



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

석사학위논문

마이크로비트를 활용한 디자인 사고 중심  
SW교육에 대한 연구

A Study on the Design Thinking-based  
SW Education using micro:bit

제주대학교 교육대학원

초등컴퓨터교육전공

박 서 영

2018년 8월

마이크로비트를 활용한 디자인 사고 중심  
SW교육에 대한 연구

A Study on the Design Thinking-based  
SW Education using micro:bit

지도교수 김 종 훈

이 논문을 교육학 석사학위 논문으로 제출함

제주대학교 교육대학원

초등컴퓨터교육전공

박 서 영

2018년 8월

박 서 영의  
교육학 석사학위 논문을 인준함

심사위원장 박 남 제



심사위원 김 성 백



심사위원 김 중 훈



제주대학교 교육대학원

2018년 6월



# 목 차

국문 초록 .....	iv
<b>I. 서론</b> .....	1
1. 연구의 필요성 .....	1
2. 연구 방향 .....	3
<b>II. 이론적 배경</b> .....	4
1. SW(소프트웨어)교육 .....	4
2. 마이크로비트 .....	5
3. 디자인사고 .....	7
4. 창의성 .....	9
5. 실물교육 .....	11
<b>III. 요구조사</b> .....	13
1. 설문 조사 대상 .....	13
2. 설문 내용 .....	13
3. 설문 결과 .....	16
4. 요구 분석 의사 결정 .....	20
<b>IV. 마이크로비트 SW교육 프로그램 설계 및 개발</b> .....	22
1. 연구 방법 .....	22
가. 교육 프로그램 개발 .....	22
나. 교육 프로그램 적용 .....	25
1) 연구 대상 .....	25
2) 연구 방법 .....	25
3) 창의성 분석 방법 .....	28
2. 연구 결과 .....	29
가. 창의성 검사 정규성 검정 .....	29
나. 창의성 집단 내 비교 .....	29
3. 연구 결과 분석 .....	31
<b>V. 결론 및 제언</b> .....	33
참고 문헌 .....	35
ABSTRACT .....	37
부 록 .....	38

## 표 목 차

〈표 II-1〉 IDEO 디자인 사고 프로세스 단계 .....	9
〈표 III-1〉 초등학생 설문 참여자 학년 분포 .....	13
〈표 III-2〉 초등학생 설문 참여자 남녀 분포 .....	13
〈표 III-3〉 컴퓨터 활용 수업에서 주로 하는 활동 .....	16
〈표 III-4〉 SW교육 경험의 유무 .....	16
〈표 III-5〉 경험한 SW교육의 종류 .....	17
〈표 III-6〉 SW교육에 대한 관심도 .....	17
〈표 III-7〉 SW교육 참여시 선호하는 학습 형태 .....	17
〈표 III-8〉 SW교육 참여시 선호하는 학습 내용 .....	18
〈표 III-9〉 선호하는 프로그래밍 결과 확인 방법 .....	18
〈표 III-10〉 ‘휴대 가능한 기기에 프로그램을 넣어 직접 실행해 본다.’는 응답을 한 이유 .....	19
〈표 III-11〉 프로그래밍 실현 기기에 넣고 싶은 기능 .....	19
〈표 III-12〉 SW교육을 통해 얻을 수 있는 가장 큰 장점 .....	20
〈표 IV-1〉 교육 프로그램 내용 설계표 .....	22
〈표 IV-2〉 디자인 사고 단계별 미션지 .....	24
〈표 IV-3〉 지원자 성별과 학년 분포 .....	25
〈표 IV-4〉 교수-학습 과정안 .....	26
〈표 IV-5〉 창의성 검사 실험 설계 .....	28
〈표 IV-6〉 창의성 하위 요소에 대한 정규성 검정 결과 .....	29
〈표 IV-7〉 정규분포를 이룬 하위요소에 대한 사전·사후 검사 결과 비교(대응표본 t검증) .....	30
〈표 IV-8〉 정규분포를 이루지 않는 하위요소에 대한 사전·사후 검사 결과 비교(Wilcoxon 부호 순위 검정) .....	30

## 그림 목 차

[그림 II-1] 마이크로비트 .....	5
[그림 II-2] 마이크로비트 웹브라우저 코딩 화면 .....	6
[그림 III-1] 요구조사 설문지 내용 .....	15

# 국 문 초 록

## 마이크로비트를 활용한 디자인 사고 중심 SW교육에 대한 연구

### 박 서 영

#### 제주대학교 교육대학원 초등컴퓨터교육전공 지도교수 김 종 훈

창의성을 기반으로 하는 디자인은 창의성 신장에 있어서 효과적임이 지금까지 여러 연구를 통해 밝혀졌다. 본 연구의 목적은 마이크로비트를 활용한 디자인 사고 중심 SW(소프트웨어)교육이 초등학생의 창의성에 미치는 효과를 규명하는데 있다. 이를 위해 무작위로 선별된 제주에 거주하는 초등학생 4, 5, 6학년을 대상으로 SW교육에 관한 사전요구조사를 실시하였으며, 이를 반영하여 마이크로비트를 활용한 디자인 사고 중심의 SW교육 프로그램 및 교재를 개발하였다. 이렇게 개발한 교육 프로그램을 이용하여 제주도 내에서 선착순 신청에 의해 선발된 4, 5, 6학년 초등학생 25명을 대상으로 총 48차시에 걸쳐 교육을 실시하였다. 창의성 신장 여부를 알아보기 위해 사전과 사후로 나누어 TTCT검사를 하였고 수집된 자료는 대응표본 t-검정을 통해 분석하였다. 그 결과 본 연구에서 개발한 교육프로그램은 초등학생의 창의성 신장에 효과적인 것으로 나타났다.

주요용어 : 창의성, 디자인 사고, 마이크로비트, SW교육

# I. 서 론

## 1. 연구의 필요성

제4차 산업혁명과 SW 중심 사회는 이제 더 이상 먼 미래의 이야기가 아니다. 첨단 정보통신기술이 사회와 경제 전반에 융합되어 혁신적인 변화를 일으키며 빠른 변화를 가져오고 있는 지금 전 세계는 제4차 산업혁명에 주목하고 있다. 제4차 산업혁명이란 컴퓨터와 인터넷을 기반으로 했던 기존의 지식정보 사회에서 더 나아가 제품과 서비스를 네트워크로 연결하고 사물을 지능화 하는 지능정보의 혁명을 말한다. 그리고 이러한 4차 산업혁명 기술의 중심에는 SW가 있다.(김다운, 2017)

기계와 제품이 지능을 가지는 인공지능(AI)을 가능하게 한 것은 방대한 데이터를 체계적이고 효율적으로 분석할 수 있게 한 SW기술의 발전에 있었다. 빠르고 정확한 데이터 분석으로 필요한 정보를 제공하며 학습을 통한 예측 능력까지 갖춘 SW기술은 그 가능성이 무궁무진하다. 이는 산업의 발전을 가져오는 것은 물론 과학과 사회과학, 다양한 인문학 분야에까지 적용돼 다가오는 미래를 예측하고 대비할 수 있게 한다.

이러한 흐름에 발 맞춰 세계적인 기업들도 SW기술을 선점하려 노력하고 있다. 메르세데스 벤츠 회장은 “자동차는 이제 가솔린이 아니라 SW로 움직인다”라는 말을 통해 앞으로의 기업 경쟁력이 SW기술에 있음을 강조했다.(중앙일보, 2015) 벤츠사뿐만 아니라 유통, 생산, 제조, 서비스 등 전 분야에 걸친 기업들도 앞 다투어 SW기술 개발에 나서고 있다. 과거 SW기술 개발이 IT기업에 국한되어있었다는 점을 비교해보면 이는 큰 변화임에 틀림없다. 이렇듯 기업의 주력 산업이라는 경계가 무너지고 SW기술에 대한 중요성이 증대되고 있는 제4차 산업혁명의 물결은 이제 SW기술이 선택이 아닌 필수임을 시사한다.

전 세계와 더불어 우리 사회는 SW가 성장과 혁신, 가치 창출의 중심이 되고 개인과 기업, 국가 경쟁력을 좌우하는 시대로 넘어가고 있다. 이러한 흐름에 맞춰 미국, 일본, 영국 등 선진국에서는 국가 경쟁력을 위한 핵심 과제로 이미 SW교육을 선정하여 실시해 오고 있다. 우리나라는 2018년도부터 SW교육의

무화를 시행중에 있다. 초등학교에서는 5~6학년 때 ICT 단원을 12시간 배우던 방식이 17시간 이상의 교육으로 늘어났으며 중학교에서는 선택과목이었던 '정보' 과목이 필수과목으로 전환되어 34시간의 코딩 의무교육이 진행되고 있다. 고등학교에서는 심화선택과목이었던 '정보'가 일반 선택과목으로 바뀌어 선택의 기회가 확대되었다. 교육부는 이러한 SW교육의 필수화가 안착되도록 2018년까지 초등교사 및 중등교사를 대상으로 한 심화연수를 추진하고 있으며 SW교육모델을 확산하기 위한 SW 선도학교를 운영 중이다.(교육부, 2016) 이렇듯 SW교육에 대한 논의가 계속되는 가운데 우리나라의 SW교육이 초기 단계인 만큼 교육에 효과적인 프로그래밍 언어 및 교육 내용, 교수 방법에 대한 많은 논의와 연구가 필요한 상황이다.

본 연구에서는 학생들의 효과적인 SW교육과 그로 인한 창의성 향상을 목적으로 하는 교육 프로그램을 개발하기 위해 프로그래밍 언어이자 도구인 마이크로비트를 사용하였다. 마이크로비트란 영국과 영국의 국영 방송사 BBC가 합작 투자해 만든 휴대용 프로그래밍 디바이스로, 포켓 크기의 코딩이 가능한 컴퓨팅 장치이다. 마이크로비트는 디스플레이, 버튼, 움직임 감지, 온도 및 빛 감지 기능이 내장되어 있으며 블루투스 기능을 지원한다. 또 현재 널리 쓰이는 프로그래밍 언어인 스크래치와 같이 블록 쌓기 형식으로 코딩이 가능하기 때문에 초보자들도 쉽게 프로그래밍을 실현할 수 있다. 마이크로비트 웹 사이트를 통해 공유가 쉬우며 마이크로비트 기기 구입비용이 저렴하다는 장점도 있다.(Sue Sentance, 2017) 무엇보다도 프로그래밍의 과정과 결과를 손 안에서 직접 구현해볼 수 있다는 점에서 실제적인 경험을 주어 학생들로 하여금 코딩 활동에 매력을 느끼게 하고 바람직한 동기를 갖게 할 수 있다. 이처럼 학생들이 실물을 통한 코딩을 경험하며 프로그래밍의 과정을 익힌다면 창의성 신장, 프로그래밍에 대한 흥미향상에 효과가 있을 것으로 여겨진다. 하지만 아직까지 마이크로비트를 활용한 교육 프로그램에 대한 국내 연구는 없었다. 이에 본 저자는 마이크로비트를 활용한 SW교육 연구를 계획하였으며 이를 통한 학생들의 창의성 신장 여부를 알아보고자 하였다.

본 연구가 SW교육으로 인한 창의성 신장을 목적으로 하는 만큼 창의성에 긍정적인 영향을 미치는 것으로 연구되어 온 디자인 사고를 SW교육에 적용하고

자 하였다. 디자인 사고란 집단적 공감과 협력을 통해 창의적인 문제 해결을 목표로 하는 사고 과정으로 디자인 분야에서 시작된 개념이다. 본 연구에서는 팀브라운(Tim Brown, 2008)이 제시한 시작하기, 발견하기, 해석하기, 아이디어내기, 실험하기, 발전시키기로 구성된 디자인 사고과정을 단계별로 수행할 수 있도록 교육 프로그램을 구성하였으며 협업을 통한 아이디어 공유가 활발히 일어날 수 있도록 과제를 제시하였다. 이러한 디자인 사고를 통해 마이크로비트를 활용한 SW교육이 학생들의 창의성 향상을 더욱 더 극대화시킬 것으로 보고, 그 효과를 검증해 보고자 하였다.

## 2. 연구 방향

본 연구는 크게 사전요구조사 - 요구분석 의사 결정 - 프로그램 개발 - 사전검사 - 적용 - 사후검사 - 평가 단계로 이루어진다.

사전요구조사 단계에서는 설문지를 이용해 학생들이 가지고 있는 기존의 SW 교육에 대한 인식 및 요구를 묻는 설문을 실시한다. 요구분석 의사 결정 단계에서는 설문 결과를 바탕으로 SW교육에 대한 학생들의 인식 및 교육 경험, SW에 대한 학생들의 요구, 마이크로비트에 대한 인식 등을 분석하여 학생들에게 적용할 SW교육 프로그램의 내용과 방향, 교수방법 등을 결정한다. 프로그램 개발 단계에서는 요구분석 의사 결정 단계에서 결정된 내용을 적용하여 마이크로비트를 활용한 디자인 사고 중심 SW교육 프로그램 및 교재를 개발한다. 사전검사에서는 프로그램 적용 전 창의성 측정 검사지인 TTCT를 투입하여 학생들의 창의성을 측정한다. 적용 단계에서는 지원한 초등학교 4~6학년 학생들을 대상으로 본 연구에서 개발한 SW교육 프로그램을 총 48차시에 걸쳐 교육한다. 사후검사 단계에서는 교육이 끝난 학생들을 대상으로 TTCT 창의성 검사를 시행한다. 평가 단계에서는 사전, 사후 검사 결과를 바탕으로 본 연구의 교육으로 인한 학생들의 창의성 향상 유무를 살펴본다.

## II. 이론적 배경

### 1. SW(소프트웨어)교육

SW교육이란 디지털 기기를 활용한 SW 활용 능력을 함양하고 우리 주변에서 일어나는 다양한 문제를 보다 더 효율적이고 창의적으로 해결할 수 있는 컴퓨팅 사고력을 기르는 것이다. 이는 기존의 정보통신기술 교육과 같이 각종 응용프로그램의 단순 활용방법이나 컴퓨터 이론만을 배우는 것과는 차이가 있다.(김나연, 2016)

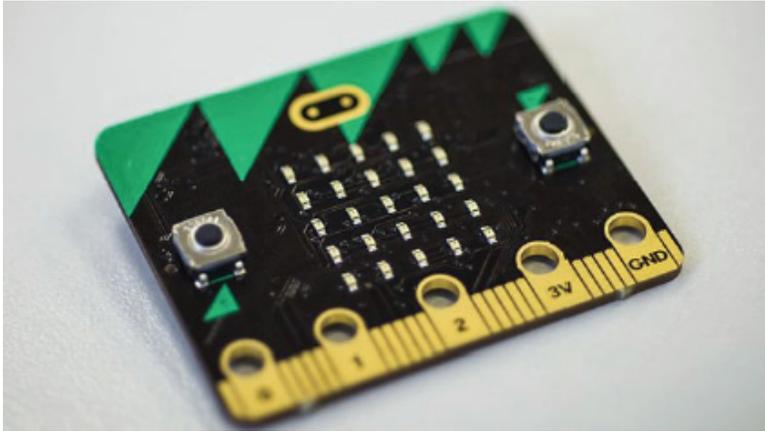
SW교육은 미래 사회의 핵심적 사고 과정이라 할 수 있는 컴퓨팅 사고력을 강조한다. 컴퓨팅 사고력이란 컴퓨터 과학의 기본 원리와 개념을 활용하여 다양한 문제를 이해하고 창의적인 해결방법을 찾아 적용할 수 있는 능력이다.(교육부, 2017) 이는 SW 관련 분야뿐만 아니라 산업, 의학, 경제 등 사회 전반에 걸쳐 활용될 수 있기 때문에 그 중요성은 더 증대되고 있다.(한국교육학술정보원, 2017) 이에 전 세계는 SW교육에 대한 중요성을 인식하고 이미 효과적인 교육을 위해 노력하고 있는 중이다.

우리나라도 이러한 흐름에 맞춰 2018년부터는 SW교육의 의무화를 시행중에 있다. 교육부는 SW교육을 '학습자들이 미래 사회에서 살아가는데 필요한 컴퓨팅 사고력을 기반으로 문제를 해결하는 역량을 기르는 교육'이라 정의하고 초, 중, 고 각급에 맞는 SW교육 정책을 발표하였다.(교육부, 2015) 초등학교에서는 SW교육이 정규 교과는 아니지만 실과 교과에 편입되어 17시간 이상의 교육을 받도록 하며 현재 재학 중인 중학생들은 기존의 선택과목이었던 '정보'과목이 필수 과목으로 전환되어 34시간의 코딩의무교육을 받게 된다. 고등학교에서는 심화선택과목인 '정보'가 일반 선택과목으로 바뀌어 SW교육에 이전보다 무게를 실고 있다. 또한 교육부는 SW교육의 안착화를 위해 교사들을 대상으로 한 다양한 연수를 실시하고 있으며 SW교육모델을 확산하기 위한 SW 선도학교를 운영하고 있다.(교육부, 2016)

그러나 우리나라의 SW교육 시행이 초기 단계이며 효과적인 SW교육 방법 및 도구에 대한 연구가 적은 만큼 미래 사회를 선도할 수 있는 인재를 기르기 위

해서는 SW교육에 대한 다양한 연구가 진행되어야 하겠다.

