
프로그래밍입문 수업에서 스크래치 활용 효과분석

박정신*, 조석봉**

The Effect of teaching Scratch in introductory programming course

JungShin Park*, SeokBong Cho**

요 약 초보자들은 프로그래밍 입문 수업을 통해 프로그래밍 언어의 문법적 지식뿐만 아니라, 알고리즘 구상과 프로그램 작성을 위한 문제해결력을 키우는 과정이 필요하다. 문제를 해결하기 위한 문제해결력을 키우는 과정 없이 프로그래밍언어의 지식과 문법위주의 교육이 추가 되는 현실에서 기초 학력수준이 상대적으로 낮은 전문대학의 컴퓨터 전공 학생들이 프로그래밍언어 수업에서 느끼는 어려움은 더 크다고 할 수 있다. 본 연구에서는 교육용 프로그래밍 언어인 스크래치를 활용한 학생들이 활용하지 않은 반 학생들에 비해 문제 해결력, 프로그래밍 작성 능력, 수업의 만족도면에서 크게 향상되었음을 비교·분석하였다.

주제어 : 스크래치, 문제해결력, 프로그래밍 언어, 교육용프로그래밍언어, 프로그램 작성능력

Abstract The college students who have relatively weak academic background feel more difficult in learning programming language grammars and programming skills in introductory course. At the end of semester, most of students had the negative attitude to programming and only a few students could write the programs for the given problems because they spent most of time to learn grammars instead of learning problem solving skills and logics. In this study, we propose to use Scratch in introductory programming course to help students to understand grammars and problem solving skills. It's necessary to educate first-time programmers how to solve the problems before they learn grammars of the programming language in their first programming language course. This paper shows that Scratch allows students not only to learn problem solving skills in programming but also to motivate students themselves in the class.

Key Words : Scratch, problem solving skill, programming language, Educational Programming Language, programming ability

1. 서론

컴퓨터를 전공하는 학생들에게 프로그래밍 언어 교과목은 컴퓨터 세상과 소통하기 위한 기본적인고도 중요한 과목이지만 가장 어렵게 생각되는 과목이기도 하다. 특히 기초 학력수준이 상대적으로 낮은 전문대학의 컴퓨터 전공 신입생의 경우에는 영어로 이루어져있는 프로그래밍 언어를 배우고 익히는 데 더 큰 어려움을 겪고 있다. 채유진(2006)에 따르면 프로그래밍과 알고리즘에 대한 개념이 없는 학생들은 프로그래밍을 하기위해 필요한

기본 문법과 구조를 이해하는데 너무 많은 노력이 요구되어진다. 따라서 학습자는 프로그램 학습초기에 가지고 있던 의욕이나 동기를 상실하고 프로그래밍에 대한 부정적인 인식만 남게 된다고 하였다[12]. 처음 컴퓨터 프로그래밍과목을 접하는 학생들을 대상으로 프로그래밍언어 교육을 효과적으로 실시하는 것은 상당히 어려운 일 이므로 이를 좀 더 쉽고 효율적으로 전달하기 위해 교육용 로봇[5]이나 교육용 프로그래밍 언어 (EPL: Educational Programming Language)와 같은 교육용 도구가 도입되고, 다양한 교육프레임 워크가 제안되는 등

*우송정보대학 컴퓨터정보과 조교수

**우송정보대학 컴퓨터정보과 교수

논문접수: 2012년 9월 6일, 1차 수정을 거쳐, 심사완료: 2012년 10월 15일

지속적으로 연구되고 있다[13].

전문대학 학생들의 학습수준에 대한 연구를 살펴보면 박성미(2009)는 전문대학생들의 학습력 부족에 따른 대학 교육의 질 저하에 대한 우려가 되고 있으며, 입학과 동시에 고난이도의 전공 관련 교과목이 운영되고 있어 대학생에게 학습 부진 현상이 나타나고 있다고 한다[6]. 또한 대학 전공에서의 학습 부진은 졸업 후 직장에서 문제 해결 능력 부족, 낮은 조직 적응력 등으로 이어질 가능성이 높은 것으로 연구되고 있다고 하였다[1][6]. 이러한 면에서 보면 컴퓨터를 전공하는 전문대학생들에게 프로그래밍언어를 효과적으로 가르치기 위해서는 프로그래밍에 대한 흥미를 가질 수 있도록 좀 더 체계적인 문제 해결 능력을 키우는 과정과 프로그래밍언어 지식을 습득하는 과정을 병행함으로써 학습 부진 현상을 극복해 나가는 수업모형이 도입되어야 할 것이다.

