

SW전공자 프로그래밍 입문 수업의 스크래치 활용 수업 모형 연구*

고 광 일*

요 약

프로그래밍 언어는 4차 산업혁명시대에서 그 중요성이 증대하고 있는 소프트웨어의 핵심 교육 영역이지만 수학적 지식과 논리적 사고력을 요구하고 있어 기초 학력이 낮은 많은 수의 지방 사립대나 전문대 학생들 - 심지어 SW전공 학생들에게도 매우 어려운 과목으로 인식되고 있다. 이런 문제로 인해 SW전공 학생이 프로그래밍 언어 입문 수업 도중 전공에 대한 흥미와 자신감을 잃고 전공을 변경하거나 학업 자체를 포기하는 상황이 발생하기도 한다. 이에 본 연구는 대표적인 프로그래밍 입문 언어인 C언어 교육에 스크래치를 활용하는 수업 모형을 설계하였다. 이를 위해, C언어가 지원하는 프로그래밍 개념들 중 스크래치로 교육 가능한 개념들을 분석하고 스크래치 실습 예제들을 개발하였다. 또한, 프로그래밍 개념에 대해서 먼저 스크래치의 구현 방식 교육과 실습 예제들을 통해 명확하게 이해하고 C언어를 교육하는 수업 모형을 설계하였고, 모 지방 사립대의 SW전공 신입생들을 대상으로 실험을 진행하여 본 수업 모형의 실효성을 검증하였다. 프로그래밍 언어 교육이 보안 관련 IT 전공자들에게도 필수적으로 요구되는 상황에서 본 연구가 그들의 프로그래밍 언어 입문 교육에 도움이 되기를 기대한다.

A Study on the Instructional Model utilizing Scratch for Introductory Programming Classes of SW-Major Students

Kwangil KO*

ABSTRACT

The programming language is a core education area of software that is becoming increasingly important in the age of the fourth industrial revolution, but it requires mathematical knowledge and logical thinking skills, so that many local private university and college students with low basic skills are having difficulties learning it. This problem occasionally causes SW-major students to lose interest and confidence in their majors during the introductory course of programming languages; making them change their majors, or give up their studies. In this study, we designed an instructional model using Scratch for educating C-language which is a typical programming introductory language. To do this, we analyzed the concepts that can be trained by Scratch among the programming concepts supported by C-language, and developed the examples of Scratch for exercising the concepts. In addition, we designed an instructional model, by which the programming concepts are first learned through Scratch and then C-language is taught, and conducted an experiment on the SW-major freshman students of a local private university to verify the effectiveness of the model. In the situation where SW education is becoming common, we expect that this study will help programming language education of security IT students.

Key words : EPL, Instructional model, C Programming language, Scratch, SW-major

접수일(2018년 5월 19일), 수정일(1차: 2018년 6월 20일),
게재확정일(2018년 6월 29일)

* 우송대학교 테크노미디어융합학부 영상콘텐츠전공

★ 본 연구는 2018년도 「우송대학교 교내 학술연구조성비」
지원에 의해 이루어진 것임.

1. 서 론

최근 4차 산업혁명시대에 대한 관심이 높아지면서 교육계를 포함해 사회 전반적으로 소프트웨어(SW: Software)에 대한 관심이 증대되고 있다. 소프트웨어의 중요성을 인정한 정부도 'SW중심사회'라는 기치아래 2015년도에 정보교과의 교육과정 개정을 통해 2018년도부터 초·중등 교과 과정에 SW교육을 의무화시켰으며, 정부 주도 하에 2014년도부터 SW교육선도 및 연구학교를 선정하여 2018년 현재 초등학교 940개, 중학교 456개, 고등학교 245개가 운영되고 있다. 정부의 이런 SW교육 강화 정책은 대학교육에도 확산되어 2015년도부터 2017년도까지 20개의 대학들을 'SW중심대학'으로 선정하고 대학교육을 SW중심으로 혁신하여 SW가치를 전파하고 있다[1].

SW교육의 핵심인 프로그래밍 교육은 프로그래밍 언어의 문법과 알고리즘의 구현, 그리고 오류 수정 등의 어려움으로 인해 프로그래밍 교육을 처음 접하는 초·중·고등학생뿐만 아니라 SW분야를 전공으로 선택한 대학생들에게도 어려운 과정이다. 비록, 아직 대학 입학 전의 학생들에게는 스크래치[2]와 엔트리[3]와 같은 그래픽 기반의 교육용 프로그래밍 언어(EPL: Educational Programming Language)를 활용하여 이런 문제점을 완화하고 있지만 텍스트 기반의 일반 프로그래밍 언어(GPL: General Purpose Programming Language)를 학습해야 하는 SW전공 대학생들 중에는 결국 어려움을 극복하지 못하고 전공을 변경하거나 포기하는 경우가 발생하고 있다. 이런 현상은 특히 기초 학력이 부족한 지방 사립대와 전문대 학생들에게 보다 심각하며 대학 진학 자원의 감소와 더불어 심화되고 있다.

