

Computer Graphics < Synthesizer Simulation >

Team Project by. 군고구마

1. Subject

Synthesizer Simulation

방 안의 Synthesizer를 구현하고, 입력장치 (마우스, 키보드)를 이용하여 연주 및 시연

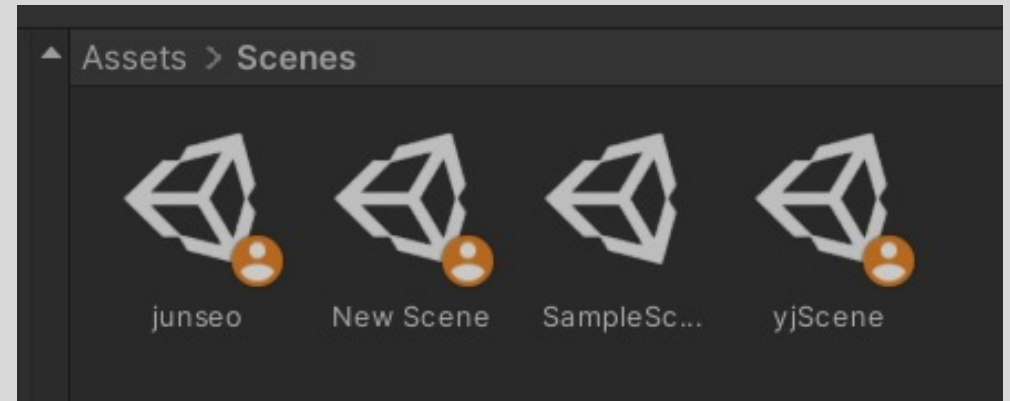
2. Platform

-Visual Studio, Live Share

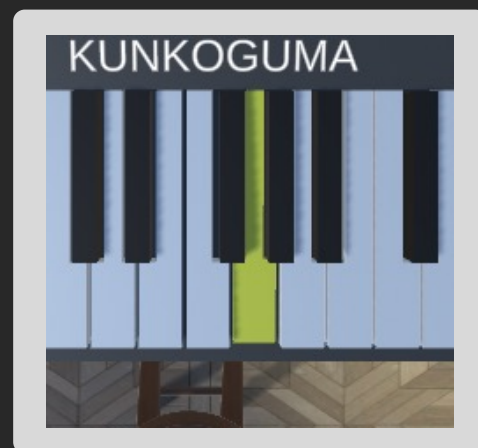
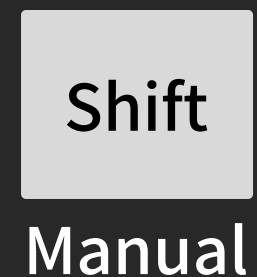
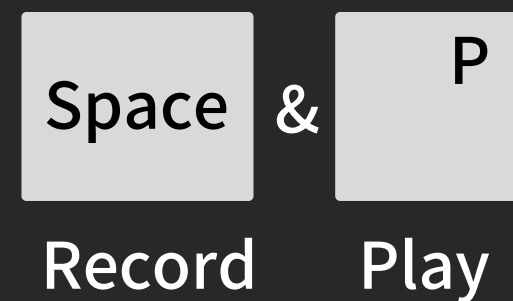
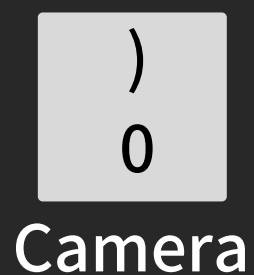
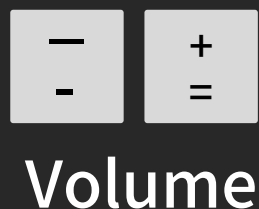
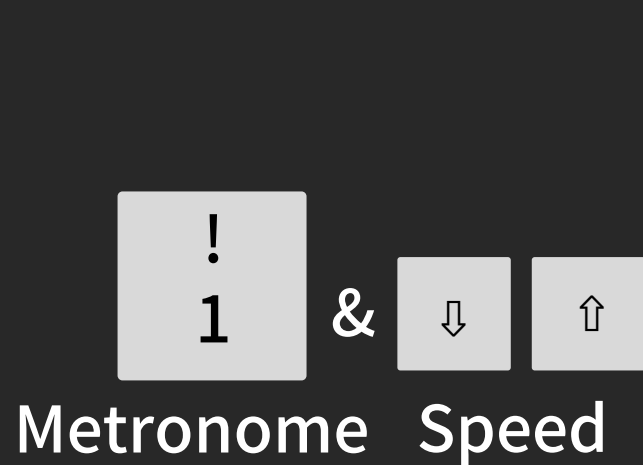
```
GameObject.Find("rec").GetComponent<Light>().enabled = true;
//GameObject.Find("redLight").GetComponent<Renderer>().material.color = Color.red;
Debug.Log("Recorder turn on");
고윤정: _onoff = 1;
}
else if (Rec_onoff == 1)
{
    GameObject.Find("rec").GetComponent<Light>().enabled = false;
    //GameObject.Find("redLight").GetComponent<Renderer>().material.color = Color.red;
    Debug.Log("Recorder turn off");
    Rec_onoff = 0; // turn off
}

if (Rec_onoff == 1) //녹음 진행 중
{
    if (Input.GetMouseButtonDown(0))
    {
        Debug.Log("clicked");
        Ray ray = Camera.main.ScreenPointToRay(Input.mousePosition);
```