본 연구에서는 초등학생부터 프로그램 초보자들까지 문제해결력과 알고리즘을 이해하고 훈련하기에 적합한 교육용 프로그래밍 언어 스크래치를 도입하였다. EPL인 스크래치는 기억해야할 복잡한 문법이 없고 초보자도 쉽게 배울 수 있으며, 기능에 맞는 컬러 블록들을 이리저리 이동하여 문제를 해결하고 결과를 화면에 보여주는 비주얼 프로그래밍 도구이다. 조성환 외(2007)는 스크래치를 통해 학습자들의 동기 유발은 물론이고 프로그래밍 교육에서 근본적으로 이루고자 하는 목적인 창조적으로 생각하고 문제를 체계적으로 해결하며 다양한 방식으로 새로운 형태의 문제를 접근할 수 있는 알고리즘 사고방식을 기를 수 있다고 하였다[10].

본 대학 컴퓨터정보과의 신입생들은 프로그래밍언어 교과목을 첫 학기에 이수하여야 하는 데 처음으로 접하는 프로그래밍 언어 수업은 학생들에게 장기적으로 전공을 지속할 것인지 포기할 것인지를 결정하는 중요한 첫 단계가 되어 왔다. 본 연구에서는 C언어 교재와 함께 스크래치를 활용하여 수업을 진행하는 반과 기존의 방식과 같이 C언어 교재만을 사용하는 반으로 나누고 스크래치를 활용한 프로그래밍 언어수업의 학습 효과분석, 문제해결력 차이분석과 수업 만족도를 비교·분석하였다.

2. 기존 연구

프로그래밍 교육을 효율적으로 전달하기 위한 다양한

연구와 스크래치를 활용한 교육의 효과를 분석한 기존의 연구를 다음과 같이 정리하였다.

송정범, 조성환, 이태욱(2008)은 프로그래밍 과정에서 학습자의 내적 동기 유발을 위한 전략과 창의적 문제 해결 수업모형(CPS:Creative Problem Solving)을 토대로 스크래치 프로그래밍 학습은 학습자의 내재적 동기와 문제 해결력 향상에 효과가 있는 것으로 나타났다고 하였다[7]. 김수환 외(2011)는 스크래치 프로그래밍 언어를 활용하여 CL(Computational Literacy)교육에서 다중지능을 고려한 교육 방법을 제안하고 효과를 검증하였다. 학습자의 다중 지능을 고려한 프로그래밍 교육이 학습자의 흥미와 동기를 부여하여 프로그래밍 수업을 효과적으로 신장하는 데 기여함을 연구하였다[4]. 조성환 외(2007)는 스크래치를 CPS 모형을 활용해 적용해 봄으로써 중학교 남녀 학습자에게 있어 문제 해결력 향상과 프로그래밍 교육에 대한 태도에 어떠한 영향을 미치는 지 검증하였다[11]. 최현중 (2010)은 초보 프로그래머가 겪는 문제점들을 정리하고 초보 프로그래머의 실력을 향상시키기 위해 프로그래밍 개발 과정, 프로그래밍 학습 요소, 교수학습 방법을 세 개의 차원으로 하는 프로그래밍 교육 프레임 워크 제안하였다[13]. David J. Malan, Henry H. Leitner (2007)는 하버드 대학 학생들 중 전공, 비전공과는 상관없이 자바과목을 듣기 전에 스크래치를 입문과정으로 채택한 결과 76%의 학생들이 스크래치가 도움이 많이 되었다고 응답하였다고 하였다[16]. 김미선, 박관우(2011)는 스크래치 언어를 이용하여 교육과정에 적용되는 학습내용을 구성하였다[2].

PBL(Project Based Learning)기반의 프로그래밍 수업에서 학습양식에 따라 학습동기에 어떠한 차이가 있는지를 분석한 연구도 있다[3]. 안유정, 김경아 (2010)는 효과적인 컴퓨터 프로그래밍 학습을 위해 정규수업과 병행하여 심화, 보충형 수준별 학습통한 효과를 검증하였다[8]. 1960년 이후 수많은 프로그래밍언어들이 만들어지고 사용되고 있으며 연령불분하고 초보자가 배우기 쉽고 가르치기 쉬운 프로그래밍언어의 특성 및 분류방법들을 제시한 연구도 있다[15].

