

기초교양필수 과목인 스크래치와 파이썬 프로그래밍 과목 수강생의 회복탄력성 분석

김세민¹ · 유강수² · 홍기천³ · 조영복^{4*}

The Analysis of Resilience of Programming Class' Students for Basic Liberal Arts

Semin Kim¹ · Kangsoo You² · Kicheon Hong³ · Youngbok Cho^{4*}

¹Adjunct professor, Department of Computer Education, Jeonju National University of Education, Jeonju 55101, Korea

²Associate Professor, Department of Library and Information Science, Jeonju University, Jeonju 55069, Korea

³Professor, Department of Computer Education, Jeonju National University of Education, Jeonju 55101, Korea

^{4*}Professor, Department of Information Security, Woosuk University, Daejeon 34520, Korea

요 약

최근 각 대학에서 소프트웨어교육을 강조하여 교양과목에 많이 편성하고 있다. 하지만 학생들은 프로그래밍 학습을 어려워하거나 프로그래밍 학습 동기가 부족하거나 배우려고 하는 의욕이 없는 경우가 많다. 본 연구에서 회복탄력성을 알아보고자 하는 이유는 프로그래밍 학습의 실패를 극복하여 포기하지 않고 다시 학습을 진행할 수 있는 회복하는 힘을 가질 수 있도록 지도하기 위함이다. 본 연구에서는 스크래치를 학습하는 학생 집단과 파이썬을 학습하는 학생 집단을 대상으로 회복탄력성 사전-사후 검사를 진행하였다. 연구결과로는 스크래치는 학생들이 다소 쉽게 받아들이고 열심히 하려는 모습을 보였지만, 파이썬은 상대적으로 스크래치보다 어려움이 있었음을 알 수 있었다. 본 연구를 통하여 프로그래밍 학습을 지속할 수 있는 요인을 파악하는데 도움이 될 것이라고 기대한다.

ABSTRACT

Recently, each university has been lecturing a lot on the liberal arts subject by emphasizing software education. However, students are often motivated by the difficulty of learning programming, the inability to recognize why they should learn programming, or even the fact that they do not try. The reason for the resilience is to guide programming learning to have the power to recover from the point of abandonment to proceed with the learning again. In this study, recovery elasticity pre-post-examination was conducted on the parts that learned scratches and those that learned Python. Studies have shown that while Scratch appears to be trying to accept and work harder, Python has been relatively more difficult than Scratch. It is expected that this study will help identify the factors that can sustain programming learning.

키워드 : 소프트웨어교육, 블록코딩도구, 프로그래밍언어, 회복탄력성, 컴퓨팅 사고

Key word : Software Education, Block Coding Tool, Programming Language, Resilience, Computational Thinking

Received 16 February 2019, Revised 27 February 2019, Accepted 15 March 2019

* Corresponding Author Youngbok Cho(E-mail:ybcho@dju.ac.kr, Tel:+82-42-280-2406)

Professor, Department of Information Security, Woosuk University, Daejeon 34520, Korea

Open Access <http://doi.org/10.6109/jkiice.2019.23.7.801>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서 론

4차 산업 혁명 시대의 일터에서는 기존의 농림수산업이나 제조업과는 다르게 초창기에 자본 및 설비 투자에 막대한 예산이 들지 않는다. 하지만 인문학적 상상력과 과학적 창의사고력을 가진 인재들이 필요하며, 컴퓨팅 사고를 통하여 문제 해결 능력을 가진 인재로 양성하는 것이 중요하다[1].

또한 전 세계적으로 소프트웨어교육의 중요성이 강조되면서 우리나라에서도 교육부와 미래창조과학부(2019년 시점은 과학기술정보통신부로 칭함)에서 2016년도에 ‘소프트웨어교육 활성화 기본 계획’을 심의하여 확정하였다. 그 영향으로 초·중등교육에서는 소프트웨어교육을 강화하고 있으며, 고등교육에서는 소프트웨어 중심대학 지원 사업을 통하여 전공자뿐만 아니라 비전공자에 이르기까지 소프트웨어 기초교육을 강조하는 노력을 기울이고 있다[2].