-Unity, Collaborate

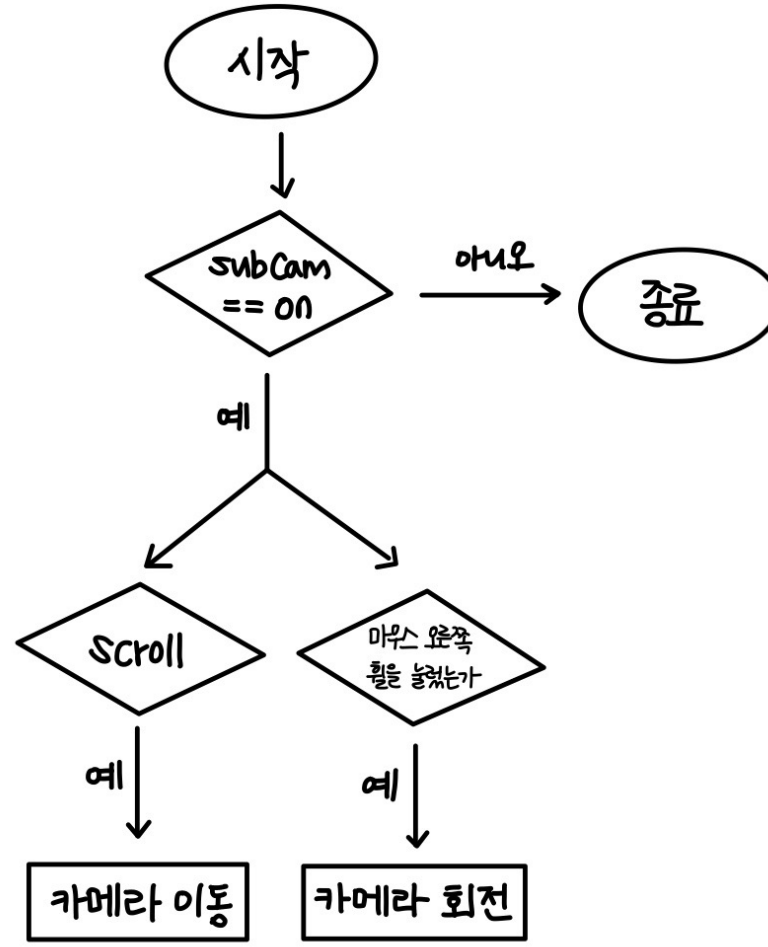
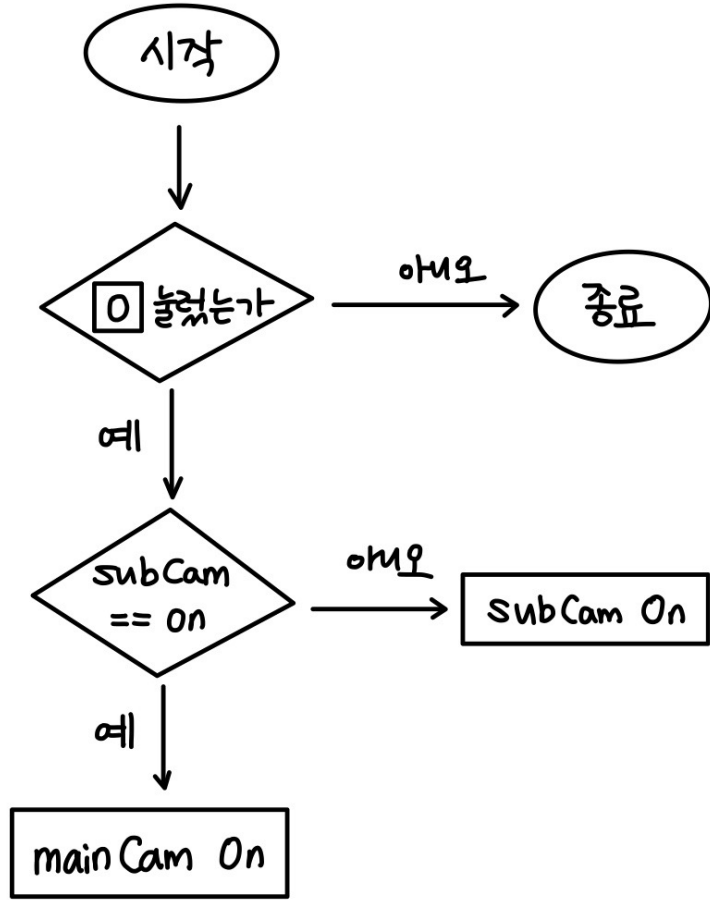


3. Algorithm

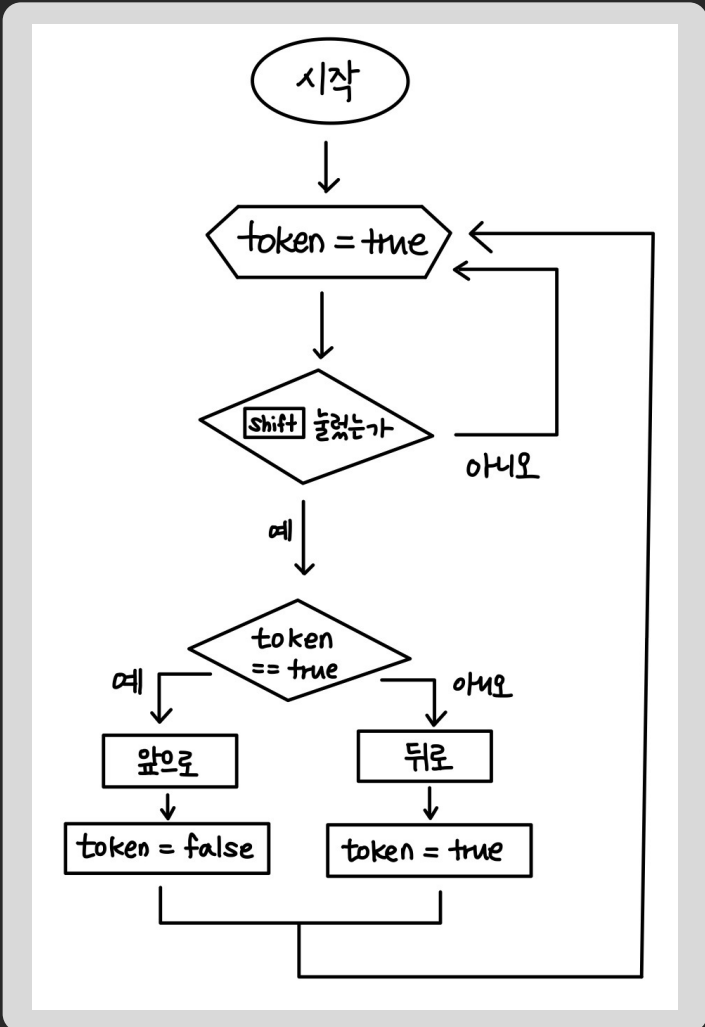


Play with Mouse & Keyboard!

