C++언어 프로그래밍 활용 교과목으로서, H/W와 S/W 교과내용들은 위의 전공심화 및 응용과정과 동일하다.

Table 2.4 The comparative analysis on major practice and new technology application curriculum of H/W and S/W

Courses	Departmental Courses of H/W	Departmental Courses of S/W
Major Courses	Integrated Circuit Design, Robotics, Radio Engineering Experiments, Communication Engineering Lab, Applied Signal Analysis Lab, Micro Robot Design & Production, Advanced Electronic Circuit, Next Generation Mobile Communications, Intelligent System	Artificial Intelligence, Internet Security, Datamining, Information Network Theory, Computer Security

위 대학교의 H/W와 S/W 교과과정의 편성에 관한 비교 분석 결과는 다음과 같다.

- 1) 수학 강좌 수는 H/W 교과과정의 강좌 수가 많다.
- 2) 수학을 필수도구로 활용하는 강좌 수는 H/W 교과과정의 강좌 수가 많다.
- 3) C언어 강좌 수는 동일하다.
- 4) C++언어 강좌 수는 2013년 이전인 경우 H/W 교과과정의 강좌 수가 적고, 2013년 이후인 경우 강좌 수는 동일하다.
- 5) C언어 활용 강좌 수는 H/W 교과과정의 강좌 수가 많다.
- 6) C++언어 활용 강좌 수는 H/W 교과과정의 강좌 수가 적다고 파악된다.

3. H/W와 S/W 교과과정에서 프로그래밍 학업성취도 실증 연구

3.1. 2010년과 2011년 프로그래밍 학업성취도 비교에 관한 연구내용

3.1.1. 연구 방법

본 절의 이 연구에서는 2010년과 2011년 2년간에 걸쳐 H/W와 S/W 분야의 졸업예정자들에게 C언어와 C++언어에 관한 프로그래밍 학업성취도를 측정하였다.

이를 위하여 H/W와 S/W 졸업예정자들 간에 각각의 전공 교과과정에 의한 4년 동안의 교육효과에 의하여 C언어와 C++언어에 관한 프로그래밍 학업성취도의 차이가 있는지를 분석하고, 차이가 있다면 그 이유를 전공 교과과정에서 파악하고, 어떠한 교육효과가 학업성취도에 영향을 주었는가를 분석하고자 한다. 이를 통하여 프로그래밍 학업성취도를 향상시켜줄 수 있는 효과적인 학습 환경을 밝히는 것이 목적이다.

이 연구를 위해 서울에 소재한 S대학교 이공대학 H/W 분야의 E학과와 S/W 분야의 S학과를 선정하여 졸업예정자들을 연구 대상 학생들로 선정하고 2010년 11월 26일, 2011년 11월 25일에 각각 학업성취도 측정을 실시하였다. 단, 학업성취도 평가는 프로그래밍 언어에 관한 기초적인 개념, 프로그래밍의 구문구조와 논리구조, 문제해결능력을 통한 프로그램 설계기법, 프로그래밍 언어를 이용하여 코드화하는 능력, 소프트웨어 개발 능력 등으로 측정하였고, 이 학업성취 수준을 근거로 심층적으로 진단하였다 (Choi와 Park, 2013).

3.1.2. 연구 결과

2010년과 2011년 2년간에 걸쳐 H/W 분야 E학과와 S/W 분야 S학과의 졸업예정자들의 프로그래밍 학업성취도에 대한 결과 분석 내용은 Table 3.1과 같다. C언어 프로그래밍 학업성취도는 H/W 졸업예정자들의 평균점수가 S/W 졸업예정자들의 평균점수보다 높은 반면, C++언어 프로그래밍 학업성취도는 H/W 졸업예정자의 평균점수가 S/W 졸업예정자들의 평균점수보다 낮음으로써, C언어와 C++언어에 대하여 두 집단 간에 통계적으로 유의한 차이를 보이고 있다.