이러한 추세에 따라 각 대학에서는 소프트웨어 관련 과목들을 교양필수로 지정하는 경우가 많다. 그러나 모든 재학생들이 소프트웨어 관련 과목을 좋아하는 것은 아니어서 교양필수로 이수하여야 하는 이유를 모르겠다고 부정적인 반응을 보이면서 학습 초기부터 의욕을 잃는 경우가 많다. 게다가 프로그래밍을 실습하면서 문제가 해결되지 않을 때 수많은 실패를 겪으며 흥미를 잃는 경우도 있다[3][4].

프로그래밍과 같은 소프트웨어 관련 과목에서 학습자들이 흥미를 갖게 하려면 문제를 해결하는 과정에서 실패하였을 때의 과정들이 잘못된 것이 아니라는 인식을 심어주는 것이 필요하다. 학교 교육에서 행하였던 결과중심의 평가로 인하여 프로그래밍 학습 시 문제 해결에 실패할 경우 학습자들의 부담이 컸을 것이다. 또한 이를 대처하는 마음가짐에서 학습의 성패가 좌우된다고 할 수 있다[5].

본 연구에서는 프로그래밍 과목 학습자들이 흥미를 지속적으로 가질 수 있도록 구성하였고 문제해결에 실패하는 경우가 많은 소프트웨어 관련 과목의 특성에 맞게 학습에 영향을 줄 수 있는 요인으로 회복탄력성을 독립변인으로 설정하였다. 또한 프로그래밍 과목을 수강하는 학습자들을 대상으로 설문지를 통하여 회복탄력성의 변화를 분석하고, 학습자들이 어려워하는 점이 무엇인지 조사하였다.

본 논문의 구성은 다음과 같다. 서론에 이어 2장에서는 교양과목에서 프로그래밍 교육이 어떻게 이루어지고 있는지와 프로그래밍 학습을 하면서 회복탄력성을 측정할 기존 연구를 탐구하고, 3장에서는 연구대상과 연구 방법을 기술하며, 4장에서는 사전-사후 검사에 대한 연구 결과를 기술한 후, 마지막으로 결론 및 제언으로 글을 맺는다.

II. 관련 연구

2.1. 교양과목에서의 프로그래밍교육

컴퓨팅 사고가 강조됨에 따라 세계적으로 여러 나라에서 소프트웨어교육이 시도되고 있으며 우리나라에서도 소프트웨어 중심대학 사업을 계기로 하여, 비전공자들에게도 전공에 상관없이 교양필수 강좌로 교육하는 추세이다. 여러 선행 연구를 살펴보면 교양 과목으로 지정된 경우에는 쉽고 재미있는 내용으로 구성하였고, 스크래치나 앱인벤터와 같은 블록 프로그래밍 도구를 활용하거나 프로그래밍 언어를 가르치더라도 파이썬과 같은 쉬운 언어를 선택하는 경향이 많았다는 것을 알 수 있다[6-8].

대학의 교양 강좌에서는 소프트웨어 관련 과목이 교양 과목으로 지정된 것에 대하여 모든 학생들이 흥미를 가지고 있지 않으며, 특히 비전공자들은 거부감이 심한 것으로 나타나고 있다[4]. 그러나 막상 수업을 진행하면 학습자의 효능감과 만족도가 높아졌다는 연구 결과가 많이 나타나고 있다. 특히 개강 초기에 학습전이를 향상시키는 것이 소프트웨어 관련 과목에서의 성패를 좌우하는 것으로 나타나고 있다[7].

위와 같이 소프트웨어 관련 과목에서는 학습이 진행될수록 학습 성과가 개선되는 경우가 많다. 따라서 본 연구에서는 학습 초기인 개강 시기에서부터 연구대상을 분석하여 진행하였다.

2.2. 프로그래밍 학습과 회복탄력성

학습에서 무기력한 상황이 발생하는 요인으로는 지나친 실패 경험을 들 수 있다[9]. 특히 소프트웨어 관련 과목 중에서도 프로그래밍을 학습하는 과목에서는 단 한 번의 시도에서 문제가 해결되는 경우가 많지 않다. 첫 프로그래밍으로 학습이 성공되는 경우가 있을 수 있

지만 많은 경우에는 자신이 만든 소스코드에 오류가 생기고 이를 디버깅하는 과정을 수 없이 거칠 수도 있다. 이 과정에서 지나치게 실패하는 과정을 겪다보면 프로그래밍 학습에 흥미를 잃고 학습 무기력이 발생한다. 특히 프로그래밍 학습은 사소한 실수에도 프로그램이 정상적으로 동작하지 않는 경우가 많으므로 학습 성취도의 편차가 큰 편이다[5]. 이러한 현상으로 인하여 다음차시 학습에서 일부 부정적인 영향이 발생한다. 하지만 실패가 반드시 부정적인 결과만 있는 것은 아니다. 적당한 빈도의 실패는 개선하여야 할 점과 소중한 경험치가 누적되기 때문에 학습에 있어서 큰 도움이 되기도 한다.