3. 스크래치 특징

스크래치는 MIT 미디어 연구소에서 개발한 교육용

프로그래밍도구이다. 미국 과학재단, 마이크로소프트, 인텔, 노키아, MIT 미디어 연구소가 연구 컨소시엄의 후원을 받아 개발하였으며 프로그래밍 학습을 효율적으로 지원할 수 있는 다양한 비주얼 적인 요소들을 가지고 있다 [9][17]. 컬러로 구분되는 블록들을 서로 끼워 맞춰서 프로그램을 작성하며 기능의 특성상 또는 문법상 맞지 않는 블록들을 서로 연결하려고 하면 블록들이 연결되지 않고 튕겨 나가기 때문에 초보자들이 갖는 오류에 대한 부담이 생기지 않는다. 스프라이트라는 객체를 생성하고 동작시키는 과정에서 이러한 블록들을 사용하므로 스크래치는 “비주얼프로그래밍 언어”로 분류된다[14].

총 8개의 블록 팔레트와 102개의 블록으로 구성되어 있는 스크래치 화면 구성은 다음과 같다[9].



[그림 1] 스크래치 화면 구성

(1) 블록 팔레트

8가지의 다른 특성을 지닌 컬러 블록 팔레트로 구성된다. 각 팔레트를 선택하면 같은 색상의 해당 프로그래밍 기능을 가지고 있는 블록들의 목록을 볼 수 있다. 예를 들어 동작 블록 팔레트를 선택하면 스프라이트의 움직임을 제어하는 블록들이 나타난다.

(2) 스크립트, 모양, 소리 탭 영역

스프라이트 모양을 바꾸거나 소리를 추가하기 위해 필요한 기능을 포함하고 있다.

스크립트 탭을 선택하면 블록 팔레트의 원하는 명령어들을 가져와서 스프라이트의 동작을 제어하거나 프

그래밍을 실습할 수 있는 공간이 주어진다.

(3) 스테이지

블록들로 구성된 스크립트를 실행하면 스테이지에 결과가 나타난다.

(4) 스프라이트 영역

스크래치는 스프라이트라는 객체를 만들고 사용하면 시 프로그래밍 실습을 하게 된다.

[그림 2]에서는 수업시간에 스크래치를 사용하여 주어진 두수가 짝수인지 홀수인지를 판별하는 코드를 작성하고 실행한 결과 화면을 보여주는 예이다.



[그림 2] 스크래치 코드 실행 화면

4. 연구 방법 및 내용

4.1 연구 내용

본 대학 컴퓨터정보과 1학년 1학기에 개설되는 “프로그래밍언어” 교과목수업을 받는 2개의 반을 대상으로 연구를 실시하였다. 이 수업은 C 프로그래밍 언어를 배우며 다른 프로그래밍 언어를 배우기전의 첫 번째 언어 수업이다. 대부분의 학생들은 이 교과목을 이수한 후 프로그래밍 언어에 더 흥미를 느껴 졸업 후의 진로를 개발 분야로 나아가고 싶어 하는 그룹과 컴퓨터전공 자체에 관심을 잃어버리거나 너무 어렵다고 생각해서 낙오하는 그룹으로 나누어지게 되므로 학생들의 진로를 결정하는 척도가 되는 수업이라고 할 수 있다. 한 반은 26~28명으로 구성되며 교육범위는 C언어 기초 문법부터 활용까지이다. 주 4시간 15주 수업이고 이론과 실습이 혼합된 과

목이다. 선택된 두 개의 반은 스크래치 활용반과 스크래치 비 활용 반으로 나뉘며 한 학기가 끝난 후 학생들의 학습결과와 문제해결능력, 프로그램 작성 능력, 수업만족도를 평가·분석하였다.