스크래치와 엔트리 같은 EPL은 자료, 변수, 문장, 연산 등과 같은 프로그램 구성 요소들을 이미 지화된 개체들로 제공하고 이들을 블록 조립하듯이 프로그래밍하는 방법을 지원한다. 이런 특징 때문에 프로그래밍에 생소한 학생들의 프로그래밍 교육 언어로 널리 사용되고 있으며 그 효용성을 기반으로 한 연구들이 진행되어 왔다. 고등학생

이전의 학생들을 대상으로 EPL을 이용한 프로그래밍 수업 효과[4,5,6]와 STEAM기반 교육프로그램에 대한 연구[7] 등이 수행되었는데 이 연구들은 EPL을 활용한 프로그래밍 교육이 학생들의 사고력, 문제해결력, 그리고 흥미 및 성취도 향상에 도움을 준다는 것을 실험적으로 입증하였다. SW비전공자의 SW교육을 위한 연구도 진행되었는데 EPL을 활용한 SW비전공자들의 SW교육에 대한 인식과 IT융합교육을 위한 SW비전공자 코딩교육의 발전방안 등이 연구되었고[8,9], EPL의 시각적 표현능력을 활용한 SW교육 콘텐츠 개발 모형도 설계되었다[10].

본 연구는 위와 같은 EPL의 효용성을 기반으로 SW전공 학생들의 프로그래밍 언어 입문 수업 모형을 설계하고 수업 모형의 효과를 입증하기 위한 실험을 수행하였다. 구체적으로, 대표적인 프로그래밍 입문 언어인 C언어 수업에 스크래치를 활용하는 수업 모형을 설계하였는데, 스크래치의 활용 효과를 높이고자 스크래치로 교육 가능한 C언어의 프로그래밍 개념들을 분석하여 스크래치 활용 범위를 명확히 하고, 그 개념들의 이해와 활용 능력 습득을 위한 실습 예제들을 개발하여 수업 모형에 포함하였다. 또한, 수업 모형의 효용성을 검증하기 위해 모 지방사립대의 SW전공 신입생들을 대상으로 본 수업 모형을 적용한 반과 C언어만을 교육하는 반으로 나누어 학습 효과를 분석하였다. 프로그래밍 언어는 보안 IT 분야 학생들에게도 필수적인 교육과정이기 때문에 본 연구가 그들의 프로그래밍 언어 입문 교육에 도움이 될 것 기대한다.

2. 연구 배경

2.1 EPL 활용 연구

EPL 활용 연구는 주로 고등학생 이전의 학생들 또는 SW비전공자 대상으로 EPL 교육이 그들의 사고력 향상과 SW에 대한 흥미에 미치는 영향과 EPL을 활용한 프로그래밍 언어 교육과정 개발에 대한 연구가 활발히 진행되었다.

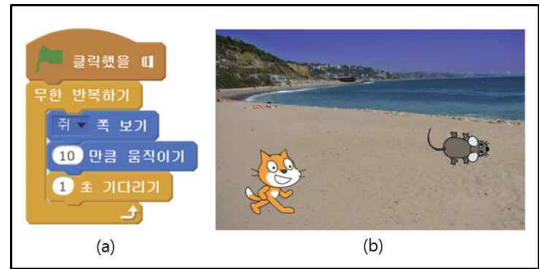
송정범 외(2008)[4]와 배학진 외(2009)[5]는 초등학생을 대상으로 EPL 교육이 학생들의 문제 해결력과 논리적 사고력 향상에 긍정적 영향을 주었음을 밝혔고, 이은경(2009)[6]과 조성환 외(2008)[11]는 중등학생을 대상으로 스크래치를 이용한 프로그래밍 수업이 학생들의 사고력 향상에 효과적임을 밝혔다. 또한, 노희진 외(2015)[12]는 고등학생 과학 학습에 스크래치를 활용한 사례를 통해 학생들의 과학 수업에 대한 흥미와 수업의 몰입도가 높아졌다는 결과를 얻었다. 이와 같은 연구 결과는 EPL를 활용한 교육과정 개발을 촉발하게 되었는데 오정철 외(2012)[13]는 스크래치를 이용한 STEAM기반 교육프로그램을 개발하였고, 안형진 외(2013)[14]는 문제해결 중심으로 초등학교 스크래치 교육과정을 개발하였다.

SW전공자를 대상으로 한 EPL 활용 연구는 상대적으로 매우 미흡한 실정인데 박정신 외(2012)[15]는 C언어 수업에 어려움을 겪고 있는 전문대 SW전공 신입생들을 대상으로 스크래치를 활용할 때 성적과 만족도 향상에 효과가 있음을 밝혔다. 이 연구는 SW전공자를 대상으로 EPL의 효과를 밝혔다는 점에서 의미가 있지만 C언어 수업 모형에 구체적인 스크래치 활용 범위와 방안이 제시되지 못한 한계가 있다.