따라서 이 연구에서는 4년간 교수-학습한 C언어 프로그래밍 교육방식에 관하여 H/W 교과과정이 S/W 교과과정보다 학습자의 C언어 프로그래밍 학업성취도 향상에 긍정적 영향을 주는 반면, C++언어 프로그래밍 교육방식에 관하여 S/W 교과과정이 H/W 교과과정보다 학습자의 C++언어 프로그래밍 학업성취도 향상에 긍정적 영향을 주었다라는 결론을 얻을 수 있었다. 단, 위의 T-검정결과는 모든 외생변수를 통제된 상태에서 평균값만 비교한 결과이다.

Table 3.1 Test results of 2010, 2011 on H/W and S/W group for programming academic achievement

Year	Grade	Test	Group	N	Mean	S.D.	t value	p-value
2010	Senior	C	H/W	32	80.718	13.438	3.168	0.002
			S/W	30	70.133	12.830		
		C++	H/W	32	72.593	15.751	-2.719	0.000
			S/W	30	84.000	17.276		
2011	Senior	C	H/W	36	69.986	15.062	3.104	0.002
			S/W	33	57.348	18.688		
		C++	H/W	36	48.694	20.538	-3.345	0.001
			S/W	33	64.781	18.917		

이 연구가 가지는 의의로서, H/W 졸업예정자들의 프로그래밍 수준을 S/W 졸업예정자들과의 상대적 비교 분석을 통하여 우수함을 파악하는 것이므로, C언어 프로그래밍 학업성취도는 H/W 졸업예정자

들이 S/W 졸업예정자들보다 높은 이유를 교과과정을 통해서 파악하여 그 요인들을 밝혀보고자 한다.

2장의 위 대학교의 H/W와 S/W 교과과정의 편성에 관한 비교 분석 결과에 의하면, 첫째, C언어 강좌 수는 H/W와 S/W 교과과정의 강좌 수가 동일하지만, C언어 활용 강좌 수는 H/W 교과과정의 강좌 수가 많다고 파악된다. 따라서 위 대학교의 H/W 교과과정에서는 전공을 강의한 후, 교과내용과 관련된 기술을 익히기 위하여 C언어 프로그래밍과 C언어를 기반으로 한 응용 소프트웨어를 이용해서 실습을 수행하고 기술을 구현함으로써 전공교과목의 이론에 대한 이해도를 높이는 교수-학습이 제공된다. 이러한 학습과정을 통하여 C언어의 직간접 학습기회가 S/W 교과과정보다 더 많이 제공된다고 파악된다.

특히 H/W 전공자는 교과과정에서 습득한 전공지식을 C언어를 활용하여 로봇, 제어시스템, 통신, 신호 등을 실전에서 물리적으로 직접 구현시키는 수업방식을 통하여 전공지식을 심도 있게 습득할 뿐만 아니라 C언어와 관련된 다양한 기술과 응용방법을 스스로 터득하여 능동적으로 전공에 적용할 수 있는 능력을 배양시킬 수 있다.

실제로 H/W 전공자는 스스로 작성한 C언어 프로그래밍을 시각적으로 구현시키는 프로그래밍 코딩 과정에서 수많은 시행착오를 거치면서 체계적인 접근에 의한 규칙을 찾아내고 코드분석 및 오류수정 등, 일련의 과정을 통하여 프로그래밍 개념들을 정확하고 깊이 있게 습득하여 C언어 프로그래밍 실력 향상을 도모할 수 있다고 사료된다. 이러한 학습과정을 통하여 H/W 전공자가 C언어로서 구현하는 기술이 더 우수하다고 판단된다.

둘째, C++언어 강좌 수는 2013년 이전인 경우, H/W 교과과정의 강좌 수가 S/W 교과과정의 강좌수보다 적고, 2013년 이후인 경우 강좌 수는 동일하다고 파악된다. 또한 C++언어 활용 강좌 수는 H/W 교과과정의 강좌 수가 적다고 파악된다. 따라서 H/W 교과과정에서는 C++언어를 활용하여 실습할 학습기회가 S/W 교과과정보다 적게 제공되므로 C++언어로서 구현하는 기술이 미흡하다고 판단된다.