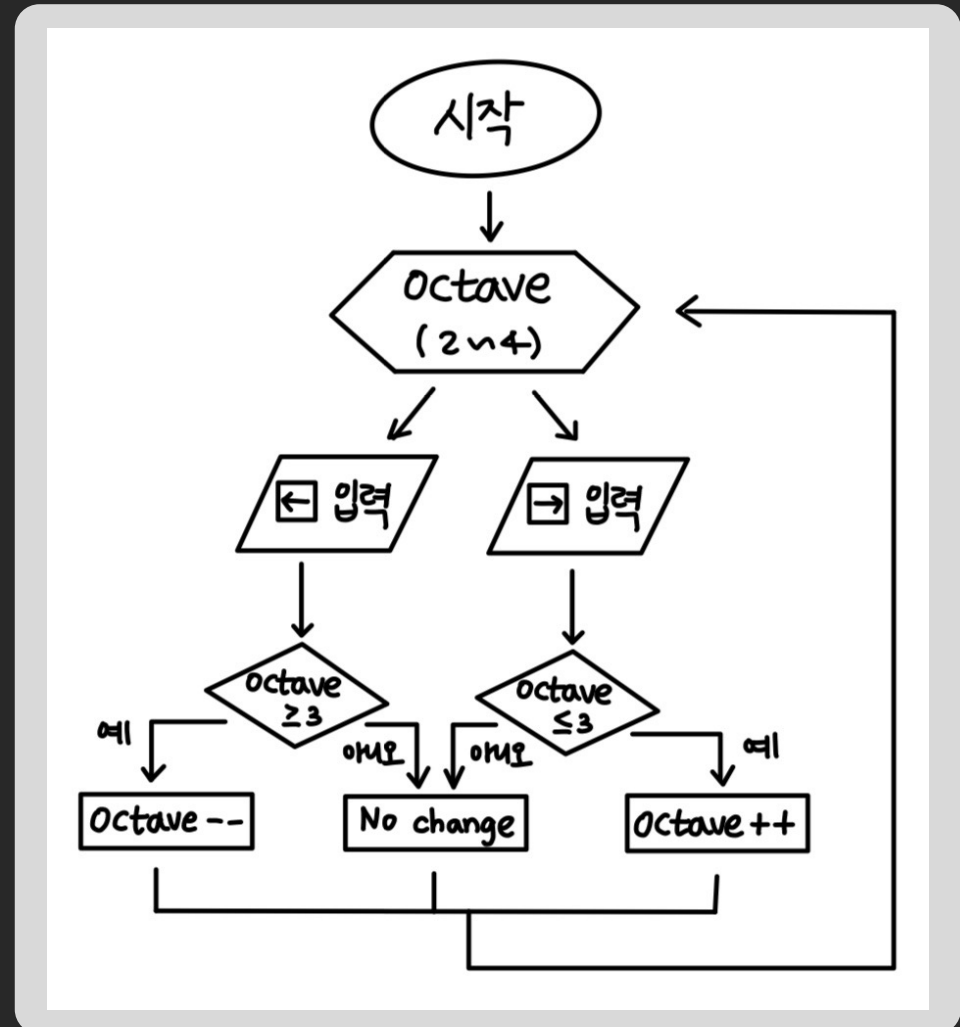
3-1. Algorithm / Camera



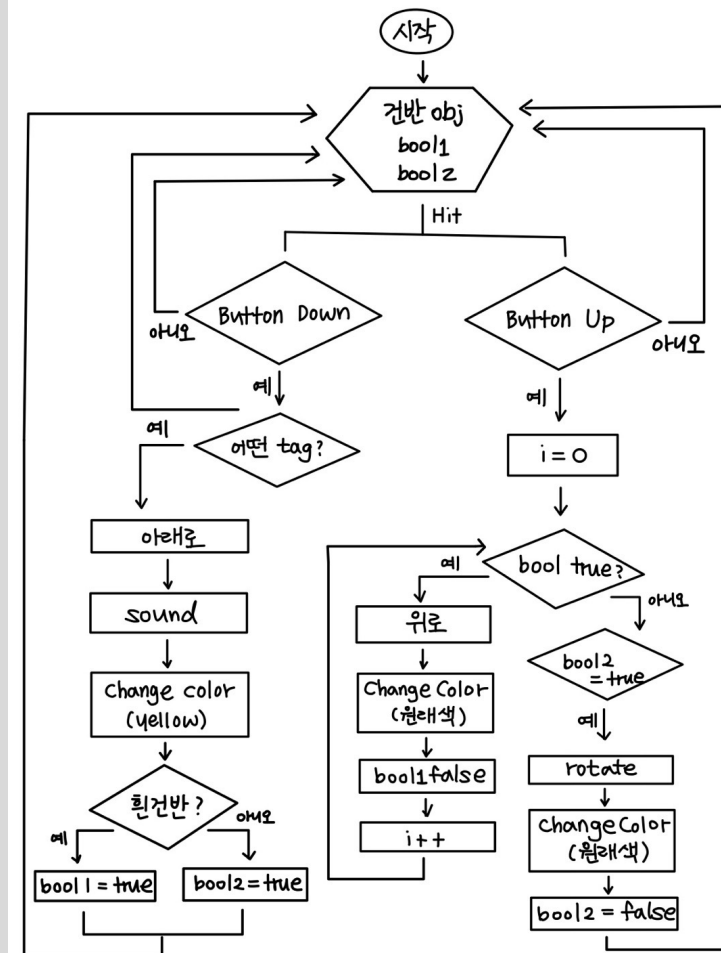
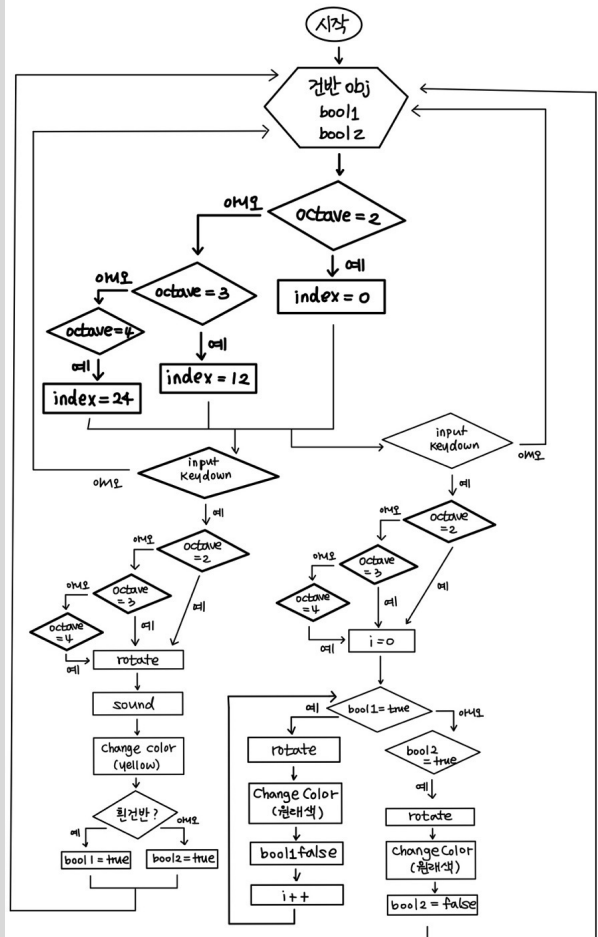
3-2. Algorithm / Manual