2.2 스크래치 소개

스크래치는 2007년 MIT 미디어랩의 ‘라이프롱 킨더가르텐’ (Lifelong Kindergarten) 그룹[16]에서 개발한 프로그래밍 교육용 언어로서 21세기에 필요한 핵심 능력인 ‘창의적 사고’, ‘논리적 추론’, 그리고 ‘협동심’ 교육을 목표로 한다. 스크래치는 프로그래밍에 필요한 다양한 기능들과 자료들을 사전에 정의하여 블록형태의 이미지로 제공함으로써 일반 프로그래밍 언어가 갖는 문법적 어려움을 최소화하고 마치 블록 조립하듯이 프로그래밍하기 때문에 누구나 쉽게 프로그래밍에 친숙해 질 수 있다는 특징이 있다 ((그림 1-(a)) 참고).

스크래치는 고양이, 쥐와 같은 다양한 객체들을 이미지로 구현한 370개의 스프라이트 (Sprite)들



(그림 1) (a) 스크래치 프로그래밍의 예, (b) 스크래치 스프라이트와 무대 배경의 예

과 무대 (Stage)를 위한 운동장, 해변과 같은 103개의 배경을 제공함으로써 게임, 인터랙티브 스토리, 애니메이션 등과 같은 창의적 콘텐츠 제작에 용이하다 ((그림 1-(b)) 참고). 또한, 239개의 음원 파일들을 제공하고 39개의 악기들을 음높이와 박자에 따라 연주할 수 있는 기능을 제공하여 음악 콘텐츠 제작도 지원한다.

3. 스크래치 활용 수업 모형

3.1 스크래치 분석 및 예제 설계

본 연구는 SW전공 신입생 대상 C언어 수업의 스크래치 활용 수업 모형 개발을 위해 스크래치를 활용하여 교육이 가능한 C언어의 주요 프로그래밍 개념들과 이들을 스크래치가 구현한 방식을 분석하였다. (<표 1> 참고)

<표 1> 스크래치로 교육 가능한 C언어의 주요 개념들과 이들의 스크래치 구현 방식

개념 구분	스크래치의 구현 방식
변수와 자료형	기본 자료형 <ul style="list-style-type: none"> • 수, 불린, 문자열 사용 • 정수와 실수를 구분하지 않음 • 문자와 문자열을 구분하지 않음 • 블록의 타원형 입력 창은 수만 입력, 사각형 입력창은 문자열 입력 가능
	변수 <ul style="list-style-type: none"> • 변수 생성 시 변수 이름만 설정 • 변수의 값에 따라 수 또는 문자열로 자동 인식
	변수 범위 <ul style="list-style-type: none"> • 한 스프라이트만 사용 가능 변수 생성 • 모든 스프라이트 사용 가능 변수 생성

입력과 출력	문자열 입출력	<ul style="list-style-type: none"> 키보드로 문자열 입력 받는 블록 제공 입력 값을 특정 변수에 자동 저장 스프라이트의 말풍선으로 문자열 출력
연산	대입	• 변수의 대입연산 블록 자동 생성
	산술	• 사칙연산을 수행하는 블록 제공
	관계	• <, >, =의 관계연산 수행 블록 제공
	논리	• ^, V, ~의 논리연산 수행 블록 제공
문장 제어	순차	• 상하로 결합된 블록들의 순차적 실행
	선택	<ul style="list-style-type: none"> if 선택을 지원하는 블록 제공 if else 선택을 지원하는 블록 제공
	반복	<ul style="list-style-type: none"> 무한 반복을 지원하는 블록 제공 특정 횟수만큼 반복하는 블록 제공 Until 반복 (조건이 거짓인 동안 반복) 블록 제공
배열	1차원 배열	<ul style="list-style-type: none"> 1차원 배열의 리스트 자료형 제공 리스트의 연산 블록들 자동 생성
함수	실행	• C언어와 동일한 함수 실행 순서 지원
	반환값	• 반환값 기능 지원하지 않음
	인자	• 값호출 방식의 인자 전달 지원
	재귀 호출	<ul style="list-style-type: none"> 함수의 재귀호출 지원 반환값 기능이 없기 때문에 전역변수를 활용하여 매 재귀 호출의 계산 값 관리

<표 2>는 <표 1>의 분석 결과를 바탕으로 개발된 스크래치 예제들과 그 예제들을 통해 학습할 주요 교육 내용을 정리한 것이다.