실패를 하여도 그것을 거울삼아 다시 적응하는 상태로 되돌아오는 것, 즉 원래의 상태로 돌이키거나 원래의 상태를 되찾음을 ‘회복’이라고 한다. 한편, 실패를 거울삼아서 정신적인 저항력이 향상되고 이를 바탕으로 성장할 수 있는 힘을 ‘탄력’이라고 한다[5][10]. 프로그래밍 학습에서도 이 두 가지 의미를 합쳐서 학습의 실패를 딛고 학습에 흥미를 가질 수 있는 ‘회복탄력성’이라는 의미를 사용하여 본 연구의 독립변수로 사용하려 한다.

회복탄력성은 개인차가 존재한다. 프로그래밍 학습에 있어서 회복탄력성이 높은 학습자는 디버깅 과정을 많이 수행한다 하여도 학습에 흥미나 만족도가 낮아지는 정도가 작으며 결국은 성공적인 소스코드를 작성할 수 있다. 그러나 학습자가 가지고 있는 회복탄력성이 낮으면 프로그래밍 학습에 실패할 수 있다. 회복탄력성에는 자기제어, 사회성, 긍정적인 마음 등의 하위요소가 존재하며, 각각의 하위요소들도 또 다른 하위요소가 존재한다. 그림 1은 회복탄력성의 하위구성요소를 간략하게 표현한 것이다[11].

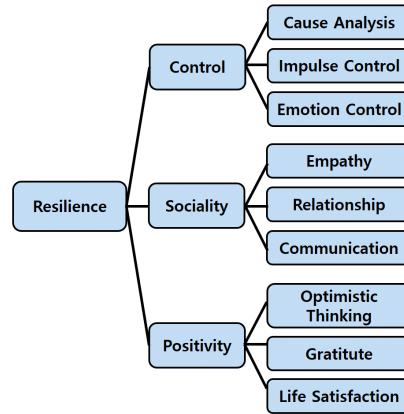


Fig. 1 Components of Resilience

3.2. 연구 방법 및 절차

본 연구는 대학의 교양 프로그래밍 강좌 중에서 스크래치를 학습하는 집단과 파이썬을 학습하는 집단으로 선정하여 진행하였다. 이 두 과목은 A 대학에서 교양필수 과목으로 편성되어 있다. 스크래치는 대표적인 블록 코딩 프로그래밍 도구이며, 파이썬은 최근 각 대학에서 많이 선택하는 프로그래밍 언어이다. 개강 첫 주에 각 집단에 대하여 회복탄력성 사전 검사를 진행하고, 마지막 주에 사후 검사를 진행하였다.

본 연구에서는 학습자 위주의 회복탄력성을 조사하는 데 목적을 두고 있어 교수자가 학습자에게 미치는 영향을 배제하고 연구를 진행하였다. 본 연구는 회복탄력성 사전 검사에 대한 차별과 편견을 배제하여야 하고, 학습자 스스로 검사 결과를 의식하여 학습에 참여하는 태도를 지양하여야 한다. 따라서 개강 첫 주에 학생들의 양해를 구해 사전 검사의 결과를 통보하지 않고, 사후 검사 후에 모두 통보하였다. 본 연구의 구체적인 실험 설계는 표 1과 같다.

III. 연구 대상 및 방법

3.1. 연구 대상

본 연구는 전라북도 A대학의 교양필수과목 중 프로그래밍 강좌의 수강생을 대상으로 진행하였다. 연구 대상은 총 399명이며, 과목별로 스크래치를 학습하는 집단의 전체 인원은 6개 분반으로 183명이고, 파이썬을 학습하는 집단의 전체 인원은 7개 분반으로 216명이다. 각 분반은 최대 35명 정원이며 대개 30명이나 31명 정도로 구성되어 있다.