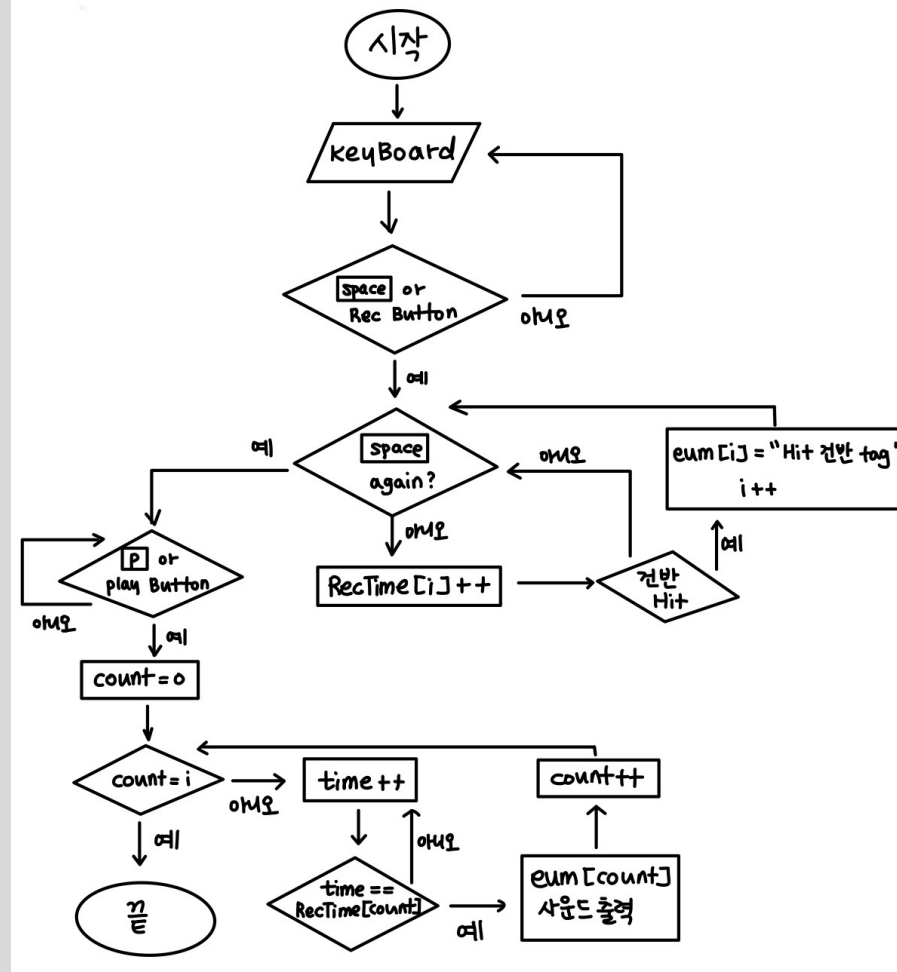
3-3. Algorithm / Octave



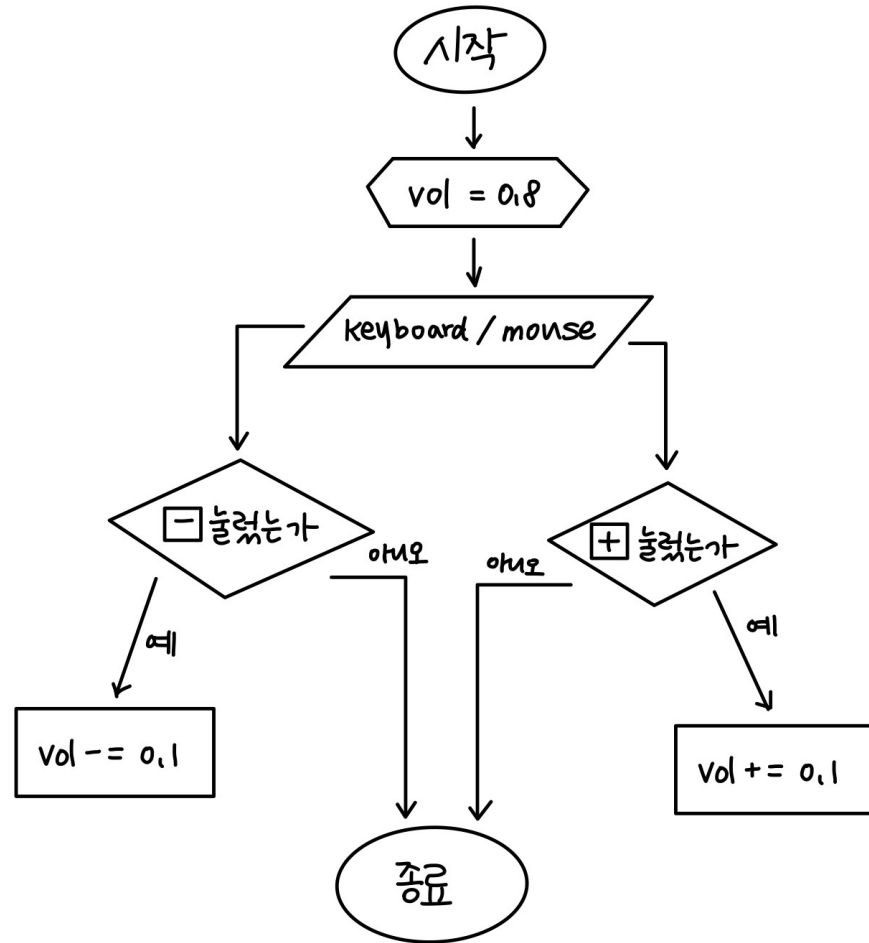
3-4. Algorithm / Play the Synthesizer (KeyBoard & Mouse)