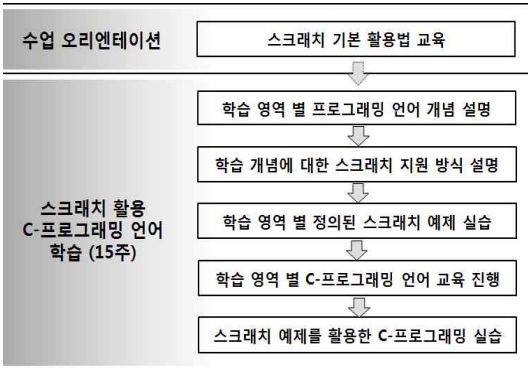
<표 2> 스크래치 예제들과 예제를 통해 학습할 주요 교육 내용

구분	스크래치 예제	교육 내용
변수와 자료형	[변수 생성] 고양이와 개 스프라이트를 등장시키고, 고양이의 변수 [나이]를 <이 스프라이트에서만 사용> 옵션으로 생성, 개의 변수 [이름]과 [나이]를 <모든 스프라이트에서 사용> 옵션으로 생성, 무대의 변수 [배경이름] 생성하기	<ul style="list-style-type: none"> 변수 생성 지역변수 활용 범위 확인 전역변수 활용 범위 확인 변수 이름 충돌 원리 이해
입력과 출력	[퀴즈 게임] 고양이 스프라이트가 퀴즈를 내고 사용자가 키보드로 답을 입력. 고양이는 정답인지 오답인지 말풍선으로 알려 줌	<ul style="list-style-type: none"> 키보드 입력 처리 기본 원리 문자열 출력 기능 경험
연산	[계산기 #1] 사용자로부터 두	<ul style="list-style-type: none"> 대입연산

	수와 연산 종류를 받아 계산 결과를 알려 줌	<ul style="list-style-type: none"> 산술연산
	[입장권 계산 #1] 입장객의 나이가 12세 이하거나 60세 이상이면 할인된 요금을 알려 줌	<ul style="list-style-type: none"> 관계연산 활용 논리연산 활용
	[윤년 계산] 사용자가 입력한 연도가 윤년인지 계산 함	<ul style="list-style-type: none"> 조건 작성
문장 제어	[짝수 구분] 사용자가 입력한 수가 짝수인지 알려 줌	<ul style="list-style-type: none"> 나머지 연산 활용
	[짝수, 홀수 구분] 사용자가 입력한 수가 짝수인지 또는 홀수인지 구분하여 알려 줌	<ul style="list-style-type: none"> if 선택 제어, if-else 선택 제어 활용
	[입장권 계산 #2] 입장객의 나이가 12세 이하거나 60세 이상, 또는 장애인 여부에 따라 요금을 할인 함	<ul style="list-style-type: none"> 선택 제어의 중첩 이해, 활용
	[1부터 무한히 수 세기] 1부터 1씩 큰 수를 무한히 말하기	<ul style="list-style-type: none"> 무한 반복 제어 활용
	[1부터 N까지 수 세기] 1부터 사용자가 입력한 수까지 말하기	<ul style="list-style-type: none"> 횟수 반복 제어 활용
	[구구단 외기] 2단부터 9단까지 구구단 말하기	<ul style="list-style-type: none"> 반복 제어의 중첩 활용
배열	[반복해서 짝수, 홀수 구분] 사용자가 'Q'를 입력할 때까지 수를 받아 짝수, 홀수 구분하여 알려 줌	<ul style="list-style-type: none"> 반복 제어와 선택 제어의 중첩 이해, 활용
	[요일 리스트] 임의의 순서로 입력된 요일들을 순서에 맞게 리스트로 만들고 리스트 내 원소를 차례대로 말하기	<ul style="list-style-type: none"> 배열의 원소 추가, 삽입, 삭제, 열람 기능 활용
	[최대 수 찾기] N개의 임의의 수로 이루어진 리스트에서 최대의 수를 찾기	<ul style="list-style-type: none"> 배열의 순회 방법 실습
함수	[계산기 #2] 사용자 인터페이스 프로시저, 사칙연산 계산 프로시저, 결과값 출력 프로시저로 구성된 두 수의 사칙연산 계산기 구현	<ul style="list-style-type: none"> 작업의 분해 개념 실습 함수 작성, 호출 실습
	[1부터 N까지 더하기] 재귀호출로 1부터 N까지의 수 더하기	<ul style="list-style-type: none"> 재귀호출의 기본 프로그램 구조 실습
	[1부터 N까지 곱하기] 재귀호출로 1부터 N까지의 수 곱하기	<ul style="list-style-type: none"> 재귀호출의 동작원리 실습
	[유클리드의 최대공약수 구하기 알고리즘] 재귀호출로 유클리드의 최대공약수 구하는 알고리즘 구현하기	<ul style="list-style-type: none"> 전역변수를 활용한 재귀호출 방법 실습

3.2 수업 모형 설계

수업 모형의 기본 절차는 다음과 같다. 우선 당일 수업 범위에 포함되는 프로그래밍 언어의 개념들을 이론적으로 교육하고, 그 개념들을 스크래치가 지원하는 방식을 설명한다. 그 후 스크래치 예제들의 실습을 통해 학습한 개념을 체득하도록 하고, C언어로 문법 교육과 실습을 진행한다. 이 때, C언어 교육의 시간적 손실을 최소화하기 위해 수업 오리엔테이션 시간에 스크래치의 기본 프로그래밍 교육을 수행하고 C언어 프로그래밍 실습에 앞서 경험한 스크래치 예제들을 포함시킨다. ((그림 2) 참고)



(그림 2) 스크래치 활용 C언어 수업 절차모형

<표 3>은 C언어의 문장 제어 영역을 교육할 때 스크래치 활용 수업 모형의 예를 보여준다. 학생들은 선택 제어와 반복 제어의 이론 강의 (선택과 반복 제어의 개념, 기본 원리, 일상생활에서의 활용 예 등)를 학습하고 대표적 선택 제어 방식 (if 선택과 if-else 선택)과 반복 제어 방식 (무한 반복, 횡수 반복, while 반복 (조건이 참인 동안 반복), until 반복 (조건이 거짓인 동안 반복))을 이해한다. 그리고 스크래치가 선택 제어와 반복 제어를 지원하는 방식을 학습한 후 표 2에서 정의한 문장 제어 영역의 스크래치 예제들을 실습한다. 이 과정이 끝나면 C언어가 지원하는 문장 제어 방식으로 학습 내용을 확장하고 (switch 선택, do-while 반복 포함) 앞서 실습한 스크래치 예제들을 포함하여 C언어 프로그래밍을 실습한다.