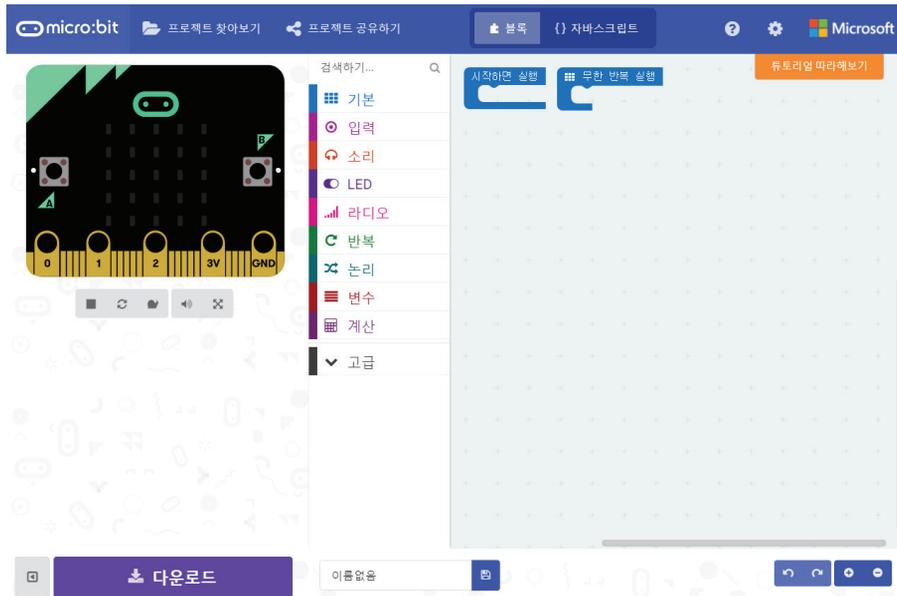
## 2. 마이크로비트



[그림 II-1] 마이크로비트

마이크로비트[그림 II-1](micro:bit 웹사이트)는 영국과 영국의 국영방송사 BBC가 효과적인 SW교육을 위해 합작 개발한 피지컬 컴퓨팅 장치(device)로, 하드웨어의 기능과 SW의 기능을 동시에 갖춘 작은 컴퓨터와 같다.

하드웨어로서의 마이크로비트는 기기 자체적으로 다양한 기능들을 가지고 있다. 마이크로비트 전판에 있는 25개의 LED램프는 디스플레이 표현을 가능하게 하며 양 옆에 위치한 두 개의 버튼은 사용자의 입력을 받을 수 있다. 또 가속도 센서, 나침반(자기)센서, 빛 센서, 온도 센서가 내장되어 있어 움직임 감지, 온도 측정 및 빛의 양을 측정 할 수 있다. 이 외에도 라디오 기능, 블루투스를 이용한 무선 통신 기능이 있으며 USB인터페이스를 지원하고 하드웨어 확장 핀으로 모니터, 전기 부품 등 원하는 다른 기기와의 연결이 가능하다.(T. Ball, 2016) 이러한 기능들은 프로그래밍을 할 때 다양한 소스로 활용될 수 있으며 학습자로 하여금 하드웨어와 SW간의 관계 이해에 도움을 줄 수 있을 것으로 생각된다.



[그림 II-2] 마이크로비트 웹브라우저 코딩 화면

마이크로비트는 특별한 SW 설치 없이도 마이크로비트 웹브라우저[그림 II-2](micro:bit 웹사이트)를 통해 프로그래밍이 가능하다. 즉 데스크톱, 노트북, 스마트패드 등의 기기와 인터넷이 가능한 환경이 갖춰진다면 언제 어디서든 코딩을 할 수 있다. 코딩 방식은 블록 쌓기 형식을 기본으로 하며, 이는 코딩 과정을 시각화하여 프로그래밍 과정을 파악하기 쉽게 하고 컴퓨터 언어의 순서도 개념의 이해를 도와 코딩을 처음 접하는 학습자도 프로그래밍을 쉽게 익힐 수 있다는 장점을 갖는다. 마이크로비트는 이러한 블록 형태뿐만 아니라 전문적인 컴퓨터 언어인 파이썬, 자바스크립트와의 연동도 가능하여 블록형태로 된 연습에서 더 나아가 실제로 어떤 구조로 코딩이 이루어지는지에 대해 알 수 있다. 또한 기초적인 코딩은 물론, 난이도가 높은 코딩도 가능하여 초보 학습자부터 심화학습을 원하는 학습자까지 폭 넓게 이용할 수 있다. 마이크로비트 웹 사이트는 최근부터 한국어를 지원을 하고 있어 학생들이 더욱 더 쉽게 이용할 수 있다.

웹브라우저를 이용한 프로그래밍 환경은 공유가 쉽다는 장점도 있다. 실제로 본 연구에서는 웹 내에 학생들이 만든 결과물을 공유할 수 있는 공간을 만들어

피드백 전달 및 과제 확인을 더욱 효율적으로 할 수 있었다. 또 마이크로비트 웹브라우저에서는 전 세계의 마이크로비트 학습자 및 연구자들이 결과물을 올리는 공유 공간이 있어 학습자들이 코딩 아이디어를 제공받기도, 제공할 수도 있어 더욱 더 발전적인 코딩 학습을 가능하게 한다.(Sue Sentance, 2017)

이러한 하드웨어의 기능과 SW의 기능을 동시에 가지고 있는 마이크로비트의 가장 큰 장점은 실물을 통한 교육이라는 것이다. 마이크로비트는 실제로 작동 가능한 기기에 내가 만든 프로그램을 넣을 수 있으며 이를 손 안에서 직접 확인해 볼 수 있다. 이는 학습자로 하여금 프로그래밍 결과에 대한 직관적 이해를 돕고 학습자가 느낄 수 있는 학습과 실제의 괴리를 줄여준다. 또한 소유하고 휴대할 수 있다는 점이 코딩 학습에 흥미를 불러일으키는 요소로 작용하여 SW학습에 대한 동기를 쉽게 유발할 수 있으며 학생들이 코딩을 친숙하게 느낄 수 있다.(Sue Sentance, 2017) 또한 마이크로비트는 다른 SW교육 기기에 비해 값이 싸 학생 개개인에게 보급이 가능하다. 이는 학교 또는 국가가 SW교육 도구를 선정하는데 있어 교육비 지출 부담을 줄일 수 있다.

영국 정부는 2015년 9월부터 11살이 되는 모든 영국 학생들에게 마이크로비트를 무료로 배포하고 이를 이용한 SW교육을 실시해오고 있다. 그리고 영국뿐만 아니라 핀란드, 아이슬란드, 싱가포르, 스리랑카까지 전 세계 학교에서 폭넓게 사용 중이다.(The Guardian, 2015) 본 연구에서는 전 세계적으로 SW교육에 활용되고, 효과적 교육 도구로서 자리 잡고 있는 마이크로비트를 활용하여 학생들을 대상으로 한 SW교육 활동을 통해 그 효과를 검증해 보고자 하였다.

### 3. 디자인 사고

디자인 사고란 집단적 공감과 협력을 통해 창의적인 문제 해결을 목표로 하는 사고 과정으로, 디자인 분야에서 시작된 개념이다.(Martin, 2009) 디자인 사고는 집단이 함께 새로운 아이디어를 창출하는데 필요한 여러 가지 사고의 과정을 거치게 되는데, 이 사고과정에는 문제를 해결함에 있어서 필요한 것은 무

엇인지, 가능한 방법은 무엇인지, 성공적인 방법은 무엇인지 등의 여러 가지 사고 요소가 존재한다. 다시 말해 디자인 사고란 인간의 사고 방법이며 관찰과 공감, 협업을 통해 아이디어에 대한 영감을 얻고 문제점을 해결해나가는 과정이라 볼 수 있는 것이다.(이선영, 2016)

디자인 사고는 디자인 분야뿐만 아니라 효과적인 아이디어 창출과 문제해결이 필요한 곳이라면 어디든 적용될 수 있다. 이에 따라 디자인 사고에 대한 연구는 활발히 이루어져 왔다. 디자인 사고의 정의와 개념은 학자마다 조금씩 다른 견해를 갖는다. 그럼에도 불구하고 디자인 사고는 크게 두 가지 방향으로 그 개념이 정의되는데, 디자인 사고를 직관적 사고에 의한 창조성과 분석적 사고에 의한 완벽한 숙련의 상호작용으로 보는 관점과 실제 디자이너들의 사고방식을 적용해보는 활동으로 보는 관점이 그것이다.(이선영, 2016)

본 연구에서 적용한 디자인 사고는 후자의 개념으로, 이는 디자인 사고의 대표적 학자인 팀브라운(Tim Brown, 2008)에 의해 연구되었다. 그는 디자인 사고가 ‘디자이너처럼 생각한다’는 것을 의미하며 디자이너는 사용자의 욕구와 실현 가능한 기술을 조화시키기 위해 디자이너적 감각과 방법을 사용한다는 점을 강조하였다. 그리고 이러한 사고가 익숙하지 않은 사람들에게 디자인 사고는 광범위한 문제를 해결하는데 있어서 효과적인 역할을 할 것으로 보았다. 또한 디자인 사고를 더 창조적이고 혁신적인 방법으로 문제를 해결해 가기 위한 사고 방법이며, 상황에 맞는 해결책을 분석하여 그에 맞는 합리적인 사고 과정을 찾는 과정이라고 간주하였다.

팀브라운은 데이빗 켈리와 함께 아이데오(IDEO)를 창업하여 더 구체적인 디자인 사고 과정을 제시하였다.(IDEO, 2012) 여섯 단계로 이루어진 디자인 사고 프로세스는 다음의 <표 II-1>과 같다.

〈표 II-1〉 IDEO 디자인 사고 프로세스 단계

단계	정의
시작하기 (Start)	<ul style="list-style-type: none"> <li>· 챌린지 정의하기(define a challenge)</li> <li>· 프로젝트 계획 세우기(create a project plan)</li> </ul>
발견하기 (Discovery)	<ul style="list-style-type: none"> <li>· 디자인챌린지 이해하기(understand the challenge)</li> <li>· 조사준비하기(prepare research)</li> <li>· 영감수집하기(gather inspiration)</li> </ul>
해석하기 (Interpretation)	<ul style="list-style-type: none"> <li>· 이야기해보기(tell stories)</li> <li>· 의미 찾아보기(search for meaning)</li> <li>· 기회포착하기(frame opportunities)</li> </ul>
아이디어내기 (Ideation)	<ul style="list-style-type: none"> <li>· 아이디어 만들기(generate ideas)</li> <li>· 아이디어 다듬기(refine ideas)</li> </ul>
실험하기 (Experimentation)	<ul style="list-style-type: none"> <li>· 프로토타입 만들기(make prototype)</li> <li>· 피드백 받기(get feedback)</li> </ul>
발전시키기 (Evolution)	<ul style="list-style-type: none"> <li>· 평가해보기(track learnings)</li> <li>· 다음 계획 세우기(move forward)</li> </ul>

본 연구에서는 위와 같은 팀 브라운의 디자인 사고 과정을 수업에 적용하여 학생들이 프로그래밍을 할 때 더욱 더 효과적인 아이디어 창출과 창의적인 문제해결을 할 수 있도록 하였다. 또한 SW교육과 디자인 사고의 결합이 학생들의 창의성을 더욱 더 향상시켜 줄 것이라 가정하고 이를 검증하고자 하였다.

#### 4. 창의성

창의성은 급변하는 현대사회에서 필요도가 어느 때보다 증가하고 있으며, 이러한 창의성 교육에 대한 관심도 점차 높아지는 실정이다. 일반적으로 창의성은 ‘새로운 아이디어를 만들어 낼 수 있는 능력’이며, 이는 어떠한 문제를 인식하고 이를 해결하는 과정에서 보편적이지 않은 새로운 방식을 떠올리고 채택할 수 있는 일련의 과정에 작용한다고 볼 수 있다.

창의성의 개념은 오래전부터 연구되어 왔으며, 학자들의 연구에 따라서 다양하게 정의되었다. Torrance(1972)는 난해한 문제의 인식과 이를 해결하기 위한 절차 설계, 그리고 문제해결의 결과를 도출하는 일련의 과정들을 창의성이라 정의하였다. Hennesey(1988)는 창의성을 새롭고 적절한 것을 생성해 낼 수 있는 개인의 능력으로 정의하였다. 이경화(2002)는 창의성을 '특정 시대상'에서 '개인적 환경'에 의해 길러진 '새로운 산물을 생성해 낼 수 있는 능력'으로 정의 하였으며, 특히 개인을 둘러싼 사회, 문화적 환경의 중요성을 강조하였다. 이처럼 창의성은 연구자마다 다양하게 정의되었으며, 이는 창의성을 보는 관점이 과정(process), 사람(person), 산물(product), 환경(press) 등으로 다양하고(Rhodes, 1961) 연구자가 중요시하는 관심에 따라 달라진다는 것을 보여준다.

창의성을 구성하는 요소 또한 학자에 따라 다양하게 제시되어왔다. Guilford(1978)는 인간의 사고를 수렴적 사고와 확산적 사고로 나누었을 때 창의성은 확산적 사고에 해당한다고 주장했고, 이를 이루는 구성요소로 유창성, 융통성, 정교성, 독창성, 민감성 등을 제시하였다. Sternberg(1996)는 지적능력, 지식, 사고유형, 성격, 동기 및 환경을 제시하며 이와 같은 요소가 서로 상호작용을 할 때 창의성이 발휘된다고 주장했다. Torrance(1972)는 창의성의 인지적 요소로 유창성, 융통성, 독창성, 정교성을 주장하였으며 이를 검사하기 위한 도구로 TTCT를 제시했고, 이는 본 연구에서 창의성 측정에 사용되었다. 국내 연구를 보면 정원식·이영덕(1993)은 창의성의 구성요소로 유창성, 융통성, 독창성, 개방성을, 전경원(2000)은 유창성, 융통성, 독창성, 상상력을 제시하였다. 이와 같이 창의성의 구성요소에 대해서도 연구자들 간에 다양한 견해가 존재하며, 이는 창의성을 이루는데 수많은 요소들이 복합적으로 작용한다는 것을 보여준다. 비교적 최근 연구에서, Davis(2005)는 유창성, 융통성, 정교성, 민감성, 독창성, 문제 규명, 시각화, 상상력, 은유적 사고, 분석력, 종합력, 평가력, 의미 확장 등의 구성요소를 제시하며 창의성에 기여하는 지적인 능력이 매우 다양하다고 하였다.

## 5. 실물교육

교육이란 어떠한 지식을 전달하는 일로 정의할 수 있다. 그리고 이러한 교육에서, '어떠한 지식을 전달할 것인가'와 비견될 만큼 중요한 문제가 '어떻게 전달할 것인가'이다. 교육은 전통적으로 언어를 통해서 행해졌다. 이는 동서고금을 막론하고 수많은 증거로부터 쉽게 인식할 수 있으며, 스승의 육성을 통한 교육이 가장 보편적인 교육이었음은 널리 알려져 있다(박진주, 2008).

하지만 언어적 전달 외에도, 다른 효과적인 교육방법에 대한 탐구는 계속되어 왔다. 비교적 최근에 들어서는 실물을 이용한 교육이 주목받고 있다. 아동의 입장에서는 언어적인 영역은 계속 발전하는 중으로서, 다르게 말하면 아직 완전히 발달하지 않은 상태이며 개인차도 심하다. 그러므로 언어적인 전달을 통한 교육만으로는 교사들이 전달하고자 하는 바를 충분히 전하지 못할 가능성이 있다. 실물을 통한 교육은 아동이나 학생들에게 훨씬 친숙한 도구를 통하여 추상적인 개념을 설명할 수 있으며, 다양한 감각의 자극을 통해 또 다른 효과를 기대할 수도 있다.

실물교육의 효과에 대한 교육 이론적 근거는 듀이에 의해 크게 발전했다. 듀이는 실용주의 인식론을 기반으로 하여 지식을 추상적이기 보다 실제적인 개념으로 인지하였다. 그러므로 지식이란 사물 내에 있는 것이 아니라 사물을 사용하는 과정에 있으며, 이는 경험을 통해서 획득할 수 있다는 것이다. 아동의 경험이란 자신이 흥미와 관심을 가지는 사물들로 구성되며, 이러한 사물에 대해 가지는 애정과 관심이 아동의 세계를 구성하는 주요 요소이다. 교과과정을 이루는 내용은 이러한 직접적 경험으로부터 추상화된 것이며, 그러므로 실물교육과 같이 실제의 생생한 경험을 통해서 언어적 교육의 한계를 극복할 수 있다고 하였다(Dewey, 1903).

브루너는 지식 전달의 방법론으로써 작동적 표현, 영상적 표현, 상징적 표현의 세 가지를 제시하였다. 단순히 생각하면 작동적 표현은 실제적 체험을, 영상적 표현은 쉽게 시각화된 자료를 통한 교육을, 상징적 표현은 언어적 교육을 대표한다고 볼 수 있으며, 각각의 전달 방법이 같은 내용을 전달하는데 모두 도움이 된다는 것이다(Bruner, 1960). 브루너는 듀이와는 이면적으로는 상이

한 이론을 제시하였으나, 표면적으로 이러한 실물교육이 전통적 교육의 단점을 보완하는데 도움이 된다는 공통점을 보였다.

저자는 이러한 인식에 근거하여 교육에 마이크로비트를 사용하였다. 위에서 설명한 바대로, 언어적 전달과 더불어 이를 실물화 시켜 직접 경험하게 하여, 언어적인 전달만으로는 부족할 수 있었던 부분을 보완할 수 있는 실물교육의 효과를 기대하였다(Dewey, 1903).

### Ⅲ. 요구 조사

#### 1. 설문 조사 대상

프로그램 설계 및 개발에 앞서 학습자들의 요구를 분석하기 위하여 초등학생 4, 5, 6학년 183명을 대상으로 SW 관련 설문을 실시하였다. 설문 참여자의 학년 분포는 <표 Ⅲ-1>과 같으며 설문에 참여한 남녀 비율은 <표 Ⅲ-2>와 같다.

<표 Ⅲ-1> 초등학생 설문 참여자 학년 분포

설문 참여자 (4, 5, 6학년)	학년		
	4	5	6
학생비율(%)	39	31.5	29.5

<표 Ⅲ-2> 초등학생 설문 참여자 남녀 분포

설문 참여자 (4, 5, 6학년)	학년	
	남	녀
학생비율(%)	52.2	47.8

#### 2. 설문 내용

사전 요구 조사를 위한 설문 내용은 크게 두 가지 주제로 나뉜다. 첫 번째는 기존의 SW교육에 대한 인식상태 및 교육 경험 여부에 관한 것이며(1~6번)

두 번째는 지향하는 SW교육 방식 및 마이크로비트와 관련된 것으로(7~9번) 총 아홉 문항이다. 그리고 마지막 문항(10번)은 SW교육을 통해 얻을 수 있는 장점에 대한 학생들의 인식에 관한 것이다. 설문지의 내용은 [그림 III-1]과 같다.