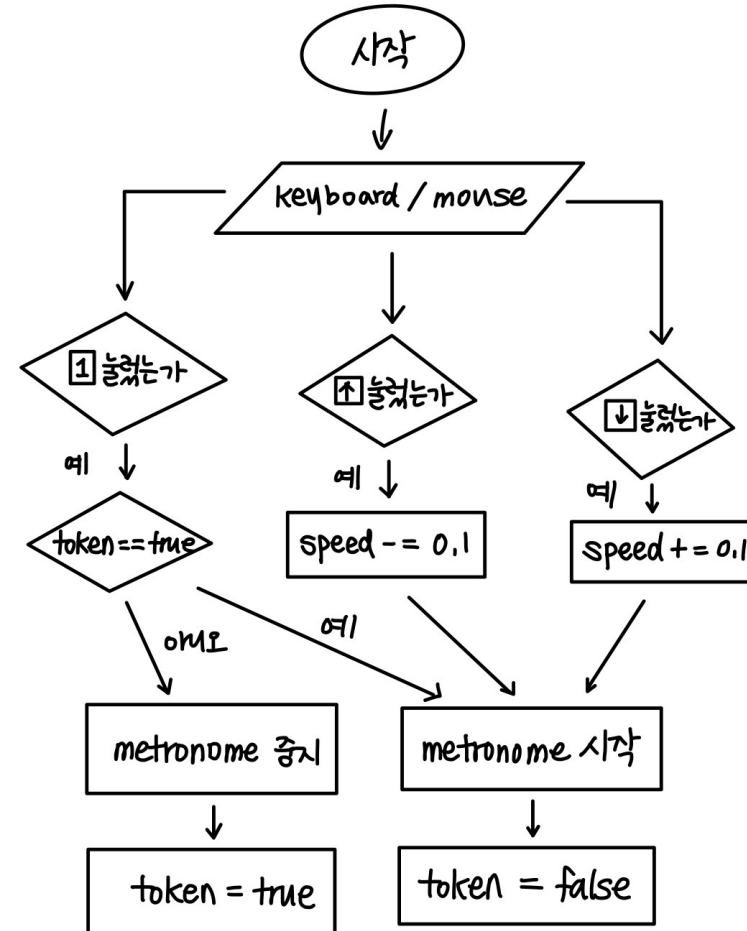
3-5. Algorithm / Recording



3-6. Algorithm / Volume

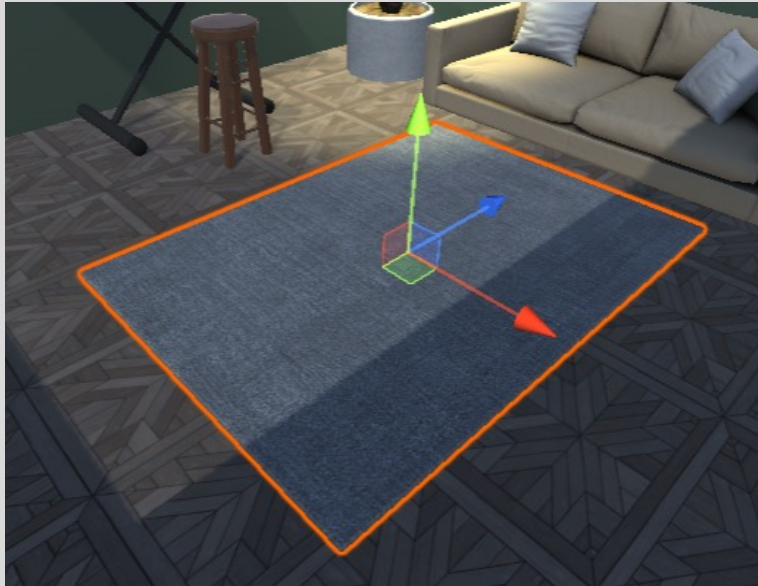
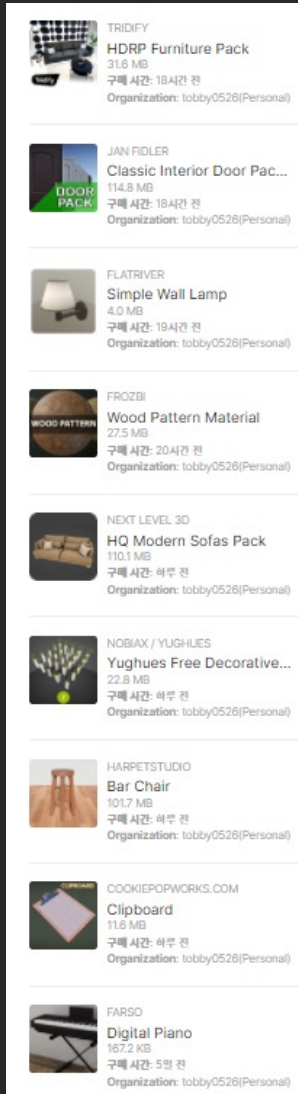