이상에 기초하여, 프로그래밍 언어는 반복적인 실습을 많이 시행한 교과과정에서 학업성취도가 상대적으로 높은 것으로 파악되므로, 프로그래밍 원리도 중요하지만 우선적으로 얼마나 많은 반복적 실습을 시행하는지가 중요 요인이라고 판단된다.

3.2. 2010년과 2011년 수학 및 프로그래밍 학업성취도 간의 연관성에 관한 연구내용

3.2.1. 연구 방법

본 논문은 H/W 졸업예정자들의 프로그래밍 수준을 S/W 졸업예정자들과의 상대적 비교 분석을 통하여 파악하는 것이므로, 3.1절에서 C언어 프로그래밍 교육방식에 관하여 H/W 교과과정이 S/W 교과과정보다 학습자의 C언어 프로그래밍 학업성취도 향상에 긍정적 영향을 주었다라는 결론을 도출했으며, 그 이유를 교과과정을 통해서 파악하여 그 요인들을 분석했다.

즉, 2절의 위 대학교의 H/W와 S/W 교과과정의 편성에 관한 비교 분석 결과에 의하면, 수학 강좌 수와 수학을 필수도구로 활용하는 강좌 수는 H/W 교과과정의 강좌 수가 많다고 파악된다. 따라서 이러한 요인이 H/W 교과과정에서 C언어 프로그래밍 학업성취도에 어떠한 영향을 미치는지를 본 절에서 파악해보고자 한다.

단, H/W 교과과정에서 C++언어 활용 강좌 수가 극소수이며 3.1절에서 C++언어 프로그래밍 교육방식에 관하여 S/W 교과과정이 H/W 교과과정보다 학습자의 C++언어 프로그래밍 학업성취도 향상에 긍정적 영향을 주었다라는 결론을 도출했으므로 C++언어와 관련된 연구는 본 절에서 제외한다.

위 대학교 H/W 수학 교과과정 편성은 1학년 대학수학1,2와 2학년 공업수학1,2로 구성되었다. 2013년 이전 H/W C언어 프로그래밍 교과과정 편성은 1학년 1학기 C언어 프로그래밍 교과목이 개

설되지 않았으나, 1학년 2학기 C프로그래밍, 2학년 1학기 기초프로그래밍으로 구성되었다. 2013년 이후 1학년 C프로그래밍1,2로 신설하였다.

본 절의 이 연구에서는 2010년과 2011년 2년간에 걸쳐 H/W 교과과정에서 대학수학 및 공업수학의 학업성취도가 C언어 프로그래밍 교과목의 학업성취도에 미치는 영향력을 파악하는 것이 목적이다. 위의 실험연구에서 대학수학과 공업수학을 선택한 이유는 수학 관련 교과목으로서, 우선 이 교과목들은 수학 교과목에 포함되고, 특히 공학수학은 필수도구로 활용하는 전공교과목들의 선수 교과목이기 때문이다.

이를 위하여 첫째, H/W 1학년 교과과정의 1학기 강좌로 개설된 대학수학1 학업성취도가 최초로 2학기 강좌로 개설된 C프로그래밍 학업성취도에 한 학기 시차를 두고 미치는 영향을 조사하고 둘째, H/W 2학년 1학기 교과과정의 강좌로 개설된 공학수학1 학업성취도가 기초프로그래밍 학업성취도에 미치는 영향을 조사하여 회귀분석을 통해 분석해보고 두 교과목간의 관련성을 밝히고자 한다. 단, 연구대상 1학년 학생들은 프로그래밍의 선행학습이 없다는 가정 하에 실험연구를 실시하였다.

이 연구를 위해 서울에 소재한 S대학교 이공대학 H/W 분야의 E학과를 선정하여 2010년과 2011년 2년간에 걸쳐 1,2학년 학생들을 연구 대상 학생들로 전수 조사하였다.

3.2.2. 연구 결과

첫째, H/W 1학년 교과과정에서 1학기에 개설된 대학수학1 교과목의 학업성취도 (이하 Mathematics1로 지칭함)가 2학기에 개설된 C프로그래밍 교과목의 학업성취도 (이하 C programming으로 지칭함)에 어느 정도 설명력이 있는지를 알아보기 위해서 2010년과 2011년 2년에 걸쳐 대학수학1과 C프로그래밍의 학업성취도를 이용하여 회귀분석을 한 결과는 Table 3.2와 같다.