스크래치는 문법의 어려움과 오류 수정의 부담이 거의 없기 때문에 개념 학습에 집중하고 문제 해결 중심으로 실습할 수 있다는 장점이 있고, 이런 특징으로 수업에 참여한 학생들의 수업 참여도가 높아지고 집중하는 시간이 길어짐을 확인할 수 있었다.

<표 3> 문장 제어 학습의 스크래치 활용 수업 모형 예

단계	수업 내용
개념 이론 강의	<ul style="list-style-type: none"> • 선택, 반복 제어의 개념, 원리 • 일상에서 선택, 반복 제어 활용의 예 <ul style="list-style-type: none"> - 1부터 N까지 수 세기 - 짝수, 홀수 구분, 구구단 외기 등 • 대표적 선택, 반복 제어 방식 <ul style="list-style-type: none"> - if 선택, if-else 선택 - 무한 반복, 횡수 반복, while 반복, until 반복
스크래치 실습	<ul style="list-style-type: none"> • 스크래치의 문장 제어 방식 <ul style="list-style-type: none"> - 선택 제어 (if 선택, if-else 선택) <ul style="list-style-type: none"> - 반복 제어 (무한 반복, 횡수 반복, until 반복) <ul style="list-style-type: none"> • 문장 제어 영역 스크래치 예제 실습 <ul style="list-style-type: none"> - 짝수 구분 (if 선택) - 짝수, 홀수 구분 (if-else 선택) - 입장권 계산 #2 (선택 제어 중첩) - 1부터 무한히 수 세기 (무한 반복) - 1부터 N까지 수 세기 (횡수 반복) - 구구단 외기 (반복 제어 중첩) - 반복해서 짝수, 홀수 구분 (반복, 선택 제어 중첩)
C언어 교육	<ul style="list-style-type: none"> • C언어의 문장 제어 문법 <ul style="list-style-type: none"> - 선택 제어 (if 선택, if-else 선택, switch 선택) - 반복 제어 (for 반복, while 반복, do-while 반복)
C언어 실습	<ul style="list-style-type: none"> • 스크래치 예제들을 C언어로 구현 <ul style="list-style-type: none"> - while 반복 제어에서 무한 반복 구현 방식 학습 - for 반복 제어에서 횡수 반복 구현 방식 학습

	<ul style="list-style-type: none"> • switch 선택 예제 실습 <ul style="list-style-type: none"> - 계산기 (연산 종류에 따른 수행할 계산을 switch 선택으로 구현) • do-while 반복 예제 실습 <ul style="list-style-type: none"> - 1부터 N까지 수 세기를 do-while 구문으로 구현
--	--

4. 수업 모형 실험 및 결과 분석

4.1 실험 사전 작업

수업 모형의 평가를 위해 모 지방 사립대의 S W진공 신입생들을 대상으로 두 개의 분반으로 나누어 C언어 수업을 진행하였다. 한 분반 (스크래치 활용 반, 20명)은 본 연구의 스크래치 활용 수업 모형을 적용하였고 다른 한 분반 (스크래치 비활용 반, 21명)은 C언어 교재만으로 수업을 진행하였다. 분반에 따라 수업 방식이 상이함에 대해 사전에 학생들의 동의를 얻었고 두 분반 학생들의 프로그래밍 언어에 대한 경험을 알아보기 위해 설문조사를 진행하여 <표 4>와 같은 결과를 얻었다. 프로그래밍 언어를 경험한 학생들은 모두 스크래치를 배운 경험이있으며 C언어와 100라인 이상의 프로그램을 작성한 경험은 없었다.

<표 4> 프로그래밍 언어 경험에 대한 사전 설문조사

설문 내용	활용 반 (20명)	비활용 반 (21명)
프로그래밍 언어를 배운 경험이 있는가?	- 있음: 1명 - 없음: 19명	- 있음: 2명 - 없음: 19명
C-프로그래밍 언어를 배운 경험이 있는가?	- 있음: 0명 - 없음: 20명	- 있음: 0명 - 없음: 21명
100라인 이상의 프로그램 작성 경험이 있는가?	- 있음: 0명 - 없음: 20명	- 있음: 0명 - 없음: 21명

4.2 실험 방법 및 결과 분석

스크래치 활용 반과 비활용 반의 평가는 중간고사와 기말고사, 그리고 프로그래밍 실습 과제로 이루어졌다. 중간고사와 기말고사의 문제 유형은 크게 ‘개념과 문법 이해’, ‘단순 코드 작성 능력’,

그리고 ‘알고리즘 이해와 프로그래밍 응용 능력’ 등으로 구분하였고 두 반 모두 동일한 문제를 출제하였다. 실습 과제는 수업 시간 외에 이루어지는 활동이기 때문에 객관적인 평가 자료로 활용하기 어려운 면이 있다. 이 에 본 연구에서는 두 반의 중간고사와 기말고사 점수로만 학습 효과를 비교하였다.