3-7. Algorithm / Metronome



4. External Reference

9 Assets from Unity Asset Store



4. External Reference

Classic Interior Door Pack 1

[Jan Fidler](#)

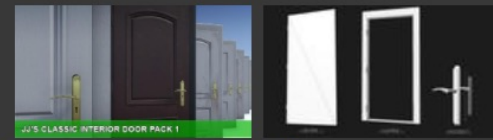
Version 1.0 - June 01, 2018 [asset store](#)

[View in the Asset Store](#) • [Publisher Website](#)

4 unique doors
11 unique variations
11 prefabs

[More...](#)

Images & Videos



[View images & videos on Asset Store](#)

Clipboard

[cookiepopworks.com](#)

Version 1.0 - January 29, 2019 [asset store](#)

[View in the Asset Store](#) • [Publisher Website](#) • [Publisher Support](#)

A simple clipboard.
(Notes: This product uses the Unity Standard Shaders)

[More...](#)

Images & Videos



[View images & videos on Asset Store](#)

HQ Modern Sofas Pack

[Next Level 3D](#)

Version 1.1 - February 07, 2017 [asset store](#)

[View in the Asset Store](#) • [Publisher Website](#)

HQ Modern sofas pack contains 5 high quality models of modern style sofas with color customization.

Also check out my new asset - HQ Modern Classic Interior

[More...](#)

Images & Videos



[View images & videos on Asset Store](#)

Package Size

Size: 110.11 MB (Number of files: 36)

Supported Unity Versions

5.5.0 or higher

Simple Wall Lamp

[FlatRiver](#)

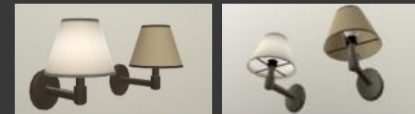
Version 1.0 - August 20, 2016 [asset store](#)

[View in the Asset Store](#) • [Publisher Website](#) • [Publisher Support](#)

Lowpoly model of a simple wall lamp
Amount of tris: 548
2 materials:

[More...](#)

Images & Videos



[View images & videos on Asset Store](#)

Package Size

Size: 3.96 MB (Number of files: 8)

Supported Unity Versions

5.0.0 or higher

Bar Chair

[HarpetStudio](#)

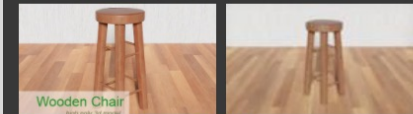
Version 1.0 - January 17, 2018 [asset store](#)

[View in the Asset Store](#) • [Publisher Website](#)

Free demo add-on for Kitchen Asset (<http://u3d.as/Yjw>). The package contains only one object - a chair for the kitchen mini bar. Performed in modern style and is a high-polygonal and high-quality product.

[More...](#)

Images & Videos



[View images & videos on Asset Store](#)

Package Size

Size: 101.69 MB (Number of files: 30)

Supported Unity Versions

5.6.0 or higher

Wood Pattern Material

[FrOzBi](#)

Version 1.0 - June 26, 2020 [asset store](#)

[View in the Asset Store](#) • [Publisher Website](#)

Wood pattern tillable material.
This Material is a part of -
Wood Patterns Material Pack

[More...](#)

Images & Videos



[View images & videos on Asset Store](#)

Package Size

Size: 27.53 MB (Number of files: 12)

Supported Unity Versions

2018.4.23 or higher

Yughues Free Decorative Plants

[NobiAx / Yughues](#)

Version 1.0 - March 28, 2015 [asset store](#)

[View in the Asset Store](#) • [Publisher Website](#)

25 free decorative plants for your interior environments.
Some modern plants prefabs (+colliders) to decorate some of your rooms.

[More...](#)

Images & Videos



[View images & videos on Asset Store](#)