Table 3.2에서 2010년과 2011년 Mathematics1이 C programming에 통계적으로 유의하게 영향을 주는 것으로 나타났고 Mathematics1이 C programming을 각각 87.5%, 88.9% 정도 설명할 수 있음을 알 수 있다. 특히 Mathematics1과 C programming간의 관계를 설명할 수 있는 추정된 회귀모형은 다음과 같다.

$$2010\text{년 } C \text{ programming} \approx 0.952 \times \text{Mathematics1}$$

$$2011\text{년 } C \text{ programming} \approx 0.945 \times \text{Mathematics1}$$

이상에 기초하여, H/W 교과과정에서 개설된 C프로그래밍 교과목의 학업성취도에 통계적으로 유의한 영향을 미치는 요인으로는 대학수학1 교과목의 학업성취도임을 알 수 있으며 이러한 점을 토대로 볼 때, 2학기 C프로그래밍 교과목의 학업성취도 향상에는 1학기 대학수학1 교과목에서 습득한 수학적 학습능력이 주요한 요인이라고 판단된다.

Table 3.2 Regression analysis results of 2010,2011 on freshman academic achievement

Year	Grade	Model	Coefficient	S.E.	t value	p-value
2010	Fresh.	Mathematics1	0.952	0.040	23.413	0.000
			$R^2 = 0.875 \quad F = 548.212 \quad p = 0.000$			
2011	Fresh.	Mathematics1	0.945	0.037	25.440	0.000
			$R^2 = 0.889 \quad F = 647.199 \quad p = 0.000$			

둘째, H/W 2학년 1학기 교과과정에서 각각 개설된 공학수학1 교과목의 학업성취도 (이하 engineer.Math1으로 지칭함)가 기초프로그래밍 교과목의 학업성취도 (이하 elementary programming으로 지칭함)에 어느 정도 설명력이 있는지를 알아보기 위해서 2010년과 2011년 2년에 걸쳐 공학수학1과 기초프로그래밍의 학업성취도를 이용하여 회귀분석을 한 결과는 Table 3.3과 같다.

Table 3.3에서 2010년과 2011년 engineer.Math1이 elementary programming에 통계적으로 유의하게 영향을 주는 것으로 나타났고 engineer.Math1이 elementary programming을 각각 92.9%, 93.5% 정도 설명할 수 있음을 알 수 있다. 특히 engineer.Math1과 elementary programming간의 관계를 설명할 수 있는 추정된 회귀모형은 다음과 같다.

$$2010\text{년 elementary programming} \approx 0.921 \times \text{engineer.Math1}$$

$$2011\text{년 elementary programming} \approx 0.957 \times \text{engineer.Math1}$$

이상에 기초하여, H/W 교과과정에서 개설된 기초프로그래밍 교과목의 학업성취도에 통계적으로 유의한 영향을 미치는 요인으로는 공학수학1 교과목의 학업성취도임을 알 수 있으며 이러한 점을 토대로 볼 때, 기초프로그래밍 교과목의 학업성취도 향상에는 공학수학1 교과목에서 습득한 수학적 학습능력이 주요한 요인이라고 판단된다.

Table 3.3 Regression analysis results of 2010,2011 on sophomore academic achievement

Year	Grade	Model	Coefficient	S.E.	t value	p-value
2010	Soph.	engineer.Math1	0.921	0.027	33.090	0.000
			$R^2 = 0.929$ $F = 1094.986$ $p = 0.000$			
2011	Soph.	engineer.Math1	0.957	0.026	35.628	0.000
			$R^2 = 0.935$ $F = 1269.420$ $p = 0.000$			

위의 결과를 종합적으로 판단해 볼 때, 수학 관련 교과목들을 교수-학습함으로서 자연스럽게 문제해결력을 함양시킬 수 있으며, 이를 통하여 프로그래밍 학습능력의 신장을 도모해야 한다고 사료된다.