Table. 1 Experimental Design

G ₁	O ₁	O ₂
G ₂	O ₃	O ₄
G ₁ : Comparison Group 1(Scratch Classes) G ₂ : Comparison Group 2(Python Classes) O ₁ , O ₃ : Pre-Test for Resilience O ₂ , O ₄ : Post-Test for Resilience		

3.3. 실험 도구

본 연구에 사용할 검사 도구는 기존 연구 중 학업적 자기 효능감과 회복탄력성을 연구하였던 논문에 제시된 회복탄력성 설문지를 이용하였다[3, 12]. 제시된 회복탄력성 설문지는 표 2의 전문가들과 함께 컴퓨터 프로그래밍 수업에 알맞게 수정하고, 역채점 문항으로 설정된 부분은 학생들의 혼선을 막기 위하여 다시 수정하여 배포하였다. 전문가들은 컴퓨터공학을 전공한 교수진, 컴퓨터교육을 전공한 교사, 교육심리학을 전공한 종교인 등으로 구성하였다. 실험 도구의 문항별 하위요소는 표 3과 같다.

Table. 2 Special Notes of Expert

Expert	Job	Special Notes
A	Teacher	Computer Education, Ph. D.
B	Protestant Minister	Theology, Th. M. Educational Psychology, Th. M.
C	Professor	Computer Engineering, Ph. D.
D	Professor	Computer Engineering, Ph. D.
E	Professor	Computer Science, Ph. D.

Table. 3 Subelement of Learning Demotivation

Classification	Factor of Resilience	No.
Control	Cause Analysis	1-3
	Emotion Control	4-6
	Impulse Control	7-9
Postivity	Gratitude	10-12
	Life Satisfaction	13-15
	Optimistic Thinking	16-18
Sociality	Relationship	19-21
	Communication	22-24
	Empathy	25-27

IV. 연구결과 및 분석

본 연구 대상은 총 399명으로 스크래치를 학습한 집단은 183명의 학생이고, 파이썬을 학습한 집단은 216명의 학생이다. 연구 대상은 각 집단별로 30명을 초과하므로 모수통계가 가능하였다. 따라서 본 연구에서는 t-검정을 이용하여 사전-사후 검사를 실시하였다.

4.1. 스크래치 집단 사전-사후 검사

스크래치를 학습하는 집단에 대하여 회복탄력성의 각 하위요소별로 사전-사후 검사를 실시하였다. 모든 하위요소에 대하여 유의미하게 회복탄력성의 개선이 이루어졌다. 특히 ‘원인분석’항목의 하위요소에서는 다른 하위요소에 비하여 스크래치 집단의 학생들 간 편차가 감소하였다. 스크래치를 학습하는 집단에 대한 사전-사후 검사 결과는 표 4와 같다.

Table. 4 Scratch Programming Studies (p>.05)

Subscales	Pre/Post	N	Avg.	SD	t	p
Cause Analysis	Pre	183	11.48	1.812	-2.401	.000
	Post		11.81	.601		
Emotion Control	Pre	183	9.28	2.308	-4.788	.007
	Post		10.34	1.909		
Impulse Control	Pre	183	10.17	1.990	-.950	.000
	Post		10.35	1.497		
Gratitude	Pre	183	8.91	2.232	-6.647	.002
	Post		10.35	1.797		
Life Satisfaction	Pre	183	10.89	1.696	-4.902	.006
	Post		11.66	1.256		
Optimistic Thinking	Pre	183	7.70	2.216	-8.698	.044
	Post		9.63	2.009		
Relation ship	Pre	183	10.50	2.120	-.469	.000
	Post		10.59	1.638		
Communi cation	Pre	183	8.68	2.353	-6.063	.001
	Post		10.00	1.760		
Empathy	Pre	183	10.90	1.813	1.296	.019
	Post		10.68	1.386		

4.2. 파이썬 집단 사전-사후 검사

파이썬을 학습하는 집단에 대하여 회복탄력성의 각 하위요소별로 사전-사후 검사를 실시하였다. 회복탄력성의 하위요소 중 ‘감정통제력’, ‘충동통제력’, ‘감사하기’, ‘낙관적인 생각’, ‘커뮤니케이션 능력’, ‘공감능력’의 하위요소 대하여 유의미하게 회복탄력성의 개선이 이루어졌다. 그러나 ‘원인분석력’, ‘생활만족도’, ‘관계성’의 하위요소에서는 유의미하게 회복탄력성이 감소하였다. 파이썬을 학습하는 집단에 대한 사전-사후 검사 결과는 표 5와 같다.