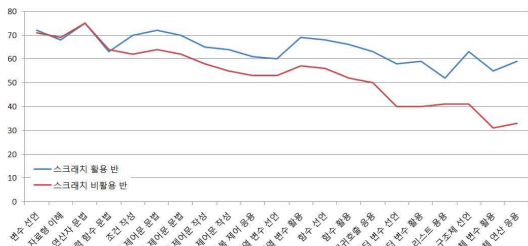
<표 5> 문제 유형별 분반 점수

문제 유형		스크래치 활용 반	스크래치 비활용 반
개념과 문법 이해	변수 선언	72	71
	자료형 이해	68	69
	연산자 문법	75	75
	입출력 함수 문법	63	64
	선택 제어문 문법	72	64
	반복 제어문 문법	70	62
	배열 변수 선언	60	53
	포인터 변수 선언	58	40
	함수 선언	68	56
단순 코드 작성 능력	구조체 선언	63	41
	조건 작성	70	62
	단순 선택 제어문 작성	65	58
	단순 반복 제어문 작성	64	55
	배열 변수 활용	69	57
	함수 활용	66	52
알고리즘 이해와 프로그래밍 응용 능력	포인터 변수 활용	59	40
	구조체 변수 활용	55	31
	조건, 선택, 반복 제어	61	53
	함수 재귀호출 응용	63	50
	포인터 리스트 응용	52	41
구조체 연산 응용	59	33	

<표 5>는 중간고사와 기말고사에 출제된 문제 유형 별로 스크래치 활용 반과 비활용 반의 평균 점수 (유형 문제별로 100점 만점 환산, 반올림 처리)를 정리한 것이다. 개념과 문법 이해 유형의 변수 선언, 자료형 이해, 연산자 문법, 그리고 입출력 함수 문법 문제들에 대해서는 두 반의 점수 차이가 크지 않은데 이는 학습 내용 자체가 수월하고 스크래치의 변수 선언, 자료형의 처리 방식, 그리고 입출력 기능 등이 C언어와 크게 상이하기

때문으로 해석된다. 하지만 선택과 반복 제어문 문법, 배열 선언, 그리고 함수 선언 문체에 대해서는 스크래치 활용 반이 비활용 반에 비해 평균 14.9% 정도 점수가 높았는데 이는 스크래치 활용을 통해 선택과 반복 제어의 개념을 상대적으로 명확하게 이해했음을 보여준다. 단순 코드 작성 유형의 조건 작성, 단순 선택과 반복 제어문 작성, 배열 변수 활용, 그리고 함수 활용 문제들은 스크래치 활용 반이 비활용 반에 비해 평균 17.6% 정도 높았는데 이는 문법적 어려움이 적고 오류 수정이 용이한 스크래치의 실습이 개념 이해와 활용 능력 배양에 효과가 높았음을 보여준다. 이런 효과는 알고리즘 이해와 프로그래밍 응용 능력 유형(스크래치가 활용되지 않은 포인터 리스트와 구조체 연산 응용문제 제외)의 문제 해결에도 도움을 주어 스크래치 활용 반이 비활용 반에 비해 20.4% 가량 점수가 높았다.

흥미로운 결과는 수업 내용 전반에 걸쳐 스크래치를 활용한 반이 비활용 반에 비해 평가 점수가 높다는 것이다. 예를 들어, 수업 후반부의 포인터와 구조체는 스크래치를 활용하지 않는 학습 내용임에도 불구하고 스크래치 활용 반이 비활용 반에 비해 53.1% 정도 점수가 높게 나타났다. 이런 결과는 스크래치 활용 반 학생들이 비활용 반 학생들에 비해 수업 후반부까지 C언어에 대한 흥미를 잃지 않고 상대적으로 높은 집중력과 참여도를 유지하였기 때문으로 해석된다.



(그림 3) 스크래치 활용 반과 비활용 반의 문제 유형별 성적 비교 (진도 순으로 문제 나열)

(그림 3)은 평가 문제들을 수업에서 배운 시간 순으로 나열하고 스크래치 활용 반과 비활용 반의

점수를 비교한 것이다. 그림에서 볼 수 있듯이 수업 후반부로 갈수록 두 반의 점수 차가 커짐을 알 수 있는데 이는 수업의 난이도가 높아짐에 따라 스크래치 비활용 반 학생들의 수업 집중도와 참여도가 떨어져 수업을 포기하는 학생들이 많아졌기 때문이다. 따라서 스크래치 활용의 효과는 스크래치 실습을 적용한 학습 내용에 국한하지 않고 평가해야 함을 알 수 있다.