3.3. 2012년과 2013년 프로그래밍 학업성취도 비교에 관한 연구내용

3.3.1. 연구 방법

본 질의 이 연구에서는 2012년과 2013년 2년간에 걸쳐 H/W와 S/W 분야의 졸업예정자들에게 C언어와 C++언어에 관한 프로그래밍 학업성취도와 수학적 학습능력을 측정하여 그 차이가 있는가를 조사해보고, 수학적 학습능력이 C언어 프로그래밍 학업성취도에 미치는 영향력을 파악하는 것이 목적이다.

이를 위하여 첫째, H/W와 S/W 졸업예정자들 간에 각각의 전공 교과과정에 의한 4년 동안의 교육효과에 의하여 C언어와 C++언어에 관한 프로그래밍 학업성취도의 차이가 있는지를 분석하고, 차이가 있다면 그 이유를 전공 교과과정에서 파악하고, 어떠한 교육효과가 학업성취도에 영향을 주었는가를 분석하고자 한다. 이를 통하여 프로그래밍 학업성취도를 향상시켜줄 수 있는 효과적인 학습 환경을 밝히고자 한다.

특히 이 실험연구에서는 C언어 프로그래밍에 관한 학업성취도 검사를 구문 에러 (syntax error), 논리적인 에러 (logical error), 시간제한 (time limit) 등으로 세분화해서 측정하여 본 실험의 신뢰도를 향상시키고자 했다.

즉, 프로그래밍 과정은 문제의 이해, 논리의 설계, 프로그래밍의 코딩, 프로그래밍의 번역, 프로그램의 테스트, 프로그램의 활용 등으로 6개의 단계로 나눌 수 있다.

프로그래밍의 번역 단계에서 프로그래밍 언어의 명령문을 부정확하게 작성하면 번역 프로그램은 구문 에러 (syntax error)가 발생했음을 나타내는 에러 메시지를 보내므로 이에 관련된 사항을 측정하고자 Table 3.4에서 syntax error test 항목으로 실시하였다.

프로그래밍의 테스트 단계에서 프로그램에 구문 에러가 없다고 해서 논리적인 에러 (logical error)도 없는 것은 아니므로, 일단 프로그램에 구문 에러가 없으면 프로그래머는 논리적 에러가 존재하는지를 테

스트해야 하므로 이에 관련된 사항을 측정하고자 Table 3.4에서 logical error test 항목으로 실시하였다.

프로그램의 활용 단계에서 주어진 시간 안에 프로그램을 정확하게 수행 (run) 또는 실행 (execute) 시킬 수 있는 능력을 측정하기 위하여 이에 관련된 사항을 측정하고자 Table 3.4에서 time limit test 항목으로 실시하였다.

둘째, H/W와 S/W 졸업예정자들 간에 수학적 학습능력의 차이가 있는지를 분석하고, 차이가 있다면 그 이유를 전공 교과과정에서 파악하고 어떠한 교육효과가 수학적 학습능력에 영향을 주었는가를 분석하고자 한다. 단, 수학적 학습능력은 H/W와 S/W 교과과정에서 공통으로 개설된 대학수학1,2 교과내용의 성취도로써 측정하였다.

셋째, H/W와 S/W 졸업예정자들의 수학적 학습능력이 C언어 프로그래밍 학업성취도에 미치는 영향을 회귀분석을 통해 조사해보고 관련성을 파악하고자 한다.

이 연구를 위해 서울에 소재한 S대학교 이공대학 H/W 분야의 E학과와 S/W 분야의 S학과를 선정하여 졸업예정자들을 연구 대상 학생들로 선정하고 2012년 11월 30일, 2013년 11월 29일에 각각 학업성취도와 수학적 학습능력 측정을 실시하였다.

3.3.2. 연구 결과

2012년과 2013년 2년간에 걸쳐 H/W 분야 E학과와 S/W 분야 S학과의 졸업예정자들의 프로그래밍 학업성취도에 대한 결과 분석 내용은 Table 3.4와 같다.