Table. 5 Python Programming Studies ($p>.05$)

Subscales	Pre/Post	N	Avg.	SD	t	p
Cause Analysis	Pre	216	11.25	1.712	3.316	.004
	Post		10.75	1.374		
Emotion Control	Pre	216	9.49	2.055	-.498	.044
	Post		9.58	1.797		
Impulse Control	Pre	216	10.24	1.464	-1.869	.009
	Post		10.47	1.141		
Gratitude	Pre	216	9.01	2.274	-1.827	.012
	Post		9.37	1.747		
Life Satisfaction	Pre	216	10.77	1.878	2.437	.018
	Post		10.37	1.540		
Optimistic Thinking	Pre	216	8.16	1.903	-3.015	.040
	Post		8.66	1.523		
Relationship	Pre	216	10.43	2.127	.976	.011
	Post		10.24	1.801		
Communication	Pre	216	9.74	1.809	-2.959	.003
	Post		10.22	1.529		
Empathy	Pre	216	10.97	1.438	-1.765	.016
	Post		11.21	1.397		

4.3. 논의

위의 연구결과를 통하여 스크래치를 학습하는 집단과 파이썬을 학습하는 집단의 회복탄력성에 대한 차이를 알 수 있었다. 스크래치는 다소 쉽게 다가갈 수 있는 블록 코딩 도구이므로, 학습자들이 회복탄력성의 모든 하위요소에서 개선된 것으로 보인다. 또한 실제 학습 시에 관찰한 기록으로도 프로그래밍 과정이 어렵고 힘들어서 포기할 것 같았지만, 이를 극복하고 다시 시도할 수 있는 힘을 가지고 학습에 임하는 모습이 발견되었다. 파이썬은 C언어나 Java보다는 쉽지만 교양필수 강좌로는 다소 어려운 편이다. 따라서 파이썬의 학습이 부진한 학생들은 원인분석력에 의한 문제해결에 어려움을 겪거나, 파이썬 강좌에 만족을 느끼지 못하는 등의 어려움을 겪고 있었다.

표 4와 표 5의 데이터를 비교하면 스크래치를 학습하는 집단이 파이썬을 학습하는 집단보다 회복탄력성 수치가 더 높게 나타난 것을 알 수 있다. 이 역시 스크래치가 파이썬보다 상대적으로 더 쉽기에 학생들이 느끼는 회복탄력성에서 차이가 있음을 알 수 있었다.

V. 결론 및 제언

본 연구에서는 전북의 A대학에서 교양필수 과목으로 설강된 스크래치 블록 코딩을 학습하는 집단과 파이썬 프로그래밍 언어를 학습하는 집단을 대상으로 회복탄력성의 변화를 분석하였다. 두 집단의 회복탄력성 사전 검사와 사후 검사를 통하여 대학에서 프로그래밍 관련 교양필수 과목에서 어떤 점을 고려하여 학생들의 학습 효율 및 만족도를 높일 수 있는지 분석할 수 있으며 다음과 같은 점을 시사할 수 있었다.

첫째, 스크래치를 학습하는 집단에서는 학습자들이 학기 초에는 낯설어 하였지만 재미있는 주제로 프로그래밍 학습을 반복한 결과, 모든 하위요소에서 회복탄력성이 개선되었다.

둘째, 파이썬을 학습하는 집단에서는 개선되지 못하고 오히려 퇴보된 하위요소가 있었다. 그러나 개선되지 못한 하위요소보다 개선된 하위요소가 더 많으므로 회복탄력성이 개선되었다고 할 수 있다.

셋째, 파이썬과 스크래치를 학습하는 집단의 데이터를 비교해 보면 파이썬을 스크래치보다 더 어려워하는 모습을 보였다. 이는 키보드 입력 방식의 학습에 비해 블록 코딩 방식의 학습이 시각적으로도 쉽게 접근할 수 있기 때문이다.