<p>1. 담임 선생님과 컴퓨터시간에 주로 하는 활동은 무엇인가요?</p> <p>① 타자연습                  ② 문서작성(한글, 엑셀, 파워포인트 등)                  ③ 인터넷 검색                  ④ 소프트웨어 교육(엔트리, 스크래치, 앱인벤터 등)                  ⑤ 기타</p>									
<p>2. SW교육을 받아본 경험이 있나요?</p> <p>① 있다.                  ② 없다.</p>									
<p>3. 이전 질문에 '있다'라고 대답했다면 어떤 SW교육을 받은 경험이 있나요?(중복응답 가능)</p> <p>① 교육용 프로그래밍 언어(엔트리, 스크래치, 앱인벤터 등)                  ② 언플러그드(보드게임 등)                  ③ 피지컬컴퓨팅(아두이노, 오조봇, 마이크로비트 등)                  ④ 기타</p>									
<p>4. SW교육에 대한 관심도는 얼마나 되나요?</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20%;">(낮음)1</td> <td style="width: 20%;">2</td> <td style="width: 20%;">3</td> <td style="width: 20%;">4</td> <td style="width: 20%;">5(높음)</td> </tr> </table>					(낮음)1	2	3	4	5(높음)
(낮음)1	2	3	4	5(높음)					
<p>5. SW교육을 받는다면 어떤 방법으로 학습을 하고 싶나요?</p> <p>① 강의·실습형(선생님 강의 후 직접 해보기)                  ② 개인과제형(과제를 스스로 해보기)                  ③ 그룹과제형(친구들과 함께 과제 해결하기)                  ④ 기타</p>									
<p>6. SW교육을 받는다면 어떤 내용의 교육을 받고 싶나요?</p> <p>① 프로그래밍(프로그램 만들기)                  ② 이진법 등의 컴퓨터 언어 배우기                  ③ 언플러그드로 익히는 컴퓨팅 사고력 배우기</p>									

④ 기타
<p>7. 수업 후 프로그램을 만들었을 때 잘 만들었는지 확인하는 방법으로 무엇을 활용하고 싶나요?</p> <p>① 틀린 곳이 없는지 눈으로 확인한다.          ② 선생님께 검사를 받는다.          ③ 휴대 가능한 기기에 프로그램을 넣어 직접 실행시켜 본다.          ④ 친구들과 토의를 한다.          ⑤ 기타</p>
<p>8. 3번(휴대 가능한 기기에 프로그램을 넣어 직접 실행해 본다.)을 고른 이유는 무엇인가요?</p> <p>① 프로그램이 작동하는지 바로 확인해 볼 수 있어서          ② 어디에서나 프로그램을 실행해 볼 수 있어서          ③ 실제로 보이고 들리는 결과가 나타나서          ④ 프로그램의 원리를 더 잘 이해할 수 있을 것 같아서          ⑤ 기타</p>
<p>9. 내가 만든 프로그램을 실행시켜 볼 수 있는 휴대 가능한 기기가 있다면 여기에 어떤 기능을 넣고 싶은가요?(중복응답 가능)</p> <p>① 스피커(소리) 기능          ② 빛과 어둠을 감지하는 센서 기능          ③ LED(불빛), 움직이는 이미지 기능          ④ 컴퓨터 또는 다른 기기와 연결할 수 있는 블루투스 기능          ⑤ 방향키 등의 누를 수 있는 버튼기능          ⑥ 기타</p>
<p>10. SW교육을 통해 얻을 수 있는 가장 큰 장점은 무엇이라고 생각하나요?</p> <p>① 컴퓨터를 활용하여 문제를 효율적으로 해결하는 능력이 좋아질 것 같다.          ② 공부를 잘 하는데 도움이 될 것 같다.(학업성취도 향상)          ③ 실생활 문제를 해결하는 능력이 좋아질 것 같다.(문제해결력 향상)          ④ 새로운 것을 만들어 내는 능력이 생길 것 같다.(창의성 향상)          ⑤ 기타</p>

[그림 Ⅲ-1] 요구조사 설문지 내용

### 3. 설문 결과

학교 컴퓨터시간에 주로 하는 활동은 무엇인지에 대한 항목에는 단순 타자 연습이 19.6%, 문서작성(한글, 엑셀, 파워포인트 등) 24.6%, 인터넷 검색 26.1%, 기타 10.1%를 보였으며 단 19.6%만이 SW교육(스크래치, 엔트리 등)을 받고 있다고 응답했다.

〈표 III-3〉 컴퓨터 활용 수업에서 주로 하는 활동

	타자 연습	문서 작성	인터넷 검색	SW교육	기타
응답 비율(%)	19.6	24.6	26.1	19.6	10.1

학교 또는 외부에서 SW교육 경험이 있는가에 대한 항목에서는 그렇다가 67.4%, 아니다가 32.6%를 보였다.

〈표 III-4〉 SW교육 경험의 유무

	있음	없음
응답 비율(%)	67.4	32.6

SW교육 경험이 있는 67.4%학생들을 대상으로 중복 응답이 가능하게 하여 어떤 종류의 SW교육 경험이 있는지에 대해 물은 답변으로는 교육용 프로그래밍언어(스크래치, 앱인벤터 등)가 74.2%, 언플러그드가 28%, 피지컬컴퓨팅(아두이노, 마이크로비트, 오조봇 등)이 39.8%를 보였다.

〈표 III-5〉 경험한 SW교육의 종류

	교육용 프로그래밍 언어	언플러그드	피지컬컴퓨팅	기타
응답 비율(%)	74.2	28	39.8	12.6

SW교육에 대한 관심의 정도를 1부터 5까지의 숫자로 놓고 고르는 항목에서는 1이 8.7%, 2가 6.5%, 3이 43.5%, 4가 21.7%, 5가 19.6%를 보였다.

〈표 III-6〉 SW교육에 대한 관심도

	1	2	3	4	5
응답 비율(%)	8.7	6.5	43.5	21.7	19.6

SW교육을 받는다면 어떤 방법으로 학습을 하고 싶은지에 대한 문항에는 강의·실습형이 29.7%, 개인과제형 11.6%, 그룹과제형 58%, 나머지는 기타로 응답했다.

〈표 III-7〉 SW교육 참여시 선호하는 학습 형태

	강의·실습형	개인과제형	그룹과제형	기타
응답 비율(%)	29.7	11.6	58	0.1

SW교육을 받는다면 어떤 내용의 교육을 받고 싶은지에 대한 항목에는 컴퓨터 프로그램 또는 기계를 작동시키는 프로그램 만들기가 81.2%, 이진법 등의 컴퓨터언어 배우기가 8%, 언플러그드로 익히는 컴퓨팅 사고력 배우기가 8%를 보였고 나머지는 기타에 머물렀다.

〈표 III-8〉 SW교육 참여시 선호하는 학습 내용

	프로그래밍	컴퓨터언어	언플러그드	기타
응답 비율(%)	81.2	8	8	3

수업 후 프로그램을 만들었을 때 잘 만들었는지 확인하는 방법으로 무엇을 활용하고 싶은지에 대한 문항에는 틀린 곳이 없는지 블록을 확인한다가 13%, 선생님께 검사를 받는다가 38.4%, 가지고 다니며 프로그램을 실행시켜 볼 수 있는 기기로 직접 실행해 본다가 35.5%, 친구들과 토의를 한다가 12.3%, 나머지는 기타 응답하였다.

〈표 III-9〉 선호하는 프로그래밍 결과 확인 방법

	블록 확인	선생님께 확인	친구들과 토의	기기로 직접 실행	기타
응답 비율(%)	13	38.4	35.5	12.3	0.8

7번 문항에서 3번(휴대 가능한 기기에 프로그램을 넣어 직접 실행해 본다.)을 고른 이유가 무엇인지 묻는 문항에는 프로그램이 작동하는지 바로 확인해 볼 수 있어서가 34.7%, 어디에서나 프로그램을 실행해 볼 수 있어서가 18.4%, 실제로 보이고 들리는 결과가 나타나서가 20.4%, 프로그램의 원리를 더 잘 이해할 수 있을 것 같아서가 26.5%를 보였다.

〈표 III-10〉 ‘휴대 가능한 기기에 프로그램을 넣어 직접 실행해 본다.’는 응답을 한 이유

	바로 확인이 가능함	어디에서나 실행이 가능함	실제로 보이고 들리는 결과가 나타남	프로그램 원리 이해가 쉬움
응답 비율(%)	34.7	18.4	20.4	26.5

중복 응답이 가능하게 하여 ‘내가 만든 프로그램을 실행시켜 볼 수 있는 휴대 가능한 기기가 있다면 여기에 어떤 기능을 넣고 싶은가?’에 대한 문항에는 전체 학생의 39.1%가 스피커기능을, 43.5%가 빛과 어둠을 감지하는 센서기능을, 55.8%가 LED로 움직이는 이미지기능을, 52.2%가 컴퓨터 또는 다른 기기와 연결할 수 있는 블루투스기능을, 42%가 방향키 등의 누를 수 있는 버튼 기능이라 답했다.

〈표 III-11〉 프로그래밍 실현 기기에 넣고 싶은 기능

	스피커기능	센서기능	이미지기능 (LED)	블루투스기능	방향키기능
응답 비율(%)	39.1	43.5	55.8	52.2	42

SW교육을 통해 얻을 수 있는 가장 큰 장점이 무엇인지에 대한 문항에는 컴퓨터를 활용하여 문제를 효율적으로 해결하는 능력이 향상될 것이다가 39.1%, 새로운 것을 만들어내는 능력이 향상될 것 같다가 36.2%, 실생활 문제를 해결하는 능력이 향상될 것 같다가 15.2%, 공부를 잘 하는데 도움이 될 것 같다가 9%, 나머지 기타 응답으로는 편리하게 정보를 알 수 있다, 여러 가지 프로그램을 알 수 있다 등이 있었다.

〈표 III-12〉 SW교육을 통해 얻을 수 있는 가장 큰 장점

	컴퓨터 활용 문제 해결력 향상	실생활 문제 해결력 향상	창의성 향상	학업성취도 향상	기타
응답 비율(%)	39.1	15.2	36.2	9	0.5

#### 4. 요구 분석 의사 결정

설문 결과 학교 컴퓨터실 활용 수업에서 SW와 관련된 교육이 거의 이루어지지 않는 것으로 나타났다. 대부분의 응답이 단순 타자연습부터 문서작성, 인터넷 검색 등 기초적인 프로그램 연습 및 사용에 그쳤다. 한편 외부에서 SW교육을 받았던 경험이 있는 학생들은 전체의 67.4%였는데, 그 중 높은 비율이 스크래치, 앱인벤터와 같은 교육용 프로그래밍 언어였으며 본 연구에서 교육하게 될 마이크로비트 활용 피지컬컴퓨팅 교육은 39.8%로 비교적 낮은 비율을 보였다. SW교육에 대한 관심도로는 긍정적 관심도로 볼 수 있는 4, 5의 응답 비율은 41.3%로 SW에 대한 관심이 양호한 것으로 생각된다.

이 결과들을 바탕으로 마이크로비트를 활용한 SW교육을 받을 대부분의 학생들이 기초적 지식부터 습득할 수 있도록 프로그램을 구성하였다.

학생들이 원하는 SW교육 방식에 대한 문항에 의하면 학생들은 친구들과 함께 과제를 해결하는 학습 방식을 가장 선호하였으며 교사의 강의 후 실습, 강의 후 과제를 스스로 해결하는 방식이 그 뒤를 이었다. 이에 따라 강의 후 학생들이 직접 실습을 해 본 다음, 친구들과 함께 과제를 해결하도록 하였으며 이 과정에서 팀 브라운의 디자인 사고 과정을 단계별로 수행할 수 있도록 구성하였다. 또한, 더욱 효과적인 학습을 위해 다음 강의 전까지 개인적으로 디자인 사고를 적용하여 해결하는 과제를 제시하여, 이전에 친구들과 토의하여 진행하던 디자인 사고과정을 개인적으로 다시 시행할 수 있도록 하였다.

마이크로비트에 대한 학생들의 인식은 매우 긍정적인 것으로 나타났다. 원하는 프로그램을 만들어 작은 기기인 마이크로비트에 넣어 직접 작동시켜보는 것

은 학생들의 흥미를 끌 수 있으며 작동 여부가 바로 확인 가능한 점은 학생들로 하여금 바로 피드백을 얻을 수 있다는 장점이 된다. 또한 마이크로비트는 학생들이 원하는 기능들을 고루 갖추고 있다. 설문 결과로 나온 학생들의 선호 기능 중 빛과 어둠을 감지하는 센서기능, LED로 움직이는 이미지기능, 컴퓨터 또는 다른 기기와 연결할 수 있는 블루투스기능, 방향키 등의 누를 수 있는 버튼 기능이 고루 적용될 수 있도록 프로그램을 구성하였다.

SW교육에 대한 경험과 관심이 많지 않지만 학생들은 SW교육에 대해 그 중요성을 충분히 인식하고 효과를 기대하고 있는 것으로 보인다. 컴퓨터를 활용하여 문제를 효율적으로 해결하는 능력이 향상될 것이라는 응답부터 새로운 것을 만들어내는 능력이 향상될 것 같다, 실생활 문제를 해결하는 능력이 향상될 것 같다, 공부를 잘 하는데 도움이 될 것 같다는 응답까지 비율이 골고루 나타났다. 나머지 기타 응답으로는 편리하게 정보를 알 수 있다, 여러 가지 프로그램을 알 수 있다 등으로 전체적으로 SW교육에 대한 긍정적인 인식을 가지고 있음을 알 수 있었다.

## IV. 마이크로비트 SW교육 프로그램 설계 및 개발

### 1. 연구 방법

#### 가. 교육 프로그램 개발

사전 요구 조사 결과 분석을 바탕으로 디자인 중심 마이크로비트를 활용한 SW교육 프로그램의 학습 내용과 교수전략을 결정하였다. 프로그램 내용은 <표 IV-1>과 같다.

<표 IV-1> 교육 프로그램 내용 설계표

차시	일자	주제	프로그램 내용
1-8	8.14(월)	오리엔테이션 및 기초 이미지표현1	<ul style="list-style-type: none"> <li>- 오리엔테이션 및 사전검사지 투입(TTCT)</li> <li>- 마이크로비트 기초기능 익히기</li> <li>- 이미지표현                             <ul style="list-style-type: none"> <li>①LED켜다 끄기 반복하기</li> <li>②LED로 다양한 이미지 만들기</li> <li>③LED이미지 조작하기(버튼, 흔들기 활용)</li> </ul> </li> </ul>
9-16	8.15(화)	무작위표현 및 연산	<ul style="list-style-type: none"> <li>- 무작위표현                             <ul style="list-style-type: none"> <li>①무작위 수 선택하기</li> <li>②무작위 수 이동하기</li> <li>③무작위로 이미지 표현하기</li> </ul> </li> <li>- 연산                             <ul style="list-style-type: none"> <li>①덧셈과 뺄셈, 곱셈과 나눗셈</li> <li>②작은 수, 큰 수</li> </ul> </li> </ul>
17-24	8.16(수)	기타 기능 및 이미지표현2	<ul style="list-style-type: none"> <li>- 기타 기능                             <ul style="list-style-type: none"> <li>①온도 센서</li> <li>②빛 센서</li> <li>③반복표현(for문, while문 활용)</li> </ul> </li> <li>- 이미지표현2                             <ul style="list-style-type: none"> <li>①이미지 움직이기</li> </ul> </li> </ul>
25-32	8.17(목)	이미지표현 심화	<ul style="list-style-type: none"> <li>- 이미지 만들기 심화                             <ul style="list-style-type: none"> <li>①이미지 나타내기(plot활용)</li> <li>②이미지 디자인하기</li> </ul> </li> </ul>
33-40	8.18(금)	프로젝트 작품 준비	<ul style="list-style-type: none"> <li>- 개인별 프로젝트 작품 계획서 작성하기</li> <li>- 개인별 프로젝트 작품 제작하기</li> </ul>
41-48	8.19(토)	프로젝트 작품 발표	<ul style="list-style-type: none"> <li>- 개인별 프로젝트 작품 발표하기</li> <li>- 사후검사지 투입(TTCT)</li> </ul>

본 연구에서는 마이크로비트를 활용한 디자인 사고 중심 SW교육을 위한 교육 프로그램으로 하루 8차시씩 6일간 총 48차시의 활동을 설계하였다.

사전 요구분석 결과 SW교육을 접해본 학생이 67.4%로 비교적 높게 나타났지만 이는 스크래치와 같은 교육용 프로그래밍 언어에만 머물러 있었다. 따라서 대부분의 학생들이 마이크로비트를 활용한 SW교육을 처음 접해봄을 고려하여 각 주제에 따라 기초부터 접근하여 점차 심화적인 내용으로 학습할 수 있도록 프로그램 내용을 설계하였다.

또한 교육 프로그램의 최종 목표를 학생들이 디자인 사고를 거쳐 마이크로비트의 여러 가지 기능을 가지고 원하는 프로그램을 자유롭게 만들도록 하는데 두어, 학생들이 마이크로비트의 전반적인 기능들을 이해하고 그 기능들을 코딩에 사용할 수 있도록 교육 주제를 선정하였다. 다만, 위의 표에는 마이크로비트의 게임 기능과 라디오에 대한 주제는 없는데, 이 두 가지 기능은 학습 시간을 고려하여 기초적인 기능을 모두 습득한 소수의 학생들에게만 별도의 심화 학습으로 제공하기로 하였기 때문이다.