첫째, C언어 프로그래밍 학업성취도 중에서 syntax error, logical error, time limit test 항목 모두 H/W 졸업예정자들의 평균점수가 S/W 졸업예정자들의 평균점수보다 높게 나온 반면, C++언어 프로그래밍 학업성취도는 H/W 졸업예정자들의 평균점수가 S/W 졸업예정자들의 평균점수보다 낮음으로써, C언어와 C++언어에 대하여 두 집단 간에 통계적으로 유의한 차이를 보이고 있다.

따라서 이 연구는 4년간 교수-학습한 C언어 프로그래밍 교육방식에 관하여 H/W 교과과정이 S/W 교과과정보다 학습자의 C언어 프로그래밍 학업성취도 향상에 긍정적 영향을 주는 반면, C++언어 프로그래밍 교육방식에 관하여 S/W 교과과정이 H/W 교과과정보다 학습자의 C++언어 프로그래밍 학업성취도 향상에 긍정적 영향을 주었다라는 결론을 얻을 수 있었다.

한편, Table 3.4에서 2013년 C++언어 프로그래밍 학업성취도 측정 비교 항목에서 *p*-value가 2010~2012년보다 다소 높게 0.039로 나온 이유로는 H/W 졸업예정자들과 S/W 졸업예정자들 간의 평균점수 차이가 2010~2012년보다 조금 줄어들었기 때문이다. 즉 2013년 이전 H/W C++언어 프로그래밍 교과과정에서는 2학년 2학기에 한 강좌 (고급프로그래밍)만 개설되었으나, 2013년 이후 H/W C++언어 프로그래밍 교과과정 개편으로 인하여 2학년 1,2학기에 두 강좌 (고급프로그래밍1,2)를 신설했고, H/W 졸업예정자들이 그 강좌들을 수강함으로써 성취도가 약간 향상되었다고 해석된다. 이로써, 2013년의 H/W C++프로그래밍 교과과정 개편이 H/W 전공자들에게 C++언어의 낮은 성취도를 개선하는데 기여했고 향후 C++언어 프로그래밍 학업성취도 수준의 질적 향상을 효과적으로 도모할 수 있으리라 예견된다.

둘째, 수학적 학습능력은 H/W 졸업예정자들의 평균점수가 S/W 졸업예정자들의 평균점수보다 높음으로써, 두 집단 간에는 통계적으로 유의한 차이를 보이고 있다. 따라서 H/W 교과과정은 S/W 교과과정보다 수학 관련 교과목이 많기 때문에 H/W 교과과정이 S/W 교과과정보다 수학적 학습능력 향상에 긍정적 영향을 주었다라는 결론을 얻을 수 있었다. 단, 위의 T-검정결과는 모든 외생변수를 통제한 상태에서 평균값만 비교한 결과이다.

Table 3.4 Test results of 2012, 2013 on H/W and S/W group for programming academic achievement

Year	Grade	Test	Group	N	Mean	S.D.	<i>t</i> value	<i>p</i> -value	
2012	Senior	Syntax error	H/W	58	71.603	17.921	2.965	0.003	
			S/W	36	60.750	16.093			
		C	Logical error	H/W	58	72.137	17.680	2.949	0.004
				S/W	36	60.527	19.883		
		Time limit	H/W	58	74.051	18.118	4.100	0.000	
			S/W	36	57.388	20.725			
C++	H/W	58	55.517	17.887	-3.280	0.001			
	S/W	36	68.166	18.621					
Math. learning ability	H/W	58	66.931	17.914	2.859	0.005			
	S/W	36	55.638	19.692					
2013	Senior	Syntax error	H/W	51	73.686	17.825	3.132	0.002	
			S/W	35	62.114	15.256			
		C	Logical error	H/W	51	74.117	15.837	3.095	0.002
				S/W	35	63.285	16.090		
		Time limit	H/W	51	75.490	17.273	5.420	0.000	
			S/W	35	55.714	15.614			
C++	H/W	51	58.921	17.243	-2.093	0.039			
	S/W	35	67.057	18.354					
Math. learning ability	H/W	51	69.313	17.137	3.804	0.000			
	S/W	35	55.457	15.756					

셋째, 수학적 학습능력 (이하 Math. learning ability로 지칭함)이 C언어 프로그래밍 학업성취도 (이하 C₄ programming으로 지칭함)에 어느 정도 설명력이 있는지를 알아보기 위해서 회귀분석을 한 결과는 Table 3.5와 같다.