4.2 사전 조사

수업을 진행하기 전에 학생들을 대상으로 이전에 프로그래밍언어를 배운 경험이 있는 지를 설문조사하였다. 스크래치를 활용하는 반의 경우 28명중 2명이 C 프로그래밍언어를 배운 경험이 있었고 스크래치를 활용하지 않는 반의 경우 3명이 비주얼 베이직, C 프로그래밍 언어를 배운 경험이 있는 것으로 조사되었다. 하지만 프로그램 작성 경험을 묻는 질문에서는 모두 대답을 하지 못하였다.

〈표 1〉 사전 설문조사 결과

	스크래치 활용반 (28명)	스크래치 비활용반 (26명)
있다. (1개월 이상)	1명	2명
조금 있다 (1개월 미만)	1명	1명
없다.	26명	23명



4.3 수업모형

〈표 2〉는 스크래치 활용 반에서 제어문을 교육 했을 때의 수업진행모형 예이다. 학생들은 C언어 제어문의 특징과 구조를 포함하여 기본 개념을 학습한 후 스크래치에 있는 제어문을 활용하여 스프라이트를 동작시키는 재미있고 다양한 예제를 반복적으로 실습한다.

이러한 실습 과정을 통해 학생들이 제어문에 대한 개념을 파악하고 코드를 생성하는 과정에 익숙하게 된다. 문법적 제약이 약한 스크래치를 활용함으로써 프로그래밍언어의 문법적 오류를 찾아내기 위한 수고를 줄이므로 문제해결력에 집중할 수가 있게 된다. 스크래치로 검증된 코드를 C언어 코드로 옮기는 과정에서 C언어만의 문법과 특징, 구조를 익히게 되고 반복적인 수정작업과 결과 확인, 오류 수정 과정을 통해 C언어에 익숙하게 된다. 스크래치는 블록으로 이루어진 명령문들을 조합하여 문제를 해결해 나가므로 특별히 암기하거나 난해한 구조를 가지고 있지 않아 학생들이 문법보다는 문제 해결하는 과정에 초점을 맞출 수 있다는 효과를 이용하였

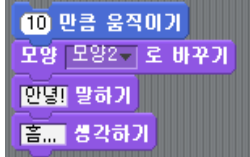
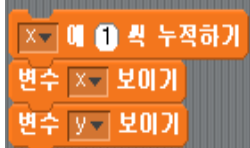
다. 또한, 비주얼적인 요소들을 사용하여 결과가 나오는 과정 때문에 학생들의 수업 참여도가 월등히 높아졌고 수업에 집중하는 시간이 길어졌다.

〈표 2〉 스크래치 활용 반에서 진행한 수업모형

수업 단계	수업 내용	비고
학습 시작 (이론 강의)	<ul style="list-style-type: none"> • 학습 목표 설정 • 제어문의 개념, 구문구조 설명 • 제어문 활용예제 코드 설명 	강의 교재 활용
스크래치 실습	<ul style="list-style-type: none"> • 제어 팔레트의 구성요소 익히기 • 제어문의 구문구조 설명 • 문제 해결력 및 알고리즘 능력 향상을 위한 예제들 ex) 스프라이트의 현재 방향이 90도인지 확인한다. • 실습예제 제시 ex) 짝수/홀수 판단하기 	
스크래치와 C병행 실습	<ul style="list-style-type: none"> • 스크래치에서 작성한 실습예제를 C로 옮기기 ex) 짝수/홀수 판단하기 • C언어 문법, 특징, 구조 익히기 	
학습 정리	실습예제 결과 체크	<pre>#include stdio.h void main() { int x=15, y; y= x%2; if (y==0) printf ("짝수"); else printf("홀수"); }</pre>

다음은 수업시간에 주로 활용되었던 스크래치의 블록들과 C언어의 개념들과의 연계성을 [16]을 참조하여 재구성하였다.

〈표 3〉 C와 스크래치의 연관 개념

C언어 개념	설명	스크래치 해당 코드
순차구조	각 단계의 처리 순서를 체계적으로 구성해야한다. 순차처리의 개념을 이해할 수 있다.	
변수	스크래치에서는 지역변수와 지역변수를 모두 지원한다.	

논리연산	and, or, not의 논리 연산을 지원한다.	
키보드입력 출력하기	사용자로부터 키보드 입력을 받아 저장한다. 화면에 문자, 변수값을 출력한다.	
제어문	조건 검토 과정을 통해 서로 다른 단계를 수행하는 과정을 통해 제어문을 이해한다.	
반복문	무한 반복과 조건에 따라 일정 단계를 반복하기 위한 프로그램을 작성함으로써 반복 구조를 이해할 수 있다.	
배열 (리스트)	스크래치는 유사한 자료형을 연속적으로 저장하거나 다양한 자료형을 연속적으로 저장하기 위한 리스트 구조 모두 지원한다.	