1일차부터 4일차까지는 차시별로 교사의 강의 후 개인 미션을 해결하도록 하였다. 개인 미션은 배운 내용을 바탕으로 몇 가지 조건을 바꾸거나 추가해보도록 하여 비슷한 프로그램을 만들어 보는 것인데, 이러한 과정을 통해 학생들은 배운 내용을 더 빨리 습득하고 이해할 수 있게 된다. 그렇게 같은 방식으로 8차시로 구성된 하나의 주제가 완성되면 팀 미션을 해결하는 시간을 갖고 수업이 마무리 된다.

팀 미션이란 8차시만큼의 강의와 개인 미션이 모두 끝난 후 추가적인 학습지를 통해 주어지는 미션이다. 즉 학생들은 1일차부터 4일차까지 총 네 번의 팀 미션을 하게 된다. 팀 미션은 그날 배운 내용과 관련된 여러 가지 질문들을 팀원들이 함께 고민하고 해결하고, 아이디어를 내는 과정을 거치면서 완성되는데, 각각의 질문들은 학생들이 디자인 사고를 단계별로 할 수 있도록 도움을 주는 내용들로 구성하였다. 디자인 사고 단계별 미션지의 질문은 다음 <표 IV-2>의 내용과 같다.

〈표 IV-2〉 디자인 사고 단계별 미션지

디자인 사고 단계	팀미션 질문
시작하기	■ 오늘 배운 기능은 어떤 기능인가요?
발견하기	■ 오늘 배운 기능은 실생활에서 어디에 쓰이고 있나요?
해석하기	■ 이런 기능이 있으면 실생활의 어떤 불편한 점을 해결해 줄 수 있을까요? ■ 이러한 기능이 쓰이는 부분에서 불편하거나 아쉬운 점은 없나요?
아이디어내기	■ 새로운 기능이나 기존의 기능을 추가시켜 기능을 보완하거나 문제점을 해결할 수 있는 방법은 없을까요?
실험하기	■ 새로운 방법을 실현시키려면 어떻게 코딩을 해야 할까요?
발전시키기	■ 새롭게 만든 프로그램에서 보완해야 할 점이나 한계점이 있나요?

수업이 끝난 후에는 집에서 해야 하는 개별 과제를 내 학생들이 배운 내용을 복습하고, 주어진 문제를 새롭게 해결해나가는 방법을 스스로 생각해 볼 수 있는 기회를 마련하고자 하였다. 이를 통해 협업을 통한 디자인 사고 과정과 더불어, 개인적으로 디자인 사고 과정을 거치는 단계를 추가하여 창의성 신장에 더 도움이 되도록 하였다. 또 교육이 단기간동안 이루어지며 학생들이 소화해야 하는 학습량이 많다는 점을 고려하여 반복 학습이 될 수 있도록 의도하였다.

교육 5일차인 33-40차시에는 그동안 배운 내용들을 바탕으로 계획을 세워 개별 프로젝트 작품을 준비하도록 하였으며 교육 6일차에 이를 발표하게 된다. 5일차부터 진행되는 프로젝트 작품 준비 및 발표 또한 팀끼리 수행한다. 이들은 디자인 사고를 거쳐 자신들의 팀 주제 즉, 주어진 문제가 무엇인지 설정하고 그 문제를 해결하는 과정을 거쳐 팀 프로젝트를 설정한다. 이는 팀 브라운의 디자인 사고 과정 중 시작하기(start), 발견하기(discovery) 및 해석하기(interpretation)에 해당하는 단계이다. 이후 팀원들과 협업을 통해 아이디어내기(ideation), 실험하기(experimentation), 발전시키기(evolution)의 단계를 거쳐 디자인 사고에 의한 문제해결의 과정을 거치게 된다. 이 때, 팀 프로젝트의 주제는 하나이고 팀원들과 그 문제를 해결해나가지만 프로그램은 개

인이 따로 만들도록 하여 집단 학습과 개별학습의 조화를 추구하였다.

정리하자면, 교수학습 방법으로 사전요구조사 결과를 바탕으로 마이크로비트 기본 내용은 강의/실습형을 기본으로 하되, 개인미션과 팀미션을 마련하여 개인과제형과 그룹과제형이 적절히 이루어지도록 하였다. 최종 프로젝트 발표 준비부터 발표 또한 그룹과제형과 개인과제형이 적절히 주어지도록 하였다.

또 팀 미션과 프로젝트 준비, 수업 후 주어지는 과제를 수행할 때 디자인 사고가 단계별로 적용되어 학생들이 배운 내용에 대해 사고를 더 확장할 수 있도록 하였다. 디자인 사고를 거쳐 실현시켜 본 프로그램은 학생들의 코딩 능력과 창의성 향상에 더욱 더 도움이 될 것이라 기대하였다.

## 나. 교육 프로그램 적용

### 1) 연구대상

제주 도내 초등학교 4, 5, 6학년층을 대상으로 신청절차에 따라 선별 없이 선착순 선발을 하였다. 선발된 학생 수 및 학년, 성별의 분포는 다음과 같다.

〈표 IV-3〉 지원자 성별과 학년 분포

성별	4학년	5학년	6학년	합계
남	7	4	6	17
여	0	6	2	8
합계				25

### 2) 연구 방법

본 연구에서 개발한 교육 프로그램과 교재를 이용하여, 계획대로 총 48차시의 교육을 실시하였다. 수업은 교수-학습 과정안을 통해 정해진 순서대로 교육하되, 성취도에 따라 교사가 개별지도 혹은 자유 심화학습을 시키며 전체적인 학습 성취도 및 창의성 향상에 도움이 되도록 하였다. 앞서 말했듯 8차시로 구성

된 하나의 주제가 완성되면 팀 미션을 수행하도록 하였으며 팀은 본 연구자가 1일차 수업을 진행한 뒤 학년과 코딩 학습 수준을 고려하여 한 팀당 3명~4명으로 구성하였다. 다음은 17-24/48차시 '마이크로비트의 여러 가지 기능을 알아보고 다양한 방법으로 이미지 표현하기'수업의 교재와 교수-학습 과정안의 일부이다.

〈표 IV-4〉 교수-학습 과정안

일시	2017. 8. 16.		대상	마이크로비트 B반
주제	온도 센서, 빛 센서, 반복표현, 이미지 움직이기		차시	17~24/48
활동명	마이크로비트의 다양한 기능과 이미지 표현 알아보기		소요 시간	400분
학습 목표	마이크로비트의 여러 가지 기능을 알아보고 다양한 방법으로 이미지 표현을 할 수 있다.			
학습 단계	학습 요소	교수·학습 활동		시간
도입	전시 학습 상기	<ul style="list-style-type: none"> <li>●복습하기</li> <li>▶ 지난 시간에 배웠던 내용을 말해 봅시다. <ul style="list-style-type: none"> <li>- 수나 이미지를 무작위로 뽑도록 하는 방법에 대해 배웠습니다.</li> <li>- 마이크로비트로 덧셈, 뺄셈, 큰 수 찾기 프로그램을 만들어 보았습니다.</li> </ul> </li> <li>▶ 지난 시간에 배웠던 내용을 복습해 봅시다.</li> </ul>		20'
	동기 유발	<ul style="list-style-type: none"> <li>●동기유발</li> <li>▶ 여러 가지 센서를 이용한 물건 사진보기</li> <li>▶ 사진 속 물건이 어떤 기능을 하는지 발표하기 <ul style="list-style-type: none"> <li>- 온도가 낮으면 난방이 켜집니다.</li> <li>- 어두워지면 가로등이 켜집니다.</li> </ul> </li> <li>▶ 사진 속 물건이 제 기능을 하기 위해 어떤 방법을 사용했는지 토의하기 <ul style="list-style-type: none"> <li>- 빛의 양을 측정하여 빛의 양이 줄어들면 불이 켜지도록 했을 것 같습니다.</li> <li>- 온도계를 이용했을 것 같습니다.</li> </ul> </li> <li>▶ 사진 속 물건들은 같은 패턴의 일을 반복합니다. 같은 일을 반복하도록 프로그래밍 하려면 어떤 방법을 사용하면 좋을까요?</li> </ul>		20'

		<ul style="list-style-type: none"> <li>- 원하는 기능을 프로그래밍한 뒤 그 부분을 복사하여 여러 번 이어 붙입니다.</li> </ul>	
	<p>학습 문제 제시 및 활동 안내</p>	<p>●학습문제 제시</p> <div style="border: 1px solid black; padding: 5px; text-align: center;"> <p><b>마이크로비트의 여러 가지 기능을 알아보고 다양한 방법으로 이미지 표현을 해 봅시다.</b></p> </div> <p>●활동 안내</p> <p>[활동1]온도센서를 이용한 온도 체크기 만들기</p> <p>[활동2]빛 센서를 이용한 가로등 만들기</p> <p>[활동3]여러 가지 방법으로 이미지 움직이기</p>	
전개	온도센서를 이용한 프로그래밍	<p>●[활동1]온도센서를 이용한 온도 체크기 만들기</p> <ul style="list-style-type: none"> <li>▶ 온도센서를 이용한 온도 체크기 시연해보기</li> <li>▶ 온도 체크기의 원리가 무엇인지 생각해보기</li> <li>- 마이크로비트에 온도를 감지하는 센서가 있는 것 같습니다.</li> <li>- if문으로 조건을 넣어 온도를 감지하도록 했을 것 같습니다.</li> <li>▶ 마이크로비트 온도 센서 확인해보기</li> <li>▶ 두 가지 방법으로 온도 체크기 만들기</li> <li>▶ 미션 해결하기(조건 추가)</li> </ul>	90'
	빛 센서를 이용한 프로그래밍	<p>●[활동2]빛 센서를 이용한 가로등 만들기</p> <ul style="list-style-type: none"> <li>▶ 빛 센서를 이용한 가로등(마이크로비트) 시연해보기</li> <li>▶ 가로등의 원리가 무엇인지 생각해보기</li> <li>- 마이크로비트에 빛을 감지하는 센서가 들어있고 그 센서 기능을 이용해 조건문을 넣어 프로그래밍 했을 것 같습니다.</li> <li>- 불이 켜지고 꺼지는 빛의 양을 정해놓는 프로그램이 적용됐을 것 같습니다.</li> <li>▶ 마이크로비트 빛 센서 확인해보기</li> <li>▶ 가로등 만들기</li> <li>▶ 미션 해결하기(스위치 기능 추가)</li> <li>▶ 블록 의미 정리하기</li> <li>- temperature, light level</li> </ul>	90'
	움직이는 이미지 프로그래밍	<p>●[활동3]여러 가지 방법으로 이미지 움직이기</p> <ul style="list-style-type: none"> <li>▶ 반복해서 수 출력하기</li> <li>▶ 미션 해결하기</li> <li>▶ While문을 이용한 반복해서 수 출력하기</li> <li>▶ 미션 해결하기</li> <li>▶ 동작을 반복하다 10초 후 멈추기</li> <li>▶ 변수를 만들어서 이미지 움직이기</li> </ul>	90'

		▶ 미션 해결하기	
정리	정리 및 과제 제시	●정리하기 ▶ 팀 미션 수행하기 ●과제 제시 ▶ 여러 가지 센서 기능과 이미지 표현 관련 자유 주제 프로그램 만들기(배운 것 활용)	90'

### 3) 창의성 분석 방법

창의성 측정을 위해 교육 전과 후를 기준으로 사전, 사후검사를 실시하였다. 본 연구는 창의성 검사도구로 Torrance(1972)의 TTCT(Torrance Tests of Creative Thinking) 검사지 중 도형 A형을 사용하였다. TTCT 도형 검사지는 사고 능력과 창의성을 측정하기 위한 검사 도구로, 과거부터 20년간 전 세계 창의성 연구의 약 75% 이상이 창의성 검사에 이용하고 있다.(김경희, 2006) TTCT 도형 검사는 도형과 연관 지어 그리는 각기 다른 세 가지 활동을 통해 창의성 영역을 '유창성', '독창성', '제목의 추상성', '정교성', '성급한 종결에 대한 저항'이라는 하위 요소로 나누어 점수화 시킨다.

창의성 사전사후검사에는 통제집단설계(pre test-post test control group design)가 사용되었다.

〈표 IV-5〉 창의성 검사 실험 설계

	사전검사	적용	사후검사
실험 집단 (25명)	O <sub>1</sub>	X	O <sub>2</sub>

X: 디자인 중심 마이크로비트 활용 SW교육 프로그램 학습 실시

O<sub>1</sub>: 사전검사(Computational Cognition Test)

- Independent sample T-test

O<sub>2</sub>: 사후검사(Computational Cognition Test)

- Independent sample T-test

## 2. 연구 결과

### 가. 창의성 검사 정규성 검정

‘Shapiro-Wilks 정규성 검정’을 통해 창의성 검사 결과에 대한 정규성 검정을 실시하였다. 검사 결과는 <표 IV-6> 와 같다.

<표 IV-6> 창의성 하위 요소에 대한 정규성 검정 결과

하위 요소	평균	표준편차	최대값	최소값	stat	유의확률
유창성	114.2692	15.54750	140.00	65.00	.919	.043*
독창성	101.6154	17.36451	127.00	70.00	.939	<b>.128</b>
제목의 추상성	68.2308	39.38965	120.00	0.00	.908	.023*
정교성	66.1538	9.90633	86.00	57.00	.826	.001*
성급한 종결에 대한 저항	28.2308	35.41221	97.00	0.00	.740	.000*
평균	75.6538	16.41595	101.80	47.00	.925	<b>.058</b>
창의성 지수	76.1538	16.84017	101.80	47.00	.917	.037*

\* $p < .05$

정규성 검정 결과 여러 하위요소 중 ‘독창성’과 ‘평균’ 항목이 정규분포를 이루는 것으로 나타났다. 나머지 하위요소들은 정규분포를 이루지 않았다.

### 나. 창의성 집단 내 비교

창의성 향상의 유무를 알아보기 위해 집단 내 사전·사후 검사 결과 비교를 실시하였다. 창의성 하위요소에 대한 정규성 검정에서 정규분포를 이뤘던 ‘독창성’과 ‘평균’항목에 대해서는 ‘대응표본 t검증’을 하였으며 정규분포를 이루지 않았던 나머지 하위요소들은 ‘Wilcoxon 부호 순위 검정’을 통해 비교하였다. 그

결과는 다음 <표 IV-7>과 <표 IV-8>과 같다.

<표 IV-7> 정규분포를 이룬 하위요소에 대한 사전·사후 검사 결과 비교(대응 표본 t검증)

하위 요소	시기	평균	표준편차	t	유의확률
독창성	사전	101.6154	17.36451	-7.734	.000***
	사후	121.6538	17.39642		
평균	사전	75.6538	16.41595	-4.237	.000***
	사후	87.5231	11.17756		

\*p<.05, \*\*p<.01, \*\*\*p<.001

<표 IV-8> 정규분포를 이루지 않는 하위요소에 대한 사전·사후 검사 결과 비교(Wilcoxon 부호 순위 검정)

하위요소	시기	평균	표준편차	Z	유의확률
유창성	사전	114.2692	15.54750	-3.241 <sup>b</sup>	.001**
	사후	124.4231	9.47491		
제목의 추상성	사전	68.2308	39.38965	-2.555 <sup>b</sup>	.011*
	사후	82.0385	31.01417		
정교성	사전	66.1538	9.90633	-.085 <sup>b</sup>	.932
	사후	66.1923	8.42861		
성급한 종결에 대한 저항	사전	28.2308	35.41221	-1.819 <sup>b</sup>	.069
	사후	43.3077	16.86124		
창의성 지수	사전	76.1538	16.84017	-3.377 <sup>b</sup>	.001**
	사후	87.9462	11.45475		

\*p<.05, \*\*p<.01, \*\*\*p<.001

창의성 사전·사후검사 결과를 비교·분석한 결과 ‘독창성’에서는 평균점수 101.61에서 121.65로 20점 이상 상승하였고 이는 통계적으로 유의했다. ‘평균’ 또한 평균점수 75.65에서 87.52로 약 12점 상승하였고 이는 통계적으로 유의했다. ‘유창성’에서는 평균점수가 10.16점 상승하였고, 이 결과는  $p=0.01(p<.05)$ 로 통계적으로 유의하였다. ‘제목의 추상성’에서는 사전검사 평균점수가 68.23에서 82.03으로 상승하였으며 유의확률  $p=0.011(p<.05)$ 로 유의했다. ‘정교성’은 큰 차이가 없었으며 ‘성급한 종결에 대한 저항’은 28.23에서 43.30으로 다소 상승하였으나 유의확률  $p=0.69$ 로 통계적 유의성이 없었다. ‘창의성 지수’는 76.15에서 87.94로 상승했으며 유의확률  $p=0.01$ 로 통계적으로 유의했다.

따라서 유창성, 제목의 추상성, 독창성, 평균, 창의성 지수에서 교육 후 통계적으로 유의한 향상이 있었음을 알 수 있었다. 이는 마이크로비트가 생각한 주제에 따라 LED화면을 구성하는 디자인을 비롯하여 여러 구성요소들의 쓰임과 프로그램의 흐름 등 다양한 방안을 고려하여 프로그래밍 하는 것이 중요하기 때문인 것으로 보인다. 학생들은 매 차시마다 각기 다른 주제의 프로그램을 제작해보는 과정을 거치면서 유창성이 향상되었을 것이다. 그리고 프로그램의 쓰임에 따른 주제와 제목을 직접 만들어보며 창의성 영역 중 제목의 추상성과 독창성이 향상된 것으로 보인다. 또한 매 수업마다 디자인 사고가 적용된 그룹 활동을 통해 학생들은 더 효과적으로 아이디어를 도출하고 주어진 문제를 해결하며 프로그래밍을 할 수 있었다. 그렇게 그룹이 함께 프로그램을 만드는 과정과 수업이 끝난 후 다시 그 과정을 개별 과제로 복습해보는 과정을 통해 학생들의 전체적인 평균과 창의성지수를 향상시켰다고 볼 수 있다.