<표 6> 스크래치 활용 반과 비활용 반의 성적에 대한 t-검정 분석

구분	학생 수	평균	표준편차	t 통계량	유의확률
활용 반	20	59.7	9.8	4.03	0.000268
비활용 반	21	42.7	15.7		

<표 6>은 스크래치 활용 반과 비활용 반의 중간고사와 기말고사의 통합 평균 점수로 t 검정을 수행한 것이다. 표의 결과는 스크래치 활용 반이 비활용 반에 비해 통계적으로 유의미하게 더 우수한 성적을 받았다는 것을 보여준다.

4.3 스크래치 활용에 대한 설문 조사

(그림 4)는 스크래치 활용 반 학생들을 대상으로 스크래치의 활용이 C언어 수업에 어떤 영향을 주었는지 조사한 설문 결과를 보여준다. 설문 조사 내용은 스크래치가 수업에 어느 정도 도움이 되었는지, 도움이 되었다면 어떤 영역에 도움이 되었는지를 파악하기 위한 문항으로 구성하였다. 설문 조사 결과를 보면 95%의 학생이 스크래치가 도움이 되었다고 응답하였다. 스크래치가 전혀 도움이 되지 않았다고 응답한 학생은 이미 스크래치를 익힌 학생으로 스크래치 수업이 오히려 C언어 수업 시간을 단축시키고 집중도를 떨어뜨린다고 느꼈다.

스크래치가 도움이 되었다는 19명의 학생들을 대상으로 스크래치가 도움이 된 유형을 살펴보면 프로그래밍 언어 수업에 흥미를 갖는 데 도움이 되었다는 응답이 18명으로 가장 높았고 그 다음으

로 문제해결을 위한 알고리즘 고안에 도움이 되었다는 응답 (15명)과 C-프로그래밍 언어의 문법을 익히는 데 도움이 되었다는 응답 (4명) 순으로 높았다. 이와 같은 결과는 스크래치 활용 반 학생들이 스크래치가 활용되지 못한 수업 후반부 학습 내용에 대해서도 더 우수한 성적을 낸 것과는 부합한다. 즉, 스크래치 활용으로 인해 높은 수업 흥미도를 보인 결과 집중도와 참여도가 유지되고 결과적으로 수업 내용 전반에 걸쳐 우수한 성적을 냈다고 볼 수 있다.

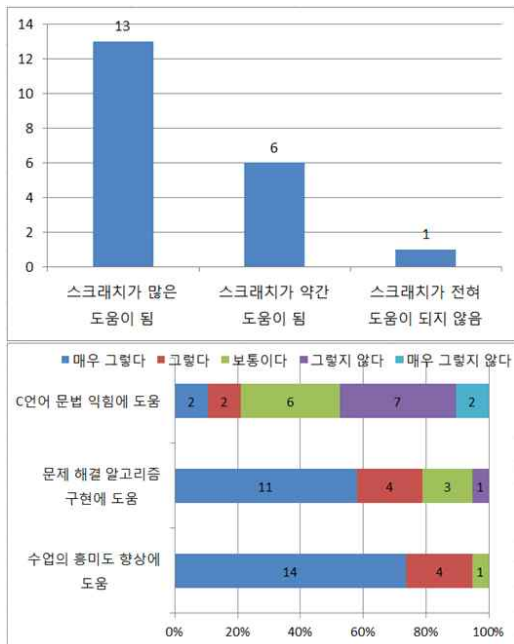
또한, 설문지의 응답 결과를 볼 때 스크래치 실습을 통해 문법과 오류 수정의 어려움 없이 문제 해결에 집중할 수 있었던 경험이 알고리즘 구현에 도움이 되었음을 알 수 있다. 하지만 스크래치 활용이 C언어 문법 익힘에 도움이 되었는가에 대해서는 현저하게 낮은 응답이 나왔기 때문에 스크래치 문법과 C언어 문법 간의 연계성 또는 호환성 학습이 가능한 수업 모형 연구의 여지를 남기고 있다.

5. 결 론

본 연구는 기초학력이 낮은 SW전공자의 프로그래밍 언어 입문 교육의 어려움을 완화하는 방안으로 C언어 수업에 스크래치를 활용하는 수업 모형을 제시하였다. 스크래치 활용에 있어서 그 활용 범위와 내용을 명확하게 하기 위해서 C언어가 지원하는 프로그래밍 개념들 중 스크래치로 교육 가능한 개념들을 파악하고 그 개념들의 이해를 돕는 스크래치 예제들을 개발하였다. 수업 모형은 스크래치 예제 실습을 통해 프로그래밍 개념을 우선 이해하고 교육의 내용을 C언어로 확장하는 절차를 가지며 스크래치 예제들을 C언어 실습에도 활용하여 스크래치 활용의 시간적 부담을 줄였다. 본 연구는 수업 모형의 효과를 검증하기 위해 모 지방 사립대 SW전공 신입생들의 C언어 수업을 본 연구의 수업 모형을 적용한 반 (스크래치 활용 반)과 적용하지 않은 반 (스크래치 비활용 반)으로 구분하여 학습 성과를 비교 분석하였다.