Package Size

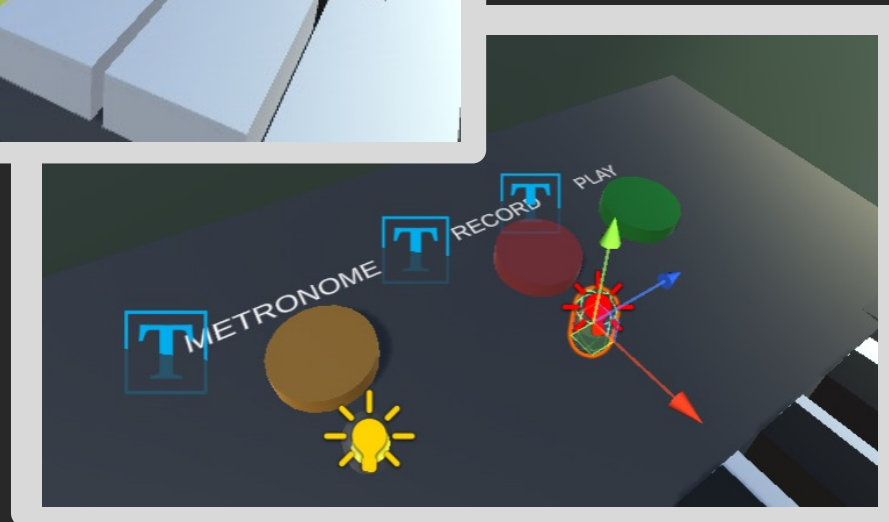
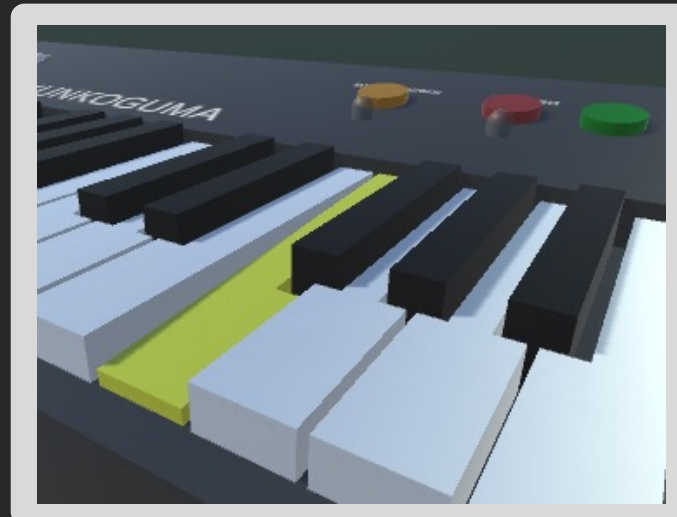
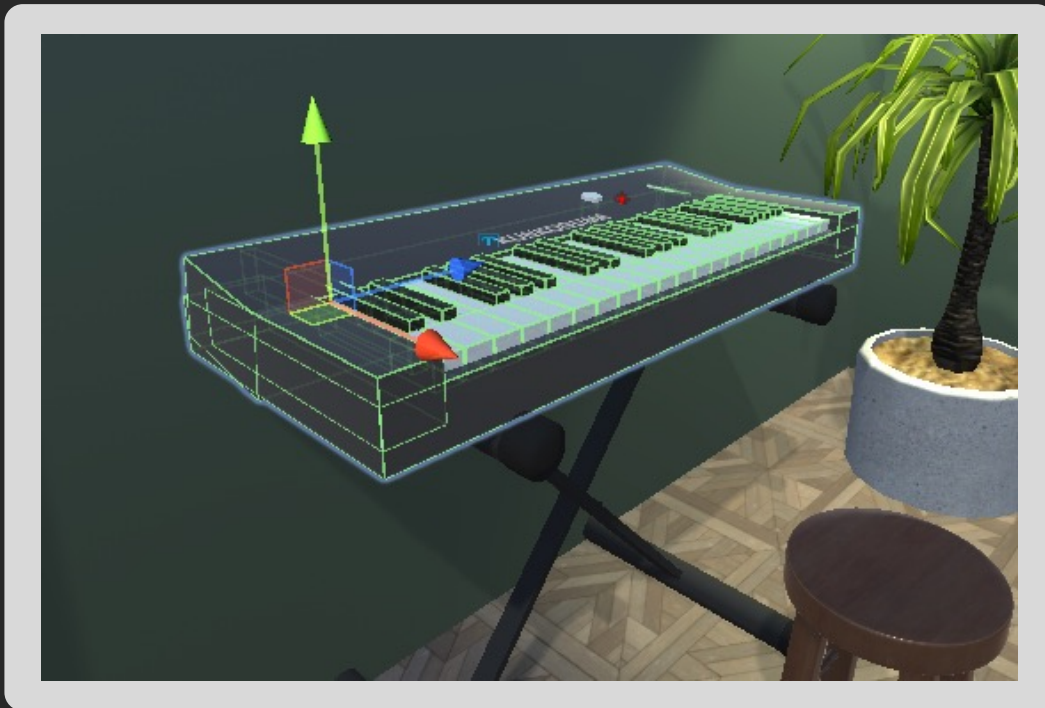
Size: 22.8 MB (Number of files: 48)

Supported Unity Versions

4.3.0 or higher

5. Process

1. Synthesizer Modeling



5. Process

2. 피아노의 모든 기능을 C# 스크립트 코딩으로 구현했습니다.



camera



keyboard



manual



metronome



mouseClick



mouseSou...



newrec

5. Contribution of each member



mouseSou...

금예인 : 마우스 입력 및 전반적 코드 빌드
박준서 : 옥타브, 로테이트 및
색팔 변경 구현, 볼륨 기능 구현
구지연 : 사운드 출력 구현
고윤정 : 볼륨 기능 구현



metronome

고윤정 : 코드 작성, 버튼 기능 구현
박준서 : 버그 수정



camera

금예인 : 코드 작성



keyboard

구지연 : 옥타브 및 입력 코드 작성,
사운드 출력, 볼륨 기능 구현
금예인 : 로테이트 및 색팔 변경 구현
고윤정 : 볼륨 기능 구현



manual

박준서 : 코드 작성



newrec

구지연 : 알고리즘 구현 및 기초 코드 작성
금예인 : 전반적 코드 빌드
박준서 : 버튼 기능 구현

5. Contribution of each member

김예인

소스 코드 설명본 작성
건반 모델링
본 유니티 씬 관리

구지연

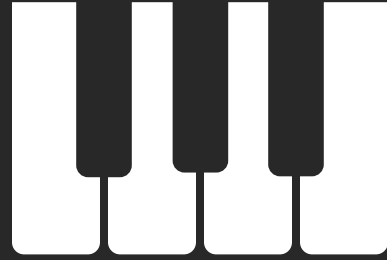
신디사이저 본체 및 주변공간 모델링
발표 PPT 준비
본 유니티 씬 관리