### 3. 연구 결과 분석

본 연구 결과, 유창성, 제목의 추상성, 독창성, 평균, 창의성 지수에서는 교육 후 통계적으로 유의한 향상이 관찰되었다. 정교성과 성급한 종결에 대한 저항에서도 통계적 유의성은 관찰되지 않았으나 대부분의 학생에서 점수가 향상되

는 경향성은 관찰되었다. 그리고 모든 평가 값을 아우르는 창의성 지수의 경우 통계적으로 유의한 향상이 관찰되었다. 창의성이란 본 연구에서 조사된 여러 요인이 복합적으로 작용하여 발현되는 능력이며, 본 연구 대상의 수가 많지 않아 세부적인 지표에 대한 분석은 용이하지 않았다. 하지만 교육 전후로 대부분의 세부지표가 통계적으로 유의미한 상승을 보였다는 점, 통계적 의미는 없었지만 나머지 지표에서도 향상되는 경향을 보였다는 점, 창의성 지수가 통계적으로 유의한 상승을 보였다는 점을 종합하면 본 교육이 학생들의 창의성 향상에 효과가 있었음을 확인할 수 있었다. 요구 조사에서 연구 대상자들은 다수가 SW교육을 받은 경험이 있었으며, 그 중 대다수가 교육용 프로그래밍 언어를 이용한 교육이었다. 또한 디자인 사고가 요구되는 그룹과제형의 교육과 실물을 이용한 교육 및 과제 확인에 대한 요구도가 높은 상태였다. 이를 통해 디자인 사고와 실물교육을 통한 SW교육이 기존에 부재했거나, 혹은 기존 경험에서 그러한 교육이 더 즐겁고 도움이 됐음을 유추할 수 있다. 그러므로 본 연구에서 도입한 디자인 사고와 마이크로비트를 이용한 실물교육이 학생들의 요구를 반영함과 동시에 창의성 신장에 도움을 줬다고 판단할 수 있었다.

## V. 결론 및 제언

본 연구는 마이크로비트를 활용한 디자인 사고 중심 SW교육이 초등학생의 창의성 향상에 미치는 효과를 알아보고자 시행하였다. 이를 위하여 디자인 사고가 적용된 마이크로비트 활용 SW교육 프로그램을 개발하였다. 교육 효과를 높이고자 프로그램 개발을 위해 도내 4, 5, 6학년 초등학생을 대상으로 사전 요구조사를 실시하였다. 그 결과를 바탕으로 학습 내용을 선정하였고 학생들이 선호하는 교수 방법인 그룹과제형과 실물교육을 채택하여, 디자인 사고를 적용한 교수법과 함께 마이크로비트의 장점을 활용하여 실물교육을 적용하였다. 이를 4,5,6학년 초등학생 25명을 대상으로 6일 동안 총 48차시로 구성된 수업을 실시하였다.

SW교육으로 인한 창의성 향상의 유무를 알아보기 위해 창의성 검사도구인 Torrance(1972)의 TTCT 검사지 도형 A형을 교육 전과 후로 나누어 실시하였다. 그 결과 유창성, 제목의 추상성, 독창성, 평균, 창의성지수 항목에서 통계적으로 유의미한 향상을 보였으며, 나머지 항목에서도 전반적인 향상을 보였다. 또한 창의성 지수도 통계적으로 의미 있는 향상을 보여, 창의성이 전반적으로 향상되었음이 확인되었다.

따라서, 본 연구를 통해 개발한 SW교육 프로그램은 초등학생의 창의성 향상에 효과적인 것으로 나타났다. 이는 기존의 이론교습과 더불어 시행된 그룹과제 및 개인과제를 통한 디자인 사고의 활용이, 요구 분석에서 보였던 학생들의 요구를 반영하여 흥미 유발에 도움이 됨과 동시에 기존에 SW교육을 받았던 학생들에게서도 유의미한 창의력 신장을 불러올 수 있는 것으로 보인다. 또한 활용 기기였던 마이크로비트가 휴대 가능하다는 점, 기기를 통해 코딩 결과를 바로 확인할 수 있어 즉각적인 피드백이 가능하다는 점은 학생들로 하여금 SW교육에 흥미를 가질 수 있는 요소가 되었고 그로 인한 교육 효과가 있음을 알 수 있다. 그러므로 마이크로비트를 활용하여 디자인 사고를 중심으로 한 SW교육 프로그램을 개발, 시행시키는 것이 앞으로의 초등 SW교육에 큰 도움이 될 것으로 예상된다.

본 연구는 창의성에 대하여 실증적으로 효과를 검증했다는 의의가 있으나 몇

가지 한계가 존재한다.

첫째, 본 연구 대상의 수는 25명으로 그 수가 적다는 단점이 있다.

둘째, 연구 결과를 비교할 수 있는 비교집단이 존재하지 않는다.

셋째, 사전요구조사를 위한 설문 대상의 수는 183명으로 4,5,6학년 초등학생들의 SW요구를 일반화하기에 그 수가 적다는 단점이 있다.

따라서 본 연구 결과를 일반화 하는 데 한계가 있으나, 향후 본 연구를 뒷받침 할 수 있는 몇 가지 한계를 개선한 연구가 시행된다면 마이크로비트를 활용한 디자인적 사고 중심 SW교육의 효과 및 초등학생의 창의성 향상 연구에 큰 도움이 될 것으로 기대된다.

## 참 고 문 헌

- 교육부. (2015). **소프트웨어 교육 운영지침**.
- 교육부. (2016). **소프트웨어 교육 활성화 기본 계획**.
- 교육부. 한국교육학술정보원. (2017). **스마트기기를 활용한 소프트웨어(SW) 교육 : 초등**.
- 김경희. (2006). **Can we trust creativity test? A Review of the Torrance Tests of creative thinking(TTCT)**. Creativity Research Journal, 18(1), 3-14.
- 김나연. (2016). **SW교육 수업 구성이 컴퓨팅 사고력과 태도 및 인식 변화에 미치는 영향 분석**. 인천대학교 석사학위논문.
- 김다은. (2017). **해외사례와 비교를 통한 국내 SW교육의 한계점과 개선방안**. 숭실대학교 석사학위 논문.
- 박진주. (2008). **교육에 있어서 실물과 언어의 위치**. 춘천교육대학교 석사학위 논문.
- 이경화. (2002). **4, 5세 유아의 창의 능력과 창의 성격**. 교육심리연구, 16(3), 147-160.
- 이선영. (2016). **디자인 사고 기반 문제 해결 학습에서 초등학생이 경험하는 집단 창의성의 속성 인식 분석**. 한국교원대학교 석사학위 논문.
- 전경원. (2000). **동·서양의 하모니를 위한 창의학**. 서울: 학문사.
- 정원식 이영덕. (1993). **표준화 창의성 검사 실시 요강 및 기준**. 서울: 코리안 테스트센터.
- 중앙일보. (2015). **소프트웨어(SW) 강자, 글로벌 시장서 살아남는다**.
- 한국교육학술정보원, 양재명, 김자미, 이원규, 윤일규, 서정희, 우호성, 양혜지, 김민정, 최희정. (2017). **초·중등 소프트웨어(SW)교육 역량 진단 도구 개발 연구**.
- Bruner Jerome S. (1960). **The process of Education**. 이홍우(역)(1966), 『브루너 교육의 과정』, 서울: 배영사.
- Davis. G. A. (2005). **Creativity is forever**. Dubuque, IA: Kendall/Hunt Publishing.

- Dewey. J.(1903). **The Child and the Curriculum**. 박철홍(역)(2006), 『아동과 교육과정』, 서울: 문음사.
- Guilford. J. P. (1978). **The nature of human intelligence**. NY: McGraw-Hill.
- Hennessey. B & Amabile. T. (1988). **Storytelling as a means of assessing creativity**. Journal of creative Behavior, 22, 235-247.
- IDEO. (2012). **Design thinking for educators**.
- Martin. R. L. (2009). **Design thinking**. Seoul: Woongjin Wings Press.
- Rhodes. M.. (1961). **Analysis of creativity**. Phi Delta Kappan, 42(7), 305-310.
- Sternberg. R. J. & Lubart. T. I. (1996). **Investigating in creativity**. American Psychologist, July.
- Sue Sentance, Jane Waite, Steve Hodges, Emily MacLeod & Lucy Yeomans. (2017). **"Creating Cool Stuff" - Pupils' experience of the BBC micro:bit**. In Proceedings of the 48th ACM Technical Symposium on Computer Science Education: SIGCSE 2017. DOI: 10.1145/3017680.3017749.
- T. Ball, J. Protzenko, J. Bishop, M. Moskal, J. de Halleux, M. Braun, S. Hodges, and C. Riley. (2016). **Microsoft touch develop and the bbc micro:bit**. In Proceedings of the 38th International Conference on Software Engineering Companion, ICSE '16, pages 637 - 640, New York, NY, USA, 2016. ACM.
- The Guardian. (2015). **BBC to give away 1m Micro:bit computers to school children**.
- Tim Brown. (2008). **Desing thinking**. Harvard Business Review. June. 2008. Harvard Business School Publishing Corporation.
- Torrance E. P. (1972). **Predictive validity of the Torrance test of creative thinking**. Journal of creative Behavior, 6, 236-252.
- micro:bit 웹사이트. <http://microbit.org>

# A B S T R A C T \*

## A Study on the Design Thinking-based SW Education using micro:bit

Seoyoung Park

Major in Elementary Practical Computer Education  
Graduate School of Education  
Jeju National University

Supervised by Professor Jonghoon Kim

The purpose of this study is to investigate the effects of design thinking software based on micro-bit on the creativity of elementary school students. To do this, we conducted preliminary surveys on SW education for 4th, 5th, and 6th grade elementary students in Jeju Island randomly selected and developed a software education program and a textbook based on design thinking using micro beat. Using these educational programs, 25 elementary school students in the 4th, 5th, and 6th grades selected on the first-come-first-served basis in Jeju Island conducted a total of 48 training sessions. The TTCT test was divided into two groups, pre- and post-test, and the collected data were analyzed by t-test. As a result, the educational programs developed in this study were found to be effective in enhancing the creativity of elementary school students.

---

\* A thesis submitted to the committee of Graduate School of Education, Jeju National University in partial fulfillment of the requirements for the degree of Master of Education conferred in August, 2018.

## 부 록

[부록 1] 교육 프로그램 교재

[부록 2] 교육 프로그램 과정안



# 마이크로비트(microbit)로 놀자!

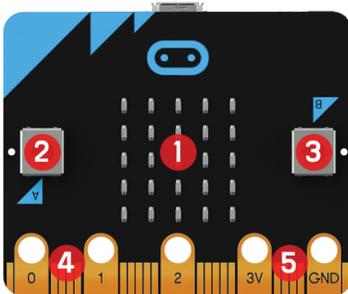
제주대학교 교육대학 초등컴퓨터교육대학원  
창의컴퓨터교실 12기 교육자 박서영

## √마이크로비트는 무엇일까?

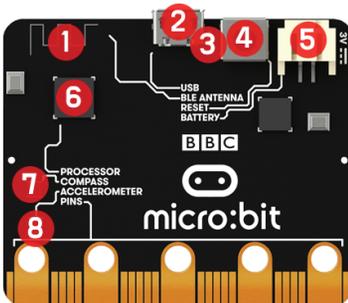
마이크로비트는 아주 단순한 컴퓨터라고 할 수 있습니다. 컴퓨터는 입력을 받고, 이를 처리하여, 결과를 보여주는 기계죠. 마이크로비트는 아주 작은 기판의 형태이지만 이 세 가지 기능을 모두 포함하고 있어요.

작고 단순한 마이크로비트는 내가 코딩하여 만든 프로그램을 컴퓨터와 같이 실행시켜보고, 직접 만져 조작해보는 것을 가능하게 합니다.

## √마이크로비트 꼼꼼히 뜯어보기!



- ① LED램프이며 왼쪽 상단이 (0,0)좌표입니다.
- ② A 버튼입니다.
- ③ B 버튼입니다.
- ④ 센서 연결 핀입니다.
- ⑤ 전원 연결에 사용합니다.



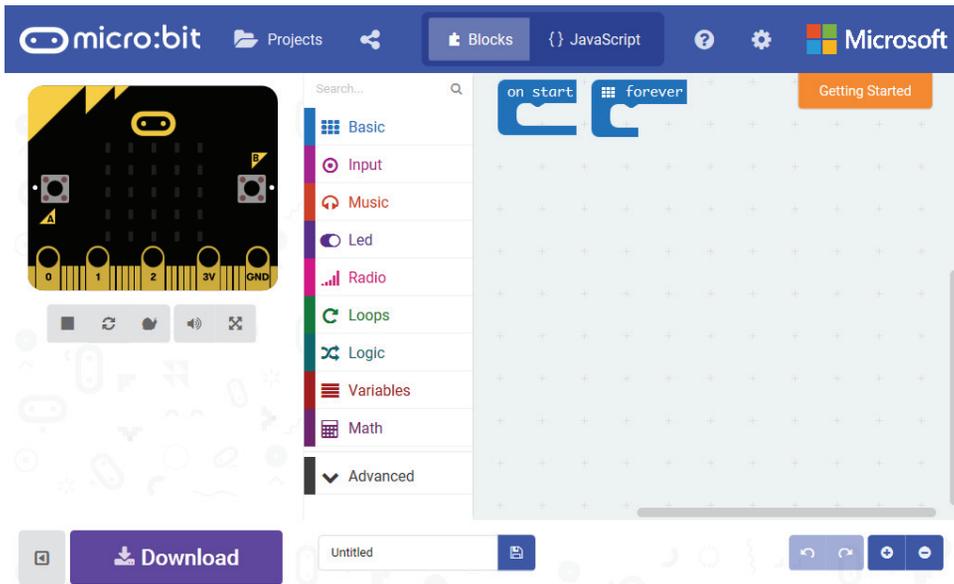
- ① 블루투스 안테나로 컴퓨터, 스마트폰 등과 연결할 수 있습니다.
- ② USB케이블로 컴퓨터와 연결할 수 있습니다.
- ③ 이상이 있을 때 불빛을 깜빡여서 알려줍니다.
- ④ 재시작 버튼입니다.
- ⑤ 배터리 팩을 연결할 때 사용합니다.
- ⑥ 프로세서
- ⑦ 컴퍼스(나침반) ⑧ 가속도센서

Q. 마이크로비트를 다섯 글자로 표현해 볼까요?

√마이크로비트 코딩 프로그램 접속하기

방법) <http://microbit.org> 접속 - Let's Code 클릭

※선생님의 설명을 들으며 각각의 기능들을 살펴보아요.



√마이크로비트 명령어블록

<b>Basic</b>	문자, 숫자 출력/LED 모두 끄기 등의 기본 기능을 수행하는 명령어 모음입니다.
<b>Input</b>	버튼/흔들기/센서 등의 외부 입력과 관련된 명령어 모음입니다.
<b>Loops</b>	반복 수행과 관련된 명령어 모음입니다.
<b>Logic</b>	논리적 판단을 추가하고자 할 때 사용하는 명령어 모음입니다.
<b>Variables</b>	데이터를 저장하는 변수를 관리하는 명령어 모음입니다.
<b>Maths</b>	수학적인 연산과 관련된 명령어 모음입니다.

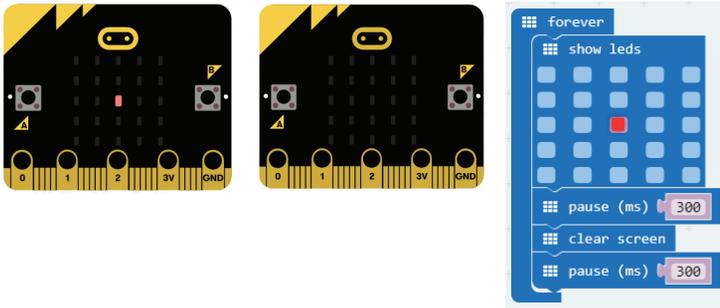
## √마이크로비트 코딩하기

### <chapter 1. 이미지표현>

마이크로비트의 가장 기본이 되는 LED조작 방법을 알아봅시다.

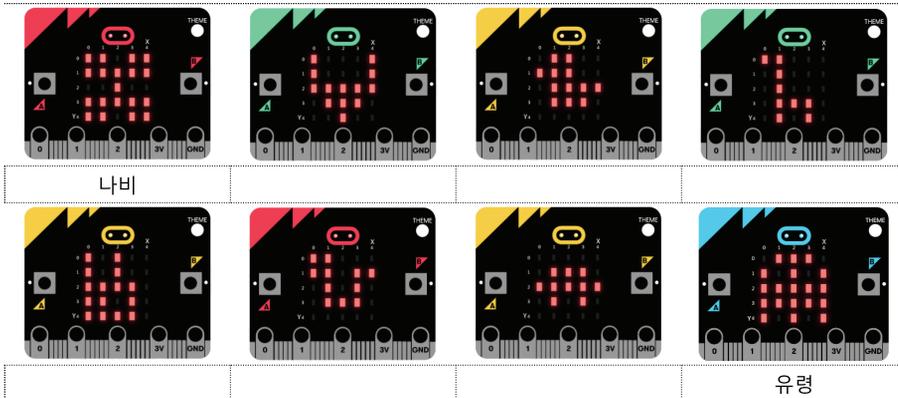
#### ☆ LED 켜다 끄기 반복하기

LED불빛이 깜빡이도록 프로그램을 만들어 봅시다.

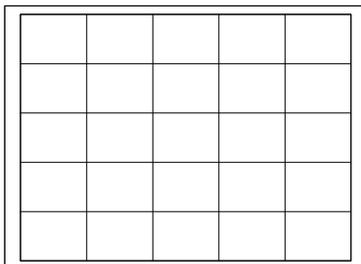


#### ☆ 다양한 동물 이미지

5X5 LED로도 다양한 이미지를 만들 수 있습니다. 다음의 모양을 보고 어떤 모양을 나타낸 것인지 예상해 봅시다.