과 시험에 낸 10 문제에 대해 알고리즘 구상과 프로그램 작성을 통해 결과를 산출해 내는 과정을 평가하였다. 문제가 주어지고 스스로 결과가 나온 학생들의 수를 누적하여 총 인원수를 10회동안의 평균으로 계산해 나온 인원수이다.

〈표 4〉 문제 유형별 정답자 수

문제 유형	스크래치 활용반		스크래치 비활용반	
	인원	비율	인원	비율
개념 및 문법 이해	변수선언문 작성	20명 71%	18명 69%	
	데이터형 이해	16명 57%	15명 58%	
	연산자 이해	20명 71%	18명 69%	
	제어문 이해	20명 71%	12명 46%	
	반복문 이해	21명 75%	13명 50%	
	배열이해 및 선언문 작성	10명 36%	7명 27%	
	포인터이해 및 선언문 작성	11명 39%	9명 35%	
단순 코드 작성	함수 이해	9명 32%	7명 27%	
	단순 제어문사용 프로그래밍	19명 68%	10명 38%	
	단순 반복문사용 프로그래밍	17명 61%	11명 42%	
	배열사용 프로그래밍	13명 46%	9명 35%	
프로그래밍 응용	함수, 포인터 활용 프로그래밍	10명 36%	8명 31%	
	문제해결을 위한 알고리즘 구상	15명 54%	8명 31%	
프로그램 작성	12명 43%	5명 19%		

4.4 평가 방법 및 연구결과

중간고사와 기말고사를 포함한 학습 형성 평가문제와 실습을 통해 스크래치를 활용한 반과 활용하지 않은 반의 학습 결과차이를 비교하였다. 개념 및 구조적 특징을 묻는 각 문제는 1점과 0점의 값을 가지며 부분점수를 부여하지 않았다. 단순개념의 이해와 구문구조 지식의 단순 이해를 평가하는 문제항목에서는 스크래치활용 반과 스크래치 비 활용 반의 학생들 중 정답을 맞힌 학생들의 수를 비교하여 분석하였다.

코드 작성 문제 유형에서는 5점부터 0점까지의 점수 차를 부여하고 다중 제어문 사용이나 다중 반복문 사용을 배제한 문제들을 출제하고 부분점수를 허용하였다. <표 4>에서 나타난 인원수는 3점 이상의 부분점수를 받은 학생들의 수를 모두 포함한 수치이다.

프로그래밍 응용능력 평가항목에 대해서는 실습시간

변수, 데이터형, 연산자의 선언과 관련된 프로그래밍의 단순 지식들을 학습하고 익히는 과정은 스크래치 활용반과 그렇지 않은 반의 차이가 크지 않았다. 그 이유는 스크래치에서는 C언어와 같이 변수, 데이터 형, 연산자에 대한 제약이 없기 때문에 스크래치가 단순개념들에 대한 학습에 크게 도움이 되었다고 볼 수가 없다.

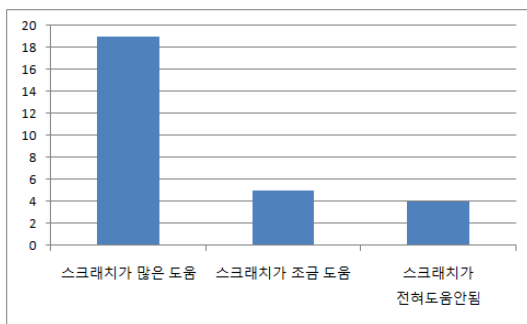
하지만 제어문, 반복문, 배열에 대한 이해를 묻는 항목과 단순 프로그래밍 코드를 작성해야 하는 항목에서는 정답을 맞힌 학생들의 수가 스크래치를 활용한 반에서 평균 20% 정도로 많았다. 프로그래밍응용 항목은 실습시간과 시험시간을 통해서 학생들이 10문제의 실습과제를 풀고 결과를 내는 과정을 추적하여 정확히 결과를 낸 학생들의 평균 인원수로 비교하였다. <표 4>에서 보여주는 바와 같이 스크래치 코드를 활용하여 다양한 실습