Table 3.5에서 2012년과 2013년 H/W와 S/W 교과과정에서 Math. learning ability가 C₄ programming에 통계적으로 유의하게 영향을 주는 것으로 나타났고 Math. learning ability가 C₄ programming을 2012년과 2013년 H/W 교과과정에서 각각 90.8%, 91.7%, S/W 교과과정에서 각각 85.4%, 83.3% 정도 설명할 수 있음을 알 수 있다. 특히 Math. learning ability와 C₄ programming간의 관계를 설명할 수 있는 추정된 회귀모형은 다음과 같다.

$$2012\text{년 H/W 분야 } C_4 \text{ programming} \approx 2.853 \times \text{Math. learning ability}$$

$$2013\text{년 H/W 분야 } C_4 \text{ programming} \approx 2.838 \times \text{Math. learning ability}$$

$$2012\text{년 S/W 분야 } C_4 \text{ programming} \approx 2.612 \times \text{Math. learning ability}$$

$$2013\text{년 S/W 분야 } C_4 \text{ programming} \approx 2.699 \times \text{Math. learning ability}$$

이상에 기초하여, H/W와 S/W 교과과정에서 C언어 프로그래밍 학업성취도에 통계적으로 유의한 영향을 미치는 요인으로는 수학적 학습능력의 성취도임을 알 수 있으며 이러한 점을 토대로 볼 때, C언어 프로그래밍 학업성취도 향상에는 수학 관련 교과목들에서 습득한 수학적 학습능력이 중요한 요인이라고 판단된다.

Table 3.5 Regression analysis results of 2012, 2013 on senior academic achievement

Curriculum	Year	Grade	Model	Coefficient	S.E.	<i>t</i> value	<i>p</i> -value
H/W	2012	Senior	Math. learning ability	2.853	0.120	23.589	0.000
				$R^2 = 0.908 \quad F = 556.464 \quad p = 0.000$			
H/W	2013	Senior	Math. learning ability	2.838	0.121	23.309	0.000
				$R^2 = 0.917 \quad F = 543.312 \quad p = 0.000$			
S/W	2012	Senior	Math. learning ability	2.612	0.185	14.113	0.000
				$R^2 = 0.854 \quad F = 199.181 \quad p = 0.000$			
S/W	2013	Senior	Math. learning ability	2.699	0.210	12.854	0.000
				$R^2 = 0.833 \quad F = 165.244 \quad p = 0.000$			

위의 결과를 종합적으로 판단해 볼 때, 수학적 학습능력 우수자는 C언어 프로그래밍 학업성취도에서 높은 성취도를 유지할 가능성을 내포하고 있다고 파악되므로, C언어의 역량과 자질 함양을 위하여 수학적 학습능력의 신장을 도모해야 한다고 사료된다.

4. 결론

H/W 분야는 대한민국의 주요 경제 성장의 원동력으로서 수십 년간 타 분야에 비해 괄목할만한 성공을 이룩하였다. 그러나 21세기 정보화 시대에서는 H/W의 독자적인 발전만으로는 그 한계에 직면해 있는 상황이며, 이 H/W 분야의 발전 한계를 극복하기 위해서는 S/W의 기반성장으로 해결될 것으로 예상된다. H/W 및 S/W 분야에서 프로그래밍은 기본도구로 사용될 뿐만 아니라 필수불가결한 요소이다. 그러나 H/W 전공자들이 프로그래밍 언어를 기피하는 경향이 있다고 사료된다.

H/W 교과과정에서 첫째, 수학 강좌 수와 수학을 필수 도구로 활용하는 전공 강좌 수는 S/W 교과과정의 강좌 수보다 많으므로 수학 관련 교과목들의 교수-학습을 통하여 프로그래밍 학습에 기본인 논리적 사고력 개발과 문제해결력 배양을 자연스럽게 교육시킬 수 있다고 판단된다.