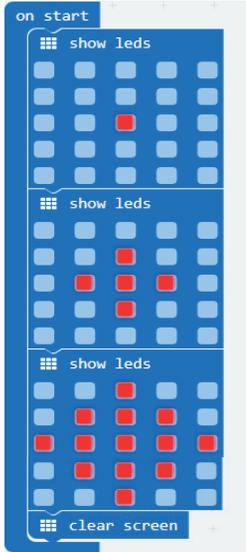
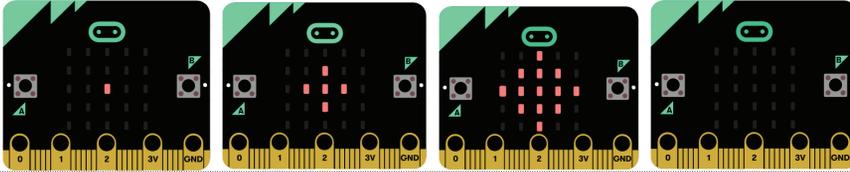
[미션①] 아래의 칸을 색칠하여 나만의 이미지를 만들어 보세요.



이미지 제목:

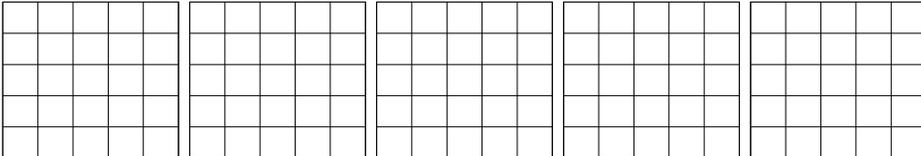
## ☆ 움직이는 이미지 만들기

다음 모양이 순서대로 나오는 프로그램을 만들어 봅시다.



비슷한 모양이 반복되면 이미지가 움직이는 것 같은 느낌을 줍니다.

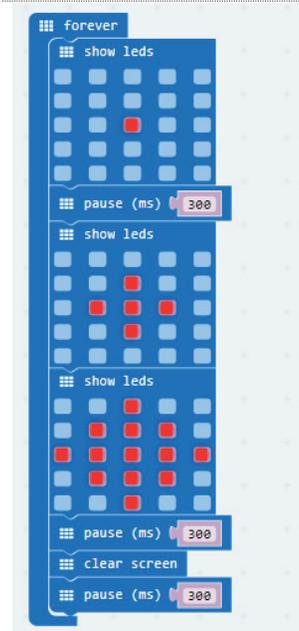
[미션②] 위와 같이 여러 개의 이미지를 이어 붙여 움직이는 이미지를 만들어 보세요.



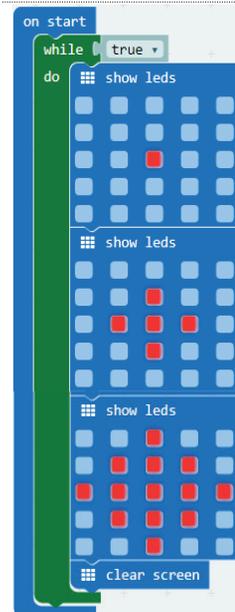
## ☆ 움직이는 이미지 반복하기

위의 움직이는 이미지를 반복하게 하려면 어떻게 해야 할까요? 여러 가지 방법을 알아봅시다.

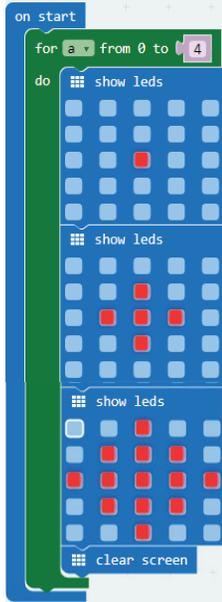
### 방법1) forever블록 사용하기



### 방법2) while블록 사용하기



방법3) for블록 사용하기

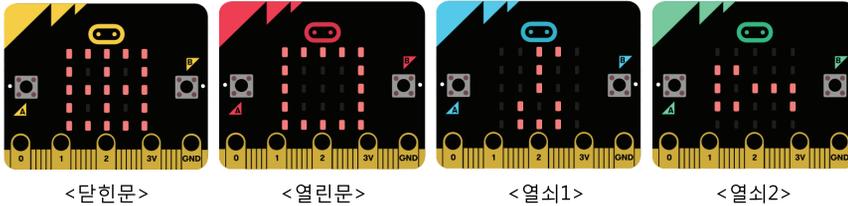


※ for블록은 변수 설정이 필요하고, 정한 수만큼 반복합니다.  
변수란 내가 만드는 가상의 수를 말합니다.

옆의 프로그램은 움직이는 모양이 몇 번 반복하게 될까요?

번

[미션③] 닫혀있던 문이 열쇠를 5번 돌리면 열리는 프로그램을 만들어 보세요.



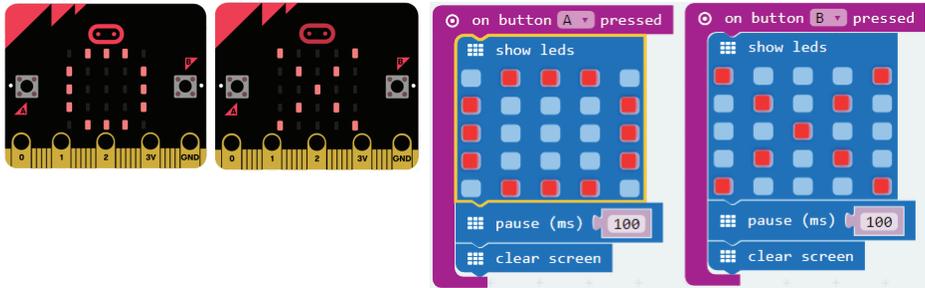
★블록의 의미를 써 봅시다.

<b>forever</b>	반복한다.
<b>while</b>	(조건만큼) 반복한다.
<b>for</b>	(조건외 횟수만큼) 반복한다.
<b>show leds</b>	만든 led화면을 보여준다.
<b>pause (ms)</b>	정한 시간만큼 멈춘다. (1초가 100)
<b>clear screen</b>	led화면을 모두 지운다.

### ❖ 버튼을 눌러 이미지 보기

마이크로비트에는 A와 B 두 개의 버튼이 있습니다. 버튼을 눌러 원하는 이미지가 나타나게 하는 방법을 알아봅시다.

A버튼을 누르면 O모양, B버튼을 누르면 X모양이 뜨는 프로그램을 만들어 봅시다.



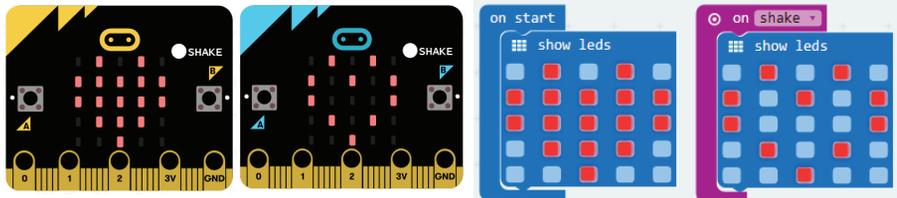
★블록의 의미를 써 봅시다.

<b>on button A pressed</b>	버튼 A가 눌렸을 때 실행한다.
----------------------------	-------------------

### ❖ 흔들어서 이미지 바꾸기

마이크로비트에는 흔들림을 감지하는 센서가 있습니다. 마이크로비트를 흔들면 이미지가 바뀌도록 하는 방법을 알아봅시다.

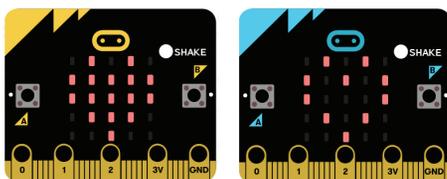
흔들면 다음 이미지로 넘어가는 프로그램을 만들어 봅시다.



★블록의 의미를 써 봅시다.

<b>on shake</b>	흔들었을 때 실행한다.
<b>on start</b>	시작했을 때 실행한다.

[미션④] 흔들면 다음 이미지로 3초간 넘어갔다가 다시 처음 이미지로 돌아오는 프로그램을 만들어 봅시다.



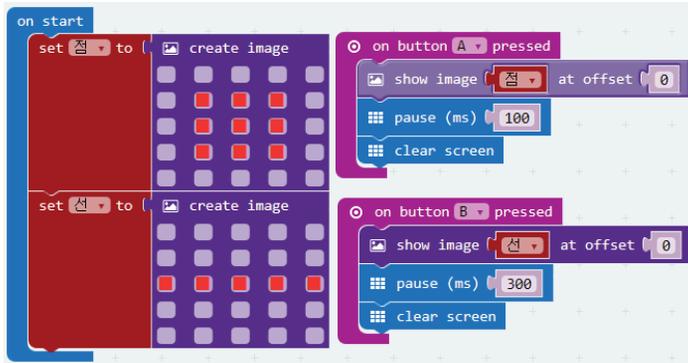
처음

흔들었을 때(3초)

★ 변수를 만들어서 이미지 바꾸기

변수 만들기는 내가 필요한 이미지 또는 가상의 수를 만들어 두는 것입니다. 변수를 만들면 조금 더 편리하게 프로그램을 만들 수 있습니다.

버튼을 눌러 모스부호를 표현할 수 있는 프로그램을 만들어 봅시다.



A	B	C	D	E	F	G	H	I	J	K	L	M
.-	-...	-.-.	-..	.	..-	--.	....	..	...-	-.-	.-.	--
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
-.	---	-.-	---.	.-.	...	-	..-	...-	.-.	-.-	---.	---

[미션⑤] 마이크로비트에 모스부호 프로그램을 넣어 'HELP'가 되도록 버튼을 눌러보세요.

HELP:

★블록의 의미를 써 봅시다.

Make a Variable	변수 만들기
set 점 to ~	'점'을 ~로 정해놓는다.
create image	이미지를 만든다.(변수용)
show image 점 at offset 0	변수로 저장한 '점'을 offset 0의 위치에서 보여준다.

❖ 변수를 이용하여 버튼 카운터 만들기

```

on start
  set 점수 to 0

on button A pressed
  change 점수 by 1
  show number 점수
  
```

[미션⑧] B버튼을 누르면 점수 1이 감소하고, A+B 버튼 누르면 점수 값을 0으로 초기화 하는 기능을 추가하세요.

❖ 누가 많이 흔들었나?

[미션] 다음의 조건을 충족하는 프로그램을 만드세요.

```

on start
  set 점수 to 0

on shake
  ?

on button A pressed
  show number 점수

on button B pressed
  ?
  show number 점수
  
```

- 마이크로비트를 흔들면 '점수' 값이 1씩 증가합니다.
- A 버튼을 누르면 '점수' 값을 보입니다.
- B 버튼을 누르면 '점수' 값이 0으로 초기화하고 '점수' 값을 보입니다.

★블록의 의미를 써 봅시다.

show number □	□의 수를 보여준다.
change 점수 by 1	점수를 1씩 증가시킨다.

♥과제♥ 이미지 관련 자유 주제의 프로그램을 만드세요. (배운 것들 활용하여 만들기)

과제 계획:

<chapter 2. 무작위표현 & 수학적 연산>

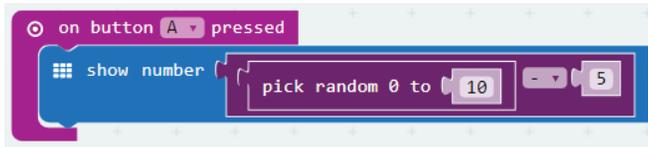
무작위 수를 출력하는 방법 및 기본적인 수학적 연산 방법을 알아봅시다.

❖ 무작위 수 선택하기

수 범위를 정해놓고 그 안에 있는 수 중 한 수를 무작위로 뽑아봅시다.



-5부터 5까지의 수 중 하나를 무작위로 뽑는 프로그램을 만들어 봅시다.



블록의 의미를 써 봅시다.

<b>pick random 0 to 9</b>	0부터 9까지의 수 중 무작위로 하나의 수를 뽑는다.
$\square + \square$	$\square$ 와 $\square$ 를 더한다.
$\square - \square$	$\square$ 에서 $\square$ 를 뺀다.

❖ 무작위로 이동하기

5X5 25개의 LED에서 무작위로 이동하는 프로그램을 만들어 봅시다.

plot 블록  
x축

(0,0)	(1,0)	(2,0)	(3,0)	(4,0)
(0,1)	(1,1)	(2,1)	(3,1)	(4,1)
(0,2)	(1,2)	(2,2)	(3,2)	(4,2)
(0,3)	(1,3)	(2,3)	(3,3)	(4,3)
(0,4)	(1,4)	(2,4)	(3,4)	(4,4)

y축

( x , y )

※ 5X5의 LED를 x좌표, y좌표로 두면 위와 같은 좌표 값을 갖습니다. x와 y로 변수를 만들어 놓고 plot 블록을 이용하면 내가 원하는 좌표의 LED를 켜거나 끌 수 있습니다.

☆ 무작위로 이미지 표현하기

몇 가지의 정해놓은 이미지 중 무작위로 하나의 이미지가 나오는 프로그램을 만들어 봅시다.

<방법1>

```

on button A pressed
  set 선택 to pick random 0 to 2
  if (선택 = 0)
  then
    show leds
  else if (선택 = 1)
  then
    show leds
  else
    show leds
  
```

<방법2>

```

on start
  set 행복함 to create image
  set 슬픔 to create image
  set 화난 to create image

on button A pressed
  set 선택 to pick random 0 to 2
  if (선택 = 0)
  then
    show image 행복함 at offset 0
  else if (선택 = 1)
  then
    show image 슬픔 at offset 0
  else
    show image 화난 at offset 0
  
```

Q. 표정을 두 가지 더 추가하려면 어떻게 해야 할까요?

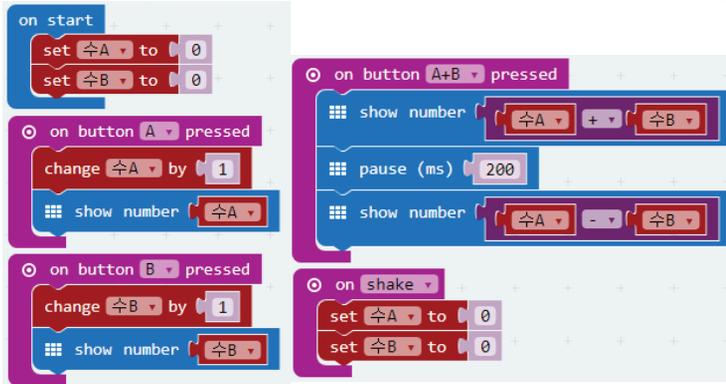
변수 이미지를 추가한 후 else if블록을 추가한다.

★블록의 의미를 써 봅시다.

<b>if □ then</b>	만약 □일 때 ~을 실행한다.
<b>if □ then else</b>	만약 □일 때 ~을 실행한다. □가 아닐 땐 ~를 실행한다.

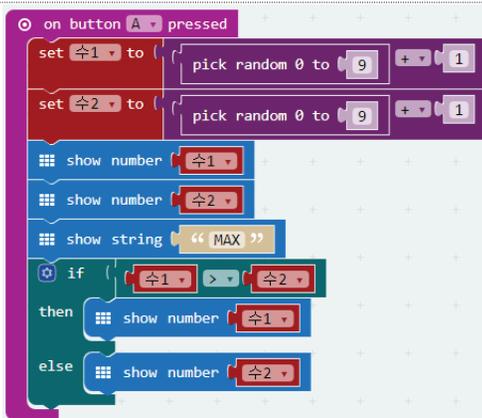
### ☆ 덧셈과 뺄셈

A버튼을 누르면 'A수'가 1씩 커지고, B버튼을 누르면 'B수'가 1씩 증가하며 A버튼과 B버튼을 동시에 누르면 'A수'와 'B수' 두 수에 대한 덧셈과 뺄셈을 하는 프로그램을 만들어 봅시다.



[미션⑨] '수A'와 '수B'의 곱셈과 나눗셈을 하는 프로그램을 만들어 보세요.

### ☆ 두 수 중 큰 수 찾기



★블록의 의미를 써 봅시다.

show string " "	" " 글씨를 보여준다. (" 안에 들어가는 글씨는 영어만 가능)
□ > □	부등호 표현(큰 수, 작은 수)

♥과제♥ 무작위 표현 & 수학적 연산 관련 자유 주제의 프로그램을 만드세요. (배운 것들 활용하여 만들기)

과제 계획:

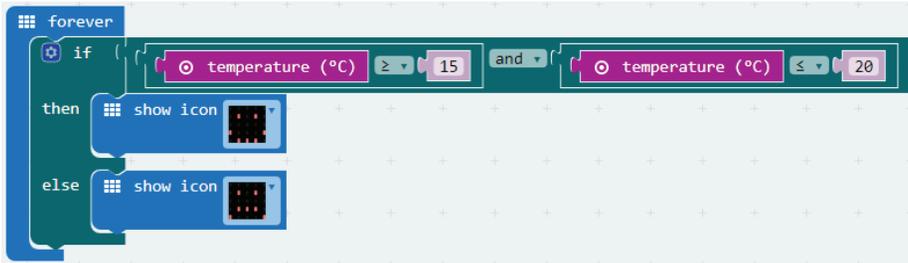
### <chapter 3. 기타기능 & 이미지 표현2>

마이크로비트에는 여러 가지 센서가 있습니다. 센서 및 기타 기능을 알아보고 새로운 이미지 표현 방법을 알아봅시다.

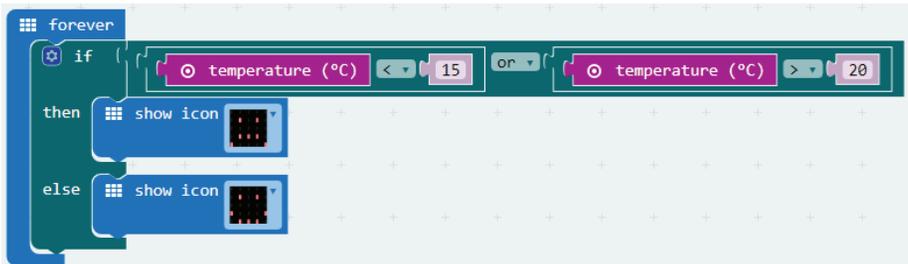
#### ⚙ 온도 체크기 만들기

15~20도는 웃는 표정, 그 외 온도는 찡그린 표정이 보이는 프로그램을 만들어 봅시다.