본 연구의 한계점으로는 해당 학교에서 개설된 과목만을 중심으로 하다 보니 블록 코딩 도구는 스크래치로 한정하였고, 프로그래밍 언어는 파이썬으로 한정지를 수밖에 없었다. 프로그래밍 도구 및 언어들은 상당히 다양하다. 따라서 향후에는 인근 학교의 여러 과목을 대상으로 연구를 진행하여야 할 필요가 있고 또한 학습자의 특성을 전공으로도 분류하여 연구를 진행할 필요가 있다고 본다.

REFERENCES

[1] Software based Society [Internet]. Available: <http://www.software.kr>

[2] Preparation for Elementary and Middle School SW Education Essential and School-Oriented Promotion of SW Education, Ministry of Education and Ministry of Future Creation Science, 2016.

[3] J. S. Lee, "The Effect of Academic Self-Efficacy and

- Resilience on Self-Directed Learning,” M.S. Thesis, The Graduate School Sunchon National University, Suncheon, Jeonam, 2016.
- [4] M. J. Oh, “Non-Major Students’ Perceptions of Programming Education using the Scratch Programming Language,” *The Journal of Korean Association of Computer Education*, vol. 20, no. 1, pp. 1-11, Jan. 2017.
- [5] S. M. Kim, and K. Y. You, “The Effect of Introverted or Extroverted Personality Type on the Resilience of Java Programming Learning : Focused on Students at Technical Specialized High School,” *Journal of the Korean Association of Information Education*, vol. 22, no. 4, pp. 439-446, Aug. 2018.
- [6] J. K. Kim, “Computational Thinking and Programming as a Liberal Arts Education for University,” *Journal of Korean Multimedia*, vol. 22, no. 1, pp. 20-25, Mar. 2018.
- [7] S. H. Kim, “Analysis about the Initial Process of Learning Transfer in Computational Thinking Education,” *The Journal of Korean Association of Computer Education*, vol. 20, no. 6, pp. 61-69, Nov. 2017.
- [8] S. Y. Pi, “A Study on Coding Education of Non-Computer Majors for IT Convergence Education,” *Journal of Digital Convergence*, vol. 14, no. 10, pp. 1-8, Oct. 2016.
- [9] S. O. Lee, Middle School Student’s of Positive Psychological Capital, Stress Coping Strategies and Learned Helplessness, M.S. Thesis, The Graduate School of Hongik University, Seoul, 2016.
- [10] J. H. Kim, *Resilience*, Goyang, Gyeonggi: Wisdom House, 2011.
- [11] W. Y. Shin, M. A. Choi, and J. H. Kim, “The Effects of the Three Resilience Factors on Problematic Online Game Uses,” *Journal of Cybercommunication Academic Society*, vol. 26, no. 3, pp. 43-81, Sep. 2009.
- [12] M. H. Ahn, A Study on Effective Music Education according to Personality Type : Focused on Theory and Praxis of MBTI, M.S. Thesis, The Graduate School Catholic Kwandong University, Gangneung, Gangwon, 2018.



김세민(Semin Kim)

2010: 공주대학교 컴퓨터교육학과 교육학박사수료
 2018: 한밭대학교 정보통신공학과 공학박사
 현재: 전주교육대학교 컴퓨터교육과 겸임교수
 ※관심분야: 소프트웨어교육, 심리정보과학, 스포츠정보과학, 로봇활용 및 메이커교육



유강수(Kangsoo You)

2005: 전북대학교 영상공학과 공학박사
 2006~2015: 전주대학교 기초융합교육원 교수
 2015~현재: 전주대학교 문헌정보학과 교수
 ※관심분야: 소프트웨어교육, 심리정보과학, 영상처리, 컴퓨터비전



홍기천(Kicheon Hong)

2000: 전북대학교 대학원 이학박사
 2015: Tufts University 방문교수
 현재: 전주교육대학교 컴퓨터교육과 교수
 ※관심분야: 소프트웨어교육, 로봇활용교육, STEAM교육



조영복(Youngbok Cho)

2005: 충북대학교 전자계산학과 공학석사
 2012: 충북대학교 전자계산학과 공학박사
 2016: 충북대학교 의학과 의학박사수료
 2012~2018: 충북대학교 소프트웨어학과 초빙교수
 현재: 대전대학교 정보보안학과 조교수
 ※관심분야: 소프트웨어교육, 의료영상처리, 정보보안, 의료정보보호, 모바일보안