둘째, C언어 활용 강좌 수는 S/W 교과과정의 강좌 수보다 많으므로 반복적인 C언어 실습을 통하여 C언어 프로그래밍의 원리 이해와 구조화된 프로그램의 논리 개발을 효과적으로 학습시킬 수 있다고 판단된다.

이를 종합적으로 판단하면 H/W 교과과정은 수학 관련 교과목들을 통하여 논리적 사고력과 문제해결력을 함양하고, 전공 교과내용과 관련된 기술을 C언어를 통하여 구현함으로써, 추상화했던 전공지식을 구체화해서 전공이론에 대한 이해도를 향상시킨다고 파악된다. H/W 교과과정은 이러한 학습과정에서 반복적인 실습을 통하여 현대적인 응용들에 사용되는 다양한 프로그래밍 기술에 관한 지식들을 습득하도록 설계되어 있다고 사료된다. 더욱이 프로그래밍의 반복적인 훈련을 통하여 프로그래밍 언어에 대한 자신감 증대 효과를 얻을 수 있다고 예견된다.

대부분의 프로그래밍 언어들은 같은 구조를 사용하지만 표현하는 방식에서 차이가 있을 뿐이므로 프로그래밍 언어를 완전하게 습득하면 다른 프로그래밍 언어를 배우는데 시간과 노력을 절감할 수 있다. 현재 C언어는 프로그래밍 언어의 기본이므로, C언어 습득을 통하여 프로그래밍의 공통적인 원리를 터득함으로써 프로그래머로서의 능력 배양과 자질 함양을 도모할 수 있다.

본 논문에서는 H/W 교과과정을 통하여 습득한 C언어 프로그래밍 코딩 능력이 상대적 비교에 의해 우수함을 보였으므로, H/W 전공자들에게 프로그래밍에 대한 인식변화를 유도하여 자신감을 부여해주는 데 그 의의가 있다. 향후 H/W 교과과정에 학과차원의 특수성과 실정을 반영한 객체지향언어 활용 강좌를 추가 신설하고 교육 프로그램 다양화를 추진한다면, C++언어 프로그래밍 학업성취도 수준의 질적 향상을 도모할 수 있으며 이를 토대로 H/W와 S/W 분야가 융합하여 동반성장을 통한 고부가가치 제품을 생산할 것이라고 사료된다.

References

- Choi, H. S. and Park, C. (2013). A study on academic achievements of college students admitted by admissions officer selection: K university case. *Journal of the Korean Data & Information Science Society*, **24**, 1149-1157.
- Kim, J. H. (2003). Training practical IT professionals in higher education. *Communications of the Korean Information Science Society*, **21**, 8-12.
- Kim, M. (2002). Alternative instructional methods and strategies for effective computer programming education. *The Journal of Korean Association of Computer Education*, **5**, 1-9.
- Lee, S. W. (2008). A study on role of Mathematics/Statistics in IT fields, *Journal of the Korean Data & Information Science Society*, **19**, 1397-1408.
- Lee, S. W. (2012). An improvement for the employment rate of the S/W and H/W majors. *Journal of the Korean Data & Information Science Society*, **23**, 525-534.

A case study on programming academic achievement: Focused on the hardware curriculum

Seung-Woo Lee¹

¹Department of Electronic Engineering, Seokyeong University

Received 12 June 2014, revised 30 June 2014, accepted 7 July 2014

Abstract

The purpose of this study is to find the programming capability for the students majoring in H/W. For implementing this purpose, first, the academic achievements on the C language and C++ language are measured for the graduates-to-be majoring in H/W and S/W. Second, the H/W and S/W curriculum are compared and analyzed to derive the relevant factors to give influence on the academic achievement of the programming. Third, to find the influence of mathematic competence on the academic achievement of the programming, the relevance is analyzed in terms of the regression analyses between mathematics curriculum and programing curriculum. This paper presents the effective teaching method for the improvement of the programming academic achievement in the H/W curriculum.

Keywords: Academic achievement, C language, curriculum, programming.

¹ Professor, Department of Electronic Engineering, Seokyeong University, Seoul 136-704, Korea.
E-mail: swlee@skuniv.ac.kr