결론적으로, 스크래치 활용 반의 학생들이 비활용 반의 학생들에 비해 통계적으로 유의미하게 성적이 높았다. 학습 내용 유형별로 성적을 비교해보면 스크래치가 활용된 내용뿐만 아니라 스크래치가 활용되지 않은 내용 (예: 포인터, 구조체 등)에서도 스크래치 활용 반 학생들의 성적이 높았는데 이는 스크래치의 활용이 학생들의 수업에 대한 흥미를 높여 집중도와 참여도를 향상시킨 결과로 볼 수 있다. 이와 같은 해석은 스크래치 활용 반 학생들의 설문 조사를 통해서도 뒷받침 되는데 스크래치 활용이 수업의 흥미도 향상에 도움이 되었다는 응답이 가장 높았다.

본 연구는 C언어 수업 모형에 있어서 스크래치의 활용 범위와 교육 내용을 구체적으로 제시하였다는 것과 실험을 통해 수업 모형의 효과를 검증하였다는 점에 의의가 있다. 하지만 수업 모형 검증 실험에 참여한 학생들의 수가 충분하지 못하다는 한계를 갖고 있기 때문에 향후 새로운 SW전공 신입생들을 대상으로 반복적인 실험을 수행하면서 수업 모형을 검증하고 발전시킬 계획이며, 끝으로 본 연구가 보안 IT 분야 학생들의 프로그



(그림 4) 스크래치 활용에 대한 설문 조사 결과:

(위) 스크래치 활용의 도움 정도

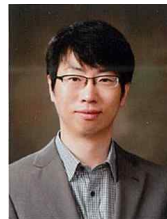
(아래) 스크래치 활용의 도움 유형

래밍 언어 입문 교육에 활용되어 도움이 되길 기대한다.

참고문헌

- [1] 소프트웨어중심사회 웹사이트.
<http://www.software.kr/um/um02/um0202/um020202.do>
- [2] 스크래치 웹사이트. <https://scratch.mit.edu/>.
- [3] 엔트리 웹사이트. <https://playentry.org/#/>.
- [4] 송정범, 조성환, 이태욱, “스크래치 프로그래밍 학습이 학습자의 동기와 문제해결력에 미치는 영향”, 정보교육학회논문지, 제12권, 제3호, 2008. 9.
- [5] 배학진, 이은경, 이영준, “문제 중심 학습을 적용한 스크래치 프로그래밍 교수 학습 모형”, 컴퓨터교육학회논문지, 제12권, 제3호, 2009. 5.
- [6] 이은경, “문제해결력 향상을 위한 과제 중심 스크래치 프로그래밍 학습 프로그램”, 컴퓨터교육학회논문지, 제12권, 제6호, 2009. 11.
- [7] 김태훈, 김종훈, “스크래치 프로그래밍 중심의 STEAM 교육프로그램 개발 및 적용”, 컴퓨터교육학회논문지, 제17권, 제6호, 2014. 11.
- [8] 오미자, “스크래치 프로그램을 활용한 프로그래밍 교육에 대한 비전공자의 인식 연구”, 컴퓨터교육학회논문지, 제20권, 제1호, 2017. 1.
- [9] 피수영, “IT 융합교육을 위한 비전공자 코딩 교육의 발전방안”, 디지털융복합연구, 제14권, 제10호, 2016. 10.
- [10] 이민정, “비전공자 SW 프로그래밍 교육과정 및 콘텐츠 개발 모형의 효과성 탐색: SW 해결안의 시각적 표현을 중심으로”, 디지털콘텐츠학회논문지, 제18권 제7호, 2017. 11.
- [11] 조성환, 송정범, 김성식, 백성혜, “스크래치를 이용한 프로그래밍 수업 효과”, 정보교육학회논문지, 제12권, 제4호, 2008. 12.
- [12] 노희진, 백성혜, “스크래치를 활용한 고등학교 과학 수업에 대한 학생 인식”, 한국교육학회지, 제35권, 제1호, 2015.
- [13] 오정철, 이지훤, 김정아, 김종훈, “스크래치를 활용한 STEAM 기반 교육 프로그램 개발 및 적용: 초등학교 6학년 과학교과를 중심으로”, 컴퓨터교육학회논문지, 제15권, 제3호, 2012. 5.
- [14] 안형진, 마대성, “문제해결력 증진을 위한 초등학교 Scratch 교육과정 개발”, 정보교육학회논문지, 제17권, 제3호, 2013. 9.
- [15] 박정신, 조석봉, “프로그래밍입문 수업에서 스크래치 활용 효과분석”, 디지털정책연구, 제10권, 제9호, 2015. 10.
- [16] MIT 미디어랩 Lifelong Kindergarten 웹사이트.
<https://www.media.mit.edu/groups/lifelong-kindergarten/>.

〔저자소개〕



고 광 일 (Kwangil KO)
1993년 2월 포항공대 학사
1995년 2월 포항공대 석사
1999년 8월 포항공대 박사
현재 우송대학교 테크노미디어융합학부 영상콘텐츠전공 교수
email: kwangil.ko@gmail.com