<방법1>



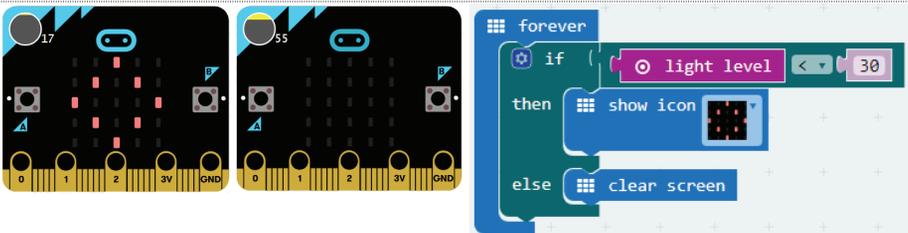
<방법2>



[미션@] 0~10도, 11~20도, 21~30도 각각의 온도마다 다른 이미지가 나오도록 하고, 현재 온도가 몇 도인지 알려주는 프로그램을 만드세요.

#### ⚙ 가로등 만들기

빛(light level)이 30 미만이면 LED가 켜지는 프로그램을 만들어 봅시다.



스위치 기능을 추가해 봅시다.

스위치 값이 off에서 on으로, on에서 off로 변경

★블록의 의미를 써 봅시다.

<b>temperature</b>	현재 온도를 나타낸다.
<b>light level</b>	빛의 밝기를 나타낸다.

### ✪ 반복해서 수 출력하기(for)

1~5까지 반복해서 출력하는 프로그램을 만들어 봅시다.

[미션①] for 블록을 이용하여 5, 4, 3, 2, 1을 반복해서 출력하는 프로그램을 만드세요.

[미션②] for 블록을 이용하여 2, 4, 6, 8을 반복해서 출력하는 프로그램을 만드세요.

### ☆ 반복해서 수 출력하기(while)

1~5까지 반복해서 출력하는 프로그램을 만들어 봅시다.

```
forever
  set a to 1
  while (a ≤ 5)
    do
      show number a
      pause (ms) 100
      change a by 1
```

[미션⑬] while블록을 이용하여 5, 4, 3, 2, 1을 반복해서 출력하는 프로그램을 만드세요.

[미션⑭] while블록을 이용하여 2, 4, 6, 8을 반복해서 출력하는 프로그램을 만드세요.

### ☆ 동작을 반복하다가 10초 후 멈추기(while 활용)

이미지가 움직이는 시간이 정해진 프로그램을 만들어 봅시다.

```
on start
  set time to 0
  forever
    while (time < 10)
      do
        show icon [Image 1]
        pause (ms) 200
        show icon [Image 2]
        pause (ms) 200
        change time by 2
    clear screen
```

❖ 변수를 만들어서 이미지 움직이기

```

on start
  set 우주선 to create image
  repeat 3 times
    do
      show image 우주선 at offset -4
      pause (ms) 300
      show image 우주선 at offset -3
      pause (ms) 300
      show image 우주선 at offset -2
      pause (ms) 300
      show image 우주선 at offset -1
      pause (ms) 300
      show image 우주선 at offset 0
      pause (ms) 300
      show image 우주선 at offset 1
      pause (ms) 300
      show image 우주선 at offset 2
      pause (ms) 300
      show image 우주선 at offset 3
      pause (ms) 300
      show image 우주선 at offset 4
      pause (ms) 300
  clear screen
  
```

1. 어떤 결과가 만들어지나요?

우주선이 왼쪽으로 이동하는 모양이 세 번 반복된다.

2. 세 번 반복되도록 하는 블록은 무엇일까요?

repeat 3 times  
do

3. show image 우주선 at offset블록에서 offset은 어떤 것을 의미할 것 같나요?

우주선의 중심이 있어야 할 자리

4. offset 자리를 표시해 봅시다.

		0	-1	

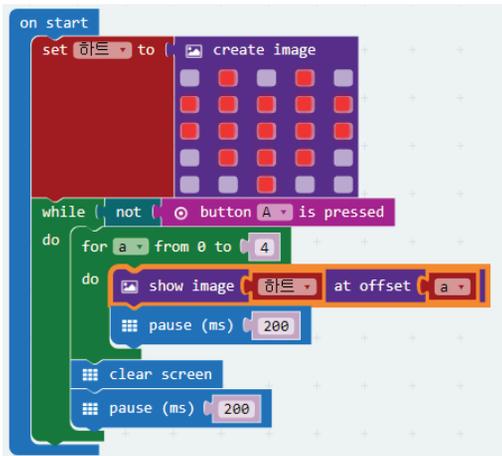
❖ 우주선 왼쪽으로 이동하기(while 활용)

```

on start
  set 우주선 to create image
  while (not button A is pressed)
    do
      for a from 0 to 8
        do
          show image 우주선 at offset a - 4
      clear screen
  
```

[미션⑥] 우주선을 오른쪽으로 이동시키는 프로그램을 만드세요.

[미션⑦] 다음 프로그램은 어떤 결과가 만들어질지 예상해 봅시다.



♥과제♥ 기타센서기능 & 이미지 표현 관련 자유 주제의 프로그램을 만드세요. (배운 것들 활용하여 만들기)

과제 계획:

<chapter 4. 이미지 표현 심화>

움직이는 LED를 더 다양한 방법으로 나타내고, 내가 원하는 대로 디자인하는 방법을 알아봅시다.

❁ LED한줄 켜기(plot블록 활용)

plot블록을 활용하여 LED의 불을 연속으로 켜고 끄는 프로그램을 만들어 봅시다.

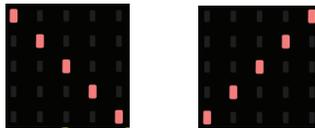
\* A 버튼 누르면

\* B 버튼 누르면

plot 블록

	0	1	2	3	4	(x)
0	(0,0)	(1,0)	(2,0)	(3,0)	(4,0)	
1	(0,1)	(1,1)	(2,1)	(3,1)	(4,1)	
2	(0,2)	(1,2)	(2,2)	(3,2)	(4,2)	
3	(0,3)	(1,3)	(2,3)	(3,3)	(4,3)	
4	(0,4)	(1,4)	(2,4)	(3,4)	(4,4)	
( y )						( x , y )

[미션⑤] A버튼을 누르면 왼쪽과 같이 LED가 차례대로 켜지고, B버튼을 누르면 오른쪽과 같이 LED가 켜지는 프로그램을 만드세요.



### ❖ 사각형 만들기

A버튼을 누르면 다음과 같은 순서로 LED가 켜지는 프로그램을 만들어 봅시다.

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

```

on button A pressed
  clear screen
  for a from 0 to 4
  do
    for b from 0 to 4
    do
      plot x a y b
      pause (ms) 100
  
```

[미션⑥] A버튼을 누르면 다음과 같은 순서로 LED가 켜지는 프로그램을 만들어 보세요.

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

### ❖ 삼각형 만들기

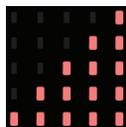
A 버튼 누르면 다음과 같은 모양으로 LED가 켜지는 프로그램을 만들어 봅시다.



```

on button A pressed
  clear screen
  for a from 0 to 4
  do
    for b from 0 to a
    do
      plot x b y a
      pause (ms) 100
  
```

[미션⑦] A 버튼 누르면 다음과 같은 모양으로 LED가 켜지는 프로그램을 만들어 봅시다.



☆ **≡자로 채우기**

A 버튼 누르면 다음과 같은 순서로 LED가 켜지는 프로그램을 만들어 봅시다.

1	2	3	4	5
10	9	8	7	6
11	12	13	14	15
20	19	18	17	16
21	22	23	24	25

```

on button A pressed
  clear screen
  for a from 0 to 4
  do
    for b from 0 to 4
    do
      if remainder of a ÷ 2 = 0
      then
        plot x b y a
      else
        plot x 4 - b y a
      pause (ms) 100
  
```

[미션⑧] A 버튼을 누르면 다음과 같은 순서로 LED가 켜지는 프로그램을 만드세요.

1	10	11	20	21
2	9	12	19	22
3	8	13	18	23
4	7	14	17	24
5	6	15	16	25

♥과제♥ 이미지 표현 심화(움직이는 LED, 디자인) 관련 자유 주제의 프로그램을 만드세요.  
(배운 것들 활용하여 만들기)

과제 계획:

<추가예제>

❁ 터지는 불꽃

불꽃이 터지는 모양이 반복되는 프로그램을 만들어 봅시다.

```
on start
  set a to 2
  set b to 2

forever
  plot x a y b
  for X from 0 to 2
    do
      set a to a + X
      set b to b - X
      plot x a y b
      plot x b y a
      plot x a y a
      plot x b y b
      pause (ms) 300
      set a to 2
      set b to 2
  for X from 0 to 2
    do
      set a to a + X
      plot x b y a
      plot x b y 4 - a
      plot x a y b
      plot x 4 - a y b
      pause (ms) 300
      set a to 2
  pause (ms) 300
  clear screen
```

### ❁ 다이아몬드 만들기

A 버튼을 누르면 다음과 같은 순서로 LED가 켜지는 프로그램을 만들어 봅시다.

		1		
	2	3	4	
5	6	7	8	9
	10	11	12	
		13		

2	0
1	1
2	1
3	1
0	2
1	2
2	2
3	2
4	2
1	3
2	3
3	3
2	4

```

on button A pressed
  set 시작 to 2
  set 끝 to 2
  for i from 0 to 4
    do
      set j to 시작
      while ( j ≤ 끝 )
        do
          plot x j y i
          pause (ms) 100
          change j by 1
        if ( i < 2 )
          then
            change 시작 by -1
            change 끝 by 1
          else
            change 시작 by 1
            change 끝 by -1
  
```

### ❁ 팔굽혀펴기 하는 사람

A버튼을 누르면 횟수가 올라가고, B버튼을 누르면 횟수가 내려갑니다. A+B버튼을 누르면 설정한 횟수를 보여줍니다. A버튼과 B버튼을 조작하여 횟수를 정하고, 마이크로비트를 흔들었을 때 정한 횟수만큼 팔굽혀펴기를 하는 이미지 나오는 프로그램을 만들어 봅시다.

```

on start
  set 횟수 to 0
  set 사람1 to create image
  set 사람2 to create image

on button A+B pressed
  clear screen
  show number 횟수

on button A pressed
  change 횟수 by 1

on button B pressed
  change 횟수 by -1

on shake
  clear screen
  repeat 횟수 times
    do
      show image 사람1 at offset 0
      pause (ms) 200
      show image 사람2 at offset 0
      pause (ms) 200
  
```

### ❁ 하트 모양 디자인

작은 하트에 불이 켜지고 작은 하트를 뺀 나머지에 불이 켜지는 반복되는 프로그램을 만들어 봅시다.

```

on button B pressed
  for a from 0 to 4
    do
      for b from 0 to 4
        do
          if point x b y a
            then
              unplot x b y a
            else
              plot x b y a
          pause (ms) 100
        
```

[부록 2]

일시	2017. 8. 14.		대상	마이크로비트 B반
주제	마이크로비트 기초기능, 이미지 표현		차시	1~8/48
활동명	마이크로비트의 기초 기능 익히기		소요 시간	400분
학습 목표	마이크로비트의 기초 기능을 알아보고 간단한 이미지 표현을 만들 수 있다.			
학습 단계	학습 요소	교수·학습 활동		시간
도입	사전 검사 및 오리엔테이션	<ul style="list-style-type: none"> <li>●사전검사 활동 및 오리엔테이션</li> <li>▶ 마이크로비트를 활용한 디자인 사고 중심 소프트웨어 (SW) 학습이 여러분의 창의성 향상에 영향을 미치는지 알아보기 위한 사전검사를 하겠습니다.</li> <li>- TTCT 검사</li> </ul>		50'
	동기 유발	<ul style="list-style-type: none"> <li>●동기유발</li> <li>▶ 마이크로비트 탐색하기</li> <li>▶ 마이크로비트는 작은 컴퓨터와 같습니다. 컴퓨터와 마이크로비트의 닮은 점을 찾아봅시다.</li> <li>- 마이크로비트의 네모 모양의 전구들이 모니터와 같은 역할을 할 것 같습니다.</li> <li>- 마이크로비트의 양 옆에 있는 누르는 부분이 키보드와 같은 역할을 할 것 같습니다.</li> <li>- 컴퓨터처럼 프로그램을 넣을 수 있을 것 같습니다.</li> </ul>		20'
	공부할 문제 확인 및 활동 안내	<ul style="list-style-type: none"> <li>●공부할 문제 확인</li> </ul> <div style="border: 1px solid black; padding: 5px; text-align: center;"> <b>마이크로비트의 기초기능을 익히고, 간단한 이미지 표현을 만들어 봅시다.</b> </div> <ul style="list-style-type: none"> <li>●활동 안내</li> <li>[활동1]마이크로비트의 기초 기능 익히기</li> <li>[활동2]움직이는 이미지 만들기</li> <li>[활동3]마이크로비트에 입력 받기</li> </ul>		
전개	기초 기능 익히기	<ul style="list-style-type: none"> <li>●[활동1]마이크로비트의 기초 기능 익히기</li> <li>▶ 마이크로비트의 구조 살펴보기</li> <li>▶ 마이크로비트 웹 브라우저 살펴보기</li> <li>▶ 마이크로비트 프로그래밍 방법 알아보기</li> <li>▶ 'show leds'블록을 활용해 마이크로비트 LED가 켜지는 프로그램 만들기</li> <li>▶ LED를 이용해 다양한 동물 이미지 만들기</li> <li>▶ 미션 해결하기</li> </ul>		60'
	움직임 이미지 프로그래밍	<ul style="list-style-type: none"> <li>●[활동2]움직이는 이미지 만들기</li> <li>▶ 이미지가 움직이는 것처럼 보이려면 어떤 방법으로 프로그래밍 해야 할까요?</li> <li>- 'show leds'블록을 여러 번 이어 붙입니다.</li> <li>- 다른 조건 블록을 써 봅니다.</li> <li>▶ 'show leds'블록을 여러 번 이어 붙여 움직이는 모양의 프로그램 만들기</li> <li>▶ 'forever'블록을 이용하여 움직이는 모양의 프로그램 만들기</li> <li>▶ 'while'블록을 이용하여 움직이는 모양의 프로그램 만들기</li> <li>▶ 'for'블록을 이용하여 움직이는 모양의 프로그램 만들기</li> <li>▶ 미션 해결하기</li> <li>▶ 블록 의미 정리하기</li> </ul>		90'
	입력 받기	<ul style="list-style-type: none"> <li>●[활동3]마이크로비트에 입력받기</li> </ul>		90'

	프로그램 명	<ul style="list-style-type: none"> <li>▶ 버튼을 눌러 투표하는 영상 보여주기</li> <li>▶ 마이크로비트의 버튼을 사용하여 사용자의 입력을 받을 수 있는 프로그램 만들기</li> <li>▶ 마이크로비트의 흔들기 인식 기능을 사용하여 프로그램 만들기</li> <li>▶ 변수를 이용한 프로그램 만들기</li> <li>▶ 변수를 만들어서 이미지 바꾸기</li> <li>▶ 변수를 이용하여 버튼 카운터 만들기</li> <li>▶ 변수를 이용하여 흔들기 카운터 만들기</li> <li>▶ 블록 의미 정리하기</li> </ul>	
정리	정리 및 과제 제시	<ul style="list-style-type: none"> <li>●정리하기</li> <li>▶ 팀 미션 수행하기</li> <li>●과제 제시</li> <li>▶ 마이크로비트의 기초 기능과 이미지와 관련된 자유 주제의 프로그램 만들기(배운 것 활용하여 만들기)</li> </ul>	90'

일시	2017. 8. 15.		대상	마이크로비트B반
주제	무작위 표현, 연산		차시	9~16/48
활동명	무작위 수를 출력하는 방법 및 수학적 연산 방법 알기	소요시간	400분	
학습목표	무작위 수를 출력하는 방법과 여러 가지 수학적 연산 방법을 알 수 있다.			
학습 단계	학습 요소	교수·학습 활동		시간
도입	전시 학습 상기	<ul style="list-style-type: none"> <li>●복습하기</li> <li>▶ 지난 시간에 배웠던 내용을 말해 봅시다. <ul style="list-style-type: none"> <li>- 마이크로비트의 기초기능들을 익혔습니다.</li> <li>- 이미지를 표현하는 방법을 배웠습니다.</li> </ul> </li> <li>▶ 지난 시간에 배웠던 내용을 복습해 봅시다.</li> </ul>		20'
	동기 유발	<ul style="list-style-type: none"> <li>●동기유발</li> <li>▶ 추첨 프로그램으로 번호 추첨하기 <ul style="list-style-type: none"> <li>- 마이크로비트로 추첨 프로그램을 만들 수 있을까요?</li> </ul> </li> <li>▶ 계산기 원리 생각해보기 <ul style="list-style-type: none"> <li>- 우리가 사용하는 계산기는 어떤 원리로 만들어졌을지 생각해 봅시다.</li> <li>- 수를 입력받아 계산하도록 프로그래밍 했을 것 같습니다.</li> <li>- 계산 프로그램을 계산기에 넣었을 것 같습니다.</li> </ul> </li> </ul>		20'
	공부할 문제 확인 및 활동 안내	<ul style="list-style-type: none"> <li>●공부할 문제 확인</li> <li style="text-align: center; border: 1px solid black; padding: 5px;"><b>마이크로비트로 무작위 수를 추출하는 방법과 여러 가지 수학적 연산의 방법을 알아봅시다.</b></li> <li>●활동 안내</li> <li>[활동1] 무작위 수 추출하기</li> <li>[활동2] 무작위로 이미지 표현하기</li> <li>[활동3] 연산 프로그램 만들기</li> </ul>		
전개	무작위 수 추출 프로그램 명	<ul style="list-style-type: none"> <li>●[활동1] 무작위 수 추출하기</li> <li>▶ 'pick random' 블록을 정해놓은 수 범위 안에서 한 수를 무작위로 뽑는 프로그램 만들기</li> <li>▶ 하나의 LED 불빛이 다른 무작위의 좌표로 이동하는 프로그램 만들기</li> <li>▶ 블록 의미 정리하기</li> </ul>		90'
	무작위 이미지	<ul style="list-style-type: none"> <li>●[활동2] 무작위로 이미지 표현하기</li> <li>▶ 무작위로 이미지 표현하기</li> </ul>		90'