예제를 익힌 반의 학생들이 스크래치를 활용하지 않은 반의 학생들보다 결과를 내는 수가 평균 23%정도로 많았음을 알 수가 있다. 프로그래밍 심화 과정으로써 문제를 해결하기 위해 알고리즘을 구상하고 프로그래밍코드로 변환하는 과정은 단순 지식의 습득이 아니라 문제를 해결하는 방법을 꾸준히 실습한 결과로 나타나기 때문에 이 항목에서도 스크래치를 활용한 반의 더 많은 학생들이 알고리즘 구상과 프로그래밍 작성에 더 익숙해져있음을 알 수가 있다.

<표 5> 평균 점수 비교

	학생 수	평균	표준편차	t	유의 확률
스크래치 활용반	28	20.7	12	2.75	0.0082
스크래치 비활용반	26	10.7	10.7		

<표 5>는 두 반의 형성평가 점수를 비교한 t분포이다. 스크래치를 활용한 반과 그렇지 않은 반의 학습형성평가 점수를 비교한 결과 t분포와 유의 확률이 위의 도표와 같이 나타났다. 이는 스크래치 활용반 학생들이 C언어 개념과 문법을 더 잘 이해하고 문제해결력을 통한 프로그래밍 작성과 이해에도 더 향상된 실력을 갖는다는 사실을 보여주고 있는 결과이다.



스크래치 활용 반 학생들을 대상으로 실시한 설문조사에 따르면 85%이상의 학생들이 스크래치를 활용한 수업에 만족을 나타냈으며 C언어 수업을 이해하는데 많은 도움을 주었다고 응답하였다. 수업시간 집중도가 떨어졌던 학생들이 비주열적인 요소와 단순한 문법으로 이루어

진 스크래치를 활용함으로써 높은 만족도를 나타냈으며, 학습의욕이 향상되는 효과를 보였다. 이는 다루기 쉬운 스크래치를 통해 문제에 대한 결과를 내는 경험의 수를 높임으로써 자신감이 향상된 결과라고 생각한다. 두 반은 실습시간에 같은 문제가 주어졌으며 알고리즘을 구상하고 문제를 해결하는 과정을 관찰하였는데 스크래치를 활용한 반의 학생들은 문제를 해결하려는 의지가 더 뚜렷하였고 실제 문제를 해결하여 결과를 내는 학생들의 수도 더 높았다. <표 4 참조>

28명중 4명에 해당하는 학생들이 스크래치가 너무 쉽고 C언어 수업에 도움이 되지 않았다고 응답하였다. 스크래치가 전혀 도움이 되지 않았다고 응답한 학생들은 사전 설문조사에서 이전에 C를 포함한 다른 프로그래밍 언어를 학습한 경험이 있는 학생들과 C언어 학습에 어려움을 느끼지 않은 학생들이었다. 이미 프로그래밍언어의 문법과 특징을 습득한 학생들에게는 스크래치가 다른 하나의 프로그래밍언어로만 간주되고 오히려 C언어에 대한 집중도를 떨어뜨린다고 평가하였다.

<표 6>은 스크래치의 어떤 부분이 C언어 수업에 도움을 주었는지 설문조사한 결과를 분석하였다.

<표 6> 스크래치 활용 유형

	매우 그렇다	그렇다	그렇지 않다	매우 그렇지 않다,
스크래치가 문제해결을 위한 알고리즘을 생각해 내는데 도움이 되었다.	17	5	5	1
스크래치가 C프로그램 코드를 작성하는데 도움이 되었다	13	7	5	3
스크래치가 C수업에 흥미를 갖는데 도움이 되었다.	20	3	5	0
스크래치가 C언어 문법을 익히는데 도움이 되었다.	12	5	10	1
스크래치를 처음 프로그래밍언어를 배우는 학생들에게 추천한다.	21	4	3	0

위의 표에서 나타난 바와 같이 스크래치가 수업에 가장 큰 도움을 준 부분은 어렵고 힘들게만 인식되어왔던 C언어 수업에 흥미를 느끼는 학생들의 수가 많아졌다는 것이다. 기초 학력수준이 낮은 전문대학에서는 수업의

질을 높이면서 학생들의 흥미를 유발하여 중도탈락을 방지하는 노력들이 많이 진행되어 왔다. 스크래치의 활용은 비주얼 적 요소와 쉬운 문법, 오류에 대한 두려움 없이 결과를 바로 확인할 수 있는 특징들로 인해 학생들의 관심을 끌 수가 있었다. 스크래치에서 지원하는 기능이 C언어의 모든 특징들을 포함할 수 없지만 초보자들에게는 프로그래밍 언어 입문에 대한 두려움을 낮추었다는 데 큰 의미가 있다. 스크래치를 통해 C언어의 문법을 익히는 데 도움을 주기도 했지만 C개념을 반복적으로 학습함으로써 문제 해결을 위한 알고리즘을 구상하는 데 더 도움을 받은 학생들이 많다는 것을 알 수가 있다. 좀 더 자세히 <표 6>을 살펴보면 주어진 문제에 대한 알고리즘을 구상하는 데 78% 학생들이 스크래치활용이 도움이 되었다고 응답했으며, 평균 66%의 학생들이 C언어의 문법과 코드작성에 도움이 되었다고 응답하였다. 그리고 82%정도의 학생들이 스크래치의 활용이 C언어 수업에 대한 흥미를 갖게 해주었다고 응답하였다.

5. 결론

기초학력 부재로 인해 야기된 전문대학생들의 전공 학습 부진 현상은 졸업 후 직장에서의 문제해결능력 부족, 낮은 조직 적응력 등으로 이어질 가능성이 높다고 하였다[6].

이러한 현상은 컴퓨터 전공자이면 필수적으로 배워야 하는 프로그래밍언어 교과목에서 잘 나타나고 있으며, 영어, 문제해결력, 기초 학력의 부재로 학생들에게는 다른 과목에 비해 전공 학습 부진 현상이 심각히 나타나고 있는 현실이다. 프로그래밍 입문과정에서 교육용프로그래밍 언어인 스크래치를 도입하고 활용함으로써 학생들의 창의력, 문제해결력 신장과 전공학습에서의 부진현상을 줄이고 수업의 만족도를 높이기 위한 연구를 진행하고 이를 통해 수업의 효과를 분석하였다.

스크래치가 학습자들의 동기와 문제해결력 향상에 긍정적인 영향을 미친다는 연구결과는 이미 나와 있다[7]. 스크래치를 도입한 반은 <표 2>에서 제시한 수업모형과 같이 C언어에서 나오는 개념을 스크래치를 활용하여 실습하고 다시 스크래치에서 작성한 코드를 C언어로 변환하는 과정을 통해 C언어의 문법과 특징, 구조를 익히는 과정을 반복하였다. 연구결과 스크래치를 통해 도움을

받았다는 학생들이 85%이상이 되었고 프로그래밍 입문 교육을 받는 다른 학생들에게도 추천하겠다는 학생들의 수가 28명중 21명이 되었다.

비주얼적인 요소와 쉬운 문법, 쉬운 오류수정 과정을 통해 학생들은 프로그램을 작성하는 것에 대한 자신감이 생겼고, 이는 C언어의 문법을 어렵고 힘들게만 생각하는 학생들에게 수업의 집중도를 높이고 흥미를 일으키는 도구가 되었다. 하지만 스크래치가 쉬운 문법과 비주얼적인 요소로 사용자들에게 친근하게 다가가고 있지만 C언어의 모든 개념과 기능을 설명하지는 못하고 있다. 스크래치를 활용하여 설명할 수 있는 C언어의 개념들은 학생들이 반복적 실습과 쉬운 작성과정을 통해 충분히 이해하고 이를 활용하여 알고리즘을 구상하고 프로그램코드를 작성하는데 충분히 사용되었다. 하지만 좀 더 복잡한 프로그래밍 개념을 설명하기 위해서는 스크래치를 활용하는 것에 대해 좀 더 보완적인 실습이 필요하다고 생각한다. 본 연구는 한 학기동안 두 개 반 학생들을 대상으로 한 실험이므로 앞으로 좀 더 많은 학생들을 대상으로 연구를 확대하여 좀 더 일반적인 자료를 찾아낼 필요가 있다고 생각한다.