	표현 프로그램밍	<ul style="list-style-type: none"> <li>▶ 정해놓은 이미지 중 무작위로 하나의 이미지가 나오도록 프로그래밍을 해 봅시다.</li> <li>▶ 이미지를 더 추가하려면 어떻게 해야 할까요? - 변수 이미지를 추가한 후 'else if'블록을 추가합니다.</li> <li>▶ 블록 의미 정리하기</li> </ul>	
	연산 프로그램밍	<ul style="list-style-type: none"> <li>●[활동3]연산 프로그램 만들기</li> <li>▶ 연산 블록을 사용하여 덧셈과 뺄셈 프로그램 만들기</li> <li>▶ 미션 해결하기</li> <li>▶ 두 수 중 큰 수를 찾는 프로그램 만들기</li> <li>▶ 블록 의미 정리하기</li> </ul>	90'
정리	정리 및 과제 제시	<ul style="list-style-type: none"> <li>●정리하기</li> <li>▶ 팀 미션 수행하기</li> <li>●과제 제시</li> <li>▶ 무작위 표현과 수학적 연산 관련 자유 주제의 프로그램 만들기</li> </ul>	90'

일시	2017. 8. 16.		대상	마이크로비트 B반
주제	온도 센서, 빛 센서, 반복표현, 이미지 움직이기		차시	17~24/48
활동명	마이크로비트의 다양한 기능과 이미지 표현 알아보기		소요 시간	400분
학습 목표	마이크로비트의 여러 가지 기능을 알아보고 다양한 방법으로 이미지 표현을 할 수 있다.			
학습 단계	학습 요소	교수·학습 활동		시간
도입	전시 학습 상기	<ul style="list-style-type: none"> <li>●복습하기</li> <li>▶ 지난 시간에 배웠던 내용을 말해 봅시다. - 수나 이미지를 무작위로 뽑도록 하는 방법에 대해 배웠습니다.</li> <li>- 마이크로비트로 덧셈, 뺄셈, 큰 수 찾기 프로그램을 만들어 보았습니다.</li> <li>▶ 지난 시간에 배웠던 내용을 복습해 봅시다.</li> </ul>		20'
	동기 유발	<ul style="list-style-type: none"> <li>●동기유발</li> <li>▶ 여러 가지 센서를 이용한 물건 사진보기</li> <li>▶ 사진 속 물건이 어떤 기능을 하는지 발표하기 - 온도가 낮으면 난방이 켜집니다. - 어두워지면 가로등이 켜집니다.</li> <li>▶ 사진 속 물건이 제 기능을 하기 위해 어떤 방법을 사용했는지 토의하기 - 빛의 양을 측정하여 빛의 양이 줄어들면 불이 켜지도록 했을 것 같습니다. - 온도계를 이용했을 것 같습니다.</li> <li>▶ 사진 속 물건들은 같은 패턴의 일을 반복합니다. 같은 일을 반복하도록 프로그래밍 하려면 어떤 방법을 사용하면 좋을까요? - 원하는 기능을 프로그래밍한 뒤 그 부분을 복사하여 여러 번 이어 붙입니다.</li> </ul>		20'
	학습문제 제시 및 공부할 문제 확인	<ul style="list-style-type: none"> <li>●학습문제 제시</li> </ul> <div style="border: 1px solid black; padding: 5px; text-align: center;"> <b>마이크로비트의 여러 가지 기능을 알아보고 다양한 방법으로 이미지 표현을 해 봅시다.</b> </div> <ul style="list-style-type: none"> <li>●활동 안내</li> <li>[활동1]온도센서를 이용한 온도 체크기 만들기</li> <li>[활동2]빛 센서를 이용한 가로등 만들기</li> </ul>		

		[활동3]여러 가지 방법으로 이미지 움직이기	
전개	온도센서 활용 프로그래밍	<ul style="list-style-type: none"> <li>●[활동1]온도센서를 이용한 온도 체크기 만들기 <ul style="list-style-type: none"> <li>▶ 온도센서를 이용한 온도 체크기 시연해보기</li> <li>▶ 온도 체크기의 원리가 무엇인지 생각해보기 <ul style="list-style-type: none"> <li>- 마이크로비트에 온도를 감지하는 센서가 있는 것 같습니다.</li> <li>- if문으로 조건을 넣어 온도를 감지하도록 했을 것 같습니다.</li> </ul> </li> <li>▶ 마이크로비트 온도 센서 확인해보기</li> <li>▶ 두 가지 방법으로 온도 체크기 만들기</li> <li>▶ 미션 해결하기(조건 추가)</li> </ul> </li> </ul>	90'
	빛 센서 활용 프로그래밍	<ul style="list-style-type: none"> <li>●[활동2]빛 센서를 이용한 가로등 만들기 <ul style="list-style-type: none"> <li>▶ 빛 센서를 이용한 가로등(마이크로비트) 시연해보기</li> <li>▶ 가로등의 원리가 무엇인지 생각해보기 <ul style="list-style-type: none"> <li>- 마이크로비트에 빛을 감지하는 센서가 들어있고 그 센서기능을 이용해 조건문을 넣어 프로그래밍 했을 것 같습니다.</li> <li>- 불이 켜지고 꺼지는 빛의 양을 정해놓는 프로그램이 적용됐을 것 같습니다.</li> </ul> </li> <li>▶ 마이크로비트 빛 센서 확인해보기</li> <li>▶ 가로등 만들기</li> <li>▶ 미션 해결하기(스위치 기능 추가)</li> <li>▶ 블록 의미 정리하기 <ul style="list-style-type: none"> <li>- temperature, light level</li> </ul> </li> </ul> </li> </ul>	90'
	이미지 움직임 프로그래밍	<ul style="list-style-type: none"> <li>●[활동3]여러 가지 방법으로 이미지 움직이기 <ul style="list-style-type: none"> <li>▶ 반복해서 수 출력하기</li> <li>▶ 미션 해결하기</li> <li>▶ While문을 이용한 반복해서 수 출력하기</li> <li>▶ 미션 해결하기</li> <li>▶ 동작을 반복하다 10초 후 멈추기</li> <li>▶ 변수를 만들어서 이미지 움직이기</li> <li>▶ 미션 해결하기</li> </ul> </li> </ul>	90'
정리	정리 및 과제 제시	<ul style="list-style-type: none"> <li>●정리하기 <ul style="list-style-type: none"> <li>▶ 팀 미션 수행하기</li> </ul> </li> <li>●과제 제시 <ul style="list-style-type: none"> <li>▶ 여러 가지 센서 기능과 이미지 표현 관련 자유 주제 프로그램 만들기(배운 것 활용)</li> </ul> </li> </ul>	90'

일시	2017. 8. 17.		대상	마이크로비트 B반
주제	이미지표현 심화		차시	25~32/48
활동명	다양한 방법으로 움직이는 이미지 나타내기		소요 시간	400분
학습 목표	움직이는 이미지를 더 다양한 방법으로 나타내는 방법을 알 수 있다.			
학습 단계	학습 요소	교수·학습 활동		시간
도입	전시 학습 상기	<ul style="list-style-type: none"> <li>●복습하기 <ul style="list-style-type: none"> <li>▶ 지난 시간에 배웠던 내용을 말해 봅시다. <ul style="list-style-type: none"> <li>- 마이크로비트의 여러 가지 센서를 이용한 프로그램을 만들어 보았습니다.</li> <li>- 이미지 표현 방법을 배웠습니다.</li> </ul> </li> <li>▶ 지난 시간에 배웠던 내용을 복습해 봅시다.</li> </ul> </li> </ul>		20'
	동기 유발	●동기유발		20'

		<ul style="list-style-type: none"> <li>▶ 전광판 보여주기</li> <li>▶ 일정한 방향과 모양으로 이미지가 나타나는 전광판은 어떻게 만들었는지 발표해 봅시다.</li> <li>- 전광판 LED의 각 좌표 값을 계산하여 불이 차례로 켜지도록 프로그램을 만들었을 것 같습니다.</li> <li>- 하나의 이미지를 만들어 놓고 원하는 방향으로 이동하도록 프로그램을 만들었을 것 같습니다.</li> </ul>	
	학습문제 제시 및 활동 안내	<ul style="list-style-type: none"> <li>●학습문제 제시</li> </ul> <div style="border: 1px solid black; padding: 5px; text-align: center;"> <b>움직이는 이미지를 더 다양한 방법으로 나타내는 방법을 알아봅시다.</b> </div> <ul style="list-style-type: none"> <li>●활동 안내</li> <li>[활동1]LED한 줄을 특정 방향으로 켜 보기</li> <li>[활동2]LED를 도형 모양으로 켜 보기</li> <li>[활동3]LED를 원하는 순서대로 채우기</li> </ul>	
전개	LED로 방향 그리기 프로그래밍	<ul style="list-style-type: none"> <li>●[활동1]LED한 줄을 특정 방향으로 켜 보기</li> <li>▶ 'plot'블록을 활용하여 LED를 연속으로 켜고 끄는 프로그램 만들기</li> <li>▶ LED좌표 알아보기</li> <li>▶ 5X5로 구성된 LED는 좌표를 갖습니다. 좌표를 x좌표와 y좌표로 설정하였을 때 가장 왼쪽의 가장 위 좌표가 (0,0)이 됩니다. 그렇다면 두 번째 줄의 가장 오른쪽 좌표는 어떻게 될까요?</li> <li>- (4,0)입니다.</li> <li>▶ 'plot'블록을 이용하여 원하는 좌표의 LED를 켜 봅시다.</li> <li>▶ 미션 해결하기</li> </ul>	90'
	LED로 도형 그리기 프로그래밍	<ul style="list-style-type: none"> <li>●[활동2]LED를 도형 모양으로 켜 보기</li> <li>▶ LED좌표를 고려하여 사각형 모양이 그려지려면 어떤 좌표의 LED가 켜져야 할지 생각해 봅시다.</li> <li>▶ A버튼을 누르면 사각형 모양으로 LED가 켜지는 프로그램 만들기</li> <li>▶ 미션 해결하기</li> <li>▶ A버튼을 누르면 삼각형 모양으로 LED가 켜지는 프로그램 만들기</li> <li>▶ 미션 해결하기</li> </ul>	90'
	LED 순서 정하기 프로그래밍	<ul style="list-style-type: none"> <li>●[활동3]LED를 원하는 순서대로 채우기</li> <li>▶ A버튼을 누르면 주어진 순서대로 LED가 켜지는 프로그램 만들기</li> <li>▶ 미션 해결하기</li> </ul>	90'
정리	정리 및 과제 제시	<ul style="list-style-type: none"> <li>●정리하기</li> <li>▶ 팀 미션 수행하기</li> <li>●과제 제시</li> <li>▶ 움직이는 이미지(LED 좌표 활용) 관련 자유 주제 프로그램 만들기</li> <li>▶ 프로젝트 발표 주제 생각해오기</li> </ul>	90'

일시	2017. 8. 18.	대상	마이크로비트B반
주제	프로젝트 발표 준비	차시	33~40/48
활동명	프로젝트 계획서 작성 및 프로그램 만들기		400분
학습 목표	프로젝트 계획서를 작성하고 계획에 맞는 프로그램을 만들 수 있다.		

학습 단계	학습 요소	교수·학습 활동	시간
도입	전시 학습 상기	<ul style="list-style-type: none"> <li>●복습하기               <ul style="list-style-type: none"> <li>▶ 지난 시간에 배웠던 내용을 말해 봅시다.                   <ul style="list-style-type: none"> <li>- LED가 규칙적으로 켜지도록 프로그램을 만들어 보았습니다.</li> <li>- 이미지 표현 방법을 배웠습니다.</li> </ul> </li> <li>▶ 지난 시간에 배웠던 내용을 복습해 봅시다.</li> </ul> </li> </ul>	40'
	동기 유발	<ul style="list-style-type: none"> <li>●동기유발               <ul style="list-style-type: none"> <li>▶ 교사가 만든 마이크로비트 프로그램과 계획서를 보여준다.</li> <li>▶ 프로젝트를 계획할 때 고려해야 할 부분은 무엇이 있는지 탐색하기                   <ul style="list-style-type: none"> <li>- 마이크로비트 프로그램 제작 및 프로젝트 발표를 위해서는 계획서가 잘 작성되어야 함을 인식한다.</li> <li>- 계획서에 꼭 넣어야 할 사항들에 대하여 다시 점검한다.</li> </ul> </li> <li>▶ 모둠이 모여 프로젝트 주제를 정하고 개별적으로 만들 프로그램을 정해 계획해 봅시다.                   <ul style="list-style-type: none"> <li>- 모둠이 모여 프로젝트로 만들 주제를 선택하여 그 주제의 문제를 해결하기 위한 방법을 브레인스토밍 한다.</li> </ul> </li> </ul> </li> </ul>	20'
	공부할 문제 확인 및 활동 안내	<ul style="list-style-type: none"> <li>●공부할 문제 확인               <div style="border: 1px solid black; padding: 5px; text-align: center;"> <b>학습한 내용을 바탕으로 프로젝트 계획서를 작성하고 계획한 프로그램을 만들어 봅시다.</b> </div> <li>●활동 안내               <ul style="list-style-type: none"> <li>[활동1]프로젝트 계획서 작성하기</li> <li>[활동2]프로젝트 프로그램 제작하기</li> </ul> </li> </li></ul>	
전개	프로젝트 계획서 만들기	<ul style="list-style-type: none"> <li>●[활동1]프로젝트 계획서 작성하기               <ul style="list-style-type: none"> <li>▶ 프로젝트 계획서 작성하기</li> <li>▶ 프로젝트 계획서 피드백                   <ul style="list-style-type: none"> <li>- 작성한 프로젝트 계획서를 친구, 선생님과 함께 보완한다.</li> <li>- 잘못된 명령이 있는지 확인한다.</li> <li>- 게임의 기능들이 잘 활용되었는지 확인한다.</li> </ul> </li> </ul> </li> </ul>	60'
	프로젝트 프로그램 제작하기	<ul style="list-style-type: none"> <li>●[활동2]프로젝트 프로그램 제작하기               <ul style="list-style-type: none"> <li>▶ 그 동안 배운 내용 활용하여 프로젝트 프로그램을 제작한다.</li> <li>▶ 프로젝트 프로그램 피드백                   <ul style="list-style-type: none"> <li>- 친구, 선생님과 함께 만든 프로그램을 보완한다.</li> <li>- 프로그램을 만들면서 궁금한 점을 선생님께 피드백 받는다.</li> </ul> </li> </ul> </li> </ul>	240'
정리	정리 및 과제 제시	<ul style="list-style-type: none"> <li>●정리하기               <ul style="list-style-type: none"> <li>▶ 발표를 멋지게 하는 방법 브레인스토밍하기                   <ul style="list-style-type: none"> <li>- 프로젝트 제작 발표회에서 효과적으로 발표하기 위한 방법 탐색</li> </ul> </li> </ul> </li> <li>●과제 제시               <ul style="list-style-type: none"> <li>▶ 프로젝트 프로그램 마무리 및 발표 연습하기</li> </ul> </li> </ul>	40'

일시	2017. 8. 19.	대상	마이크로비트B반
주제	프로젝트 작품 발표하기		41~48/48
활동명	마이크로비트 프로그램 제작 발표회	소요 시간	400분

학습 목표		자신이 제작한 마이크로비트 프로그램을 발표할 수 있다.	
학습 단계	학습 요소	교수·학습 활동	시간
도입	동기 유발	<ul style="list-style-type: none"> <li>●동기유발</li> <li>▶ 실제 게임 제작 발표회의 영상 시청</li> <li>- 영상을 보며 자신의 발표 내용을 점검한다.</li> </ul>	20'
	공부할 문제 확인 및 활동 안내	<ul style="list-style-type: none"> <li>●공부할 문제 확인</li> <li style="text-align: center; border: 1px solid black; padding: 5px;"><b>자신이 제작한 마이크로비트 프로그램을 발표하여 봅시다.</b></li> <li>●활동 안내</li> <li>[활동1]사후검사 활동</li> <li>[활동2]마이크로비트 프로그램 제작 발표회</li> </ul>	
전개	사후 검사	<ul style="list-style-type: none"> <li>●[활동1]사후검사 활동</li> <li>▶ 마이크로비트를 활용한 디자인 사고 중심 SW학습을 통하여 여러분의 컴퓨팅사고력과 창의성에 변화가 생겼는지 확인하는 검사를 하겠습니다.</li> <li>- TTCT 검사</li> </ul>	60'
	제작한 프로그램 발표하기	<ul style="list-style-type: none"> <li>●[활동2]마이크로비트 프로그램 제작 발표회</li> <li>▶ 부모님들을 모시고 자신이 제작한 프로그램에 대해 발표합니다.</li> <li>- 자신이 제작한 프로그램의 주제와 제작 원리, 사용 방법에 대하여 발표한다.</li> <li>▶ 교사는 발표한 내용에 대하여 즉시즉시 피드백 해준다.</li> </ul>	300'
정리	정리 하기	<ul style="list-style-type: none"> <li>●정리하기</li> <li>▶ 궁금한 점 질문</li> <li>- 마이크로비트 프로그램 제작 발표회에서 궁금했던 점에 대하여 질문한다.</li> <li>▶ 앞으로 SW의 인재가 되기 위한 방법에 대한 논의, 서로에 대하여 격려하기</li> </ul>	20'