참 고 문 헌

- [1] 권두승 (2000). 전문대학 졸업생의 실태 및 산업체 요구조사. 한국 전문대학협의회.
- [2] 김미선·박관우 (2011). 스크래치 언어를 이용한 초등 학교 프로그래밍 교육 과정 연구, 정보교육학회 학술논문집. 2권 1호.
- [3] 김병욱·김한성·이원규 (2010). PBL기반 프로그래밍 수업에서 학습양식에 따른 학습동기 차이분석을 통한 시사점 도출. 한국컴퓨터교육학회 논문지. 13권 5호. 15-27.
- [4] 김수환·한선관·한희섭·김현철 (2011). Computational Literacy 교육에서 다중지능전략 교육방법의 효과. 한국컴퓨터교육학회 논문지. 14권 6호. 11-18.
- [5] 김태희·강문설 (2010). 레고마인드 스톨을 이용한 프로그래밍 입문 교육의 효과 측정. 한국 인터넷 정보학회. 11권 4호. 159-173.
- [6] 박성미 (2009). 전문대학생을 위한 학습전략 진단 도구의 개발. 수산해양교육연구. 21(1). 16-27.
- [7] 송정범·조성환·이태욱 (2008). 스크래치 프로그래밍

학습이 학습자의 동기와 문제해결력에 미치는 영향. 한국정보교육학회. 12권 3호. 323-332.

- [8] 안유정·김경아 (2010). 심화·보충형 수준별 학습에 의한 컴퓨터 프로그래밍 영역별 학습 효과 분석. 한국 컴퓨터정보학회 논문집. 137-144.
- [9] 이영준·이은경 (2011). 스크래치 프로그래밍. (주)교학사
- [10] 조성환·송정범·김성식·백성혜(2008). 스크래치를 이용한 프로그래밍 수업효과. 한국정보교육학회. 12권 4호. 375-384.
- [11] 조성환·송정범·김성식·이경화 (2007). CPS에 기반한 스크래치 EPL이 문제해결력과 프로그래밍 태도에 미치는 효과. 한국정보교육학회. 12권 1호. 77-88.
- [12] 채유진 (2006), 컴퓨팅 교육을 위한 교육용 프로그래밍언어 두리틀, 스크의 비교분석, 고려대학교 교육대학원 석사학위 논문.
- [13] 최현중 (2010). 대학 프로그래밍 강좌를 위한 프로그래밍 교육 프레임워크. 한국컴퓨터교육학회 논문지. 14권 1호. 69-79.
- [14] Amanda Wilson·David C.Moffat (2010). Evaluating Ssrcatch to introduce younger schoolchildren to programming. Proceedings of the 22nd Annual Workshop of the Psychology of Programming Interest Group.
- [15] Caitlin Kelleher·Randy Pausch (2003). Lowering the Barriers ro Programming: a survey of programming environments and languages for novice programmers. ACM.
- [16] David J.Malan·Henry H.Leitner (2007). Scratch for Budding Computer Scientists. SIGCSE' 07. March 7-10.
- [17] Scratch Homepage. <http://scratch.mit.edu>

박정신



미디어 CAI 개발팀

- 1987년 2월 : 연세대학교 전산과학과(이학사)
- 1989년 2월 : 연세대학교 대학원 전산전공(이학석사)
- 1995년 12월 : University of Maryland, college park 박사과정 수료
- 1989년 4월 ~ 1990년 2월 : (주)웅진
- 1997년 9월 ~ 1998년 2월 : (주)버추얼텍 개발팀
- 1998년 2월 ~ 2008년 2월 : 우송공업대학 컴퓨터정보계열
- 2008년 3월 ~ 현재 : 우송정보대학 컴퓨터정보과 교수
- 관심분야 : 프로그래밍 교육, 모바일 소프트웨어, 소프트웨어 엔지니어링
- E-Mail: jspark@wst.ac.kr

조석봉



- 1986년 2월 : 충남대학교 계산통계학과(이학사)
- 1988년 8월 : 중앙대학교 전자계산학과(이학석사)
- 1998년 8월 : 아주대학교 컴퓨터공학과(공학박사)
- 1989년 3월 ~ 현재 : 우송정보대학 컴퓨터정보과 교수
- 관심분야 : 병렬처리, 데이터베이스, 컴퓨터교육
- E-Mail : space012@hanmail.net