고윤정

소스 코드 설명본 작성
알고리즘 순서도 파일 제작
전반적 제출 파일 정리

박준서

오브젝트 버그 수정
발표 PPT 준비
알고리즘 순서도 작성



Synthesizer Simulation Time

Team Project by. **군고구마**

보고서 내용

1. 소스 코드 부가 설명

[mouseSound.cs]

: 마우스 클릭을 통해 피아노 건반을 누르고, 건반의 색상을 변경하기 위한 코드

: 흰 건반의 경우 box 두 개로 구성되어 있고, 하나의 부모 오브젝트를 가집니다. 건반의 부모 오브젝트는 Empty object로 각 건반의 끝에 위치합니다. 따라서 마우스 왼쪽 버튼이 DOWN일 때(좌클릭) 클릭된 오브젝트의 부모 오브젝트를 회전시켜 흰 건반을 누르듯 보이도록 하였고, 동시에 부모 오브젝트의 모든 자식 오브젝트 색상을 변경하였습니다. bool C1[]에 건반의 상태를 저장하고, 다시 마우스 버튼이 UP될 경우 DOWN되었던 건반을 반대로 회전시키고 흰색으로 변경하여 본 모습으로 돌아가도록 하였습니다. 검은 건반의 경우 하나의 box이므로, 클릭된 오브젝트를 곧바로 회전시키고 색상이 바뀌도록 하였습니다. 클릭된 검은 건반 오브젝트를 GameObject clicked에 담아 형태를 되돌릴 때 사용하였습니다. 또한 모든 피아노 음을 AudioClip[] note에 삽입하여 각 건반이 클릭될 때마다 해당 음을 연주하도록 하였습니다. keyboard.cs의 static 변수 vol을 사용해 신디사이저 상단 볼륨 조절 버튼을 클릭하면 음량 조절이 가능하도록 구현하였습니다.

보고서 내용

1. 소스 코드 부가 설명

[keyboard.cs]

: 키보드 입력을 통해 피아노 건반을 누르고, 건반의 색상을 변경하기 위한 코드

: 키보드 입력은 [A(C), W(C#), S(D), E(D#), D(E), F(F), T(F#), G(G), Y(G#), H(A), U(A#), J(B), K(C)]키를 통해 받아, 실제 신디사이저와 유사하게 연주할 수 있도록 하였습니다. 또한 왼쪽 방향키, 오른쪽 방향키로 옥타브 조절이 가능하도록 코드를 작성하고, 현재 옥타브를 나타낼 변수 `current_octave`를 사용해 모델링한 모든 건반을 연주할 수 있도록 만들었습니다. `AudioClip[] note`에 모든 음을 담아 상황에 맞추어 재생될 수 있도록 하였고, 각 게이름에 해당하는 키보드를 눌렀을 때 `current_octave`에 따라 다른 클립을 재생시켰습니다. 또한 동시에 두 가지 키를 눌렀을 때도 재생이 가능하도록 `PlayOneShot()` 함수를 사용하였습니다. 건반을 마우스로 클릭하였을 때와 동일하게 키보드를 누르면 오브젝트를 회전시키고 색상을 변경하였으며, 키보드가 다시 올라오면 반대 과정을 거치도록 했습니다. `static` 변수 `vol`을 통해, +키와 -키를 누르면 음량 조절이 가능하도록 하였습니다.

보고서 내용

1. 소스 코드 부가 설명

[newrec.cs]

: 직전에 연주한 곡을 바로 들어볼 수 있도록 신디사이저의 녹음 기능을 구현한 코드

: 키보드나 마우스로 연주한 곡을 녹음하고, 다시 재생시킬 수 있도록 코드를 작성하였습니다. 해당 기능은 스페이스바를 누르거나, 신디사이저 상단의 record 버튼을 클릭할 때 변수 Rec_onoff가 1이 되며 시작됩니다. 녹음이 시작되면 Rec_onoff이 1인 동안 uem[u]에 클릭한, 또는 키를 누른 음이 삽입되고 새로운 음이 삽입되기 전까지 recTime[u]가 증가합니다. 즉, 한 음의 지속 시간(음과 음 사이의 간격)이 recTime에 기록됩니다. 새로운 음이 들어올 때마다 u 또한 증가하며, 이에 따라 종료될 때까지 연주된 모든 음의 순서와 타이밍이 배열에 저장됩니다. 다시 스페이스바를 누르거나 record 버튼을 클릭하면 Rec_onoff가 0이 되며 녹음이 종료됩니다. 이렇게 저장된 녹음본이 있을 때 bool record의 값이 true가 되며, p키를 누르거나 record 재생 버튼을 클릭하면 녹음본이 재생됩니다. uem에 기록된 음에 해당하는 클립을 재생하고, recTime만큼 지난 후 다음 음을 재생하도록 하였습니다. 이때 recTime만큼의 프레임 변화를 측정할 변수로 frameTime을 사용하였습니다. 한 음이 재생되면 재생된 음의 수를 나타내는 변수인 playu가 증가하고, 이 playu가 기록된 음의 개수에 다르면 녹음본 재생은 종료됩니다. 새로운 녹음이 시작되기 전까지는 기존 녹음본이 존재하며 언제든지 다시 재생해볼 수 있습니다. 더불어 녹음 중에는 신디사이저 record 버튼 옆 조명이 켜지도록 하여, 사용자가 쉽게 녹음 여부를 파악할 수 있도록 하였습니다. 녹음 기능 역시 AudioClip[]에 모든 음을 담아 사용하였고, 키보드 입력을 위해 방향키를 통한 옥타브 조절이 가능하도록 구현하였습니다.

보고서 내용

1. 소스 코드 부가 설명

[metronome.cs]

: 메트로놈을 키보드 혹은 마우스로 조작하기 위한 코드

: 메트로놈 오브젝트를 만들어서 마우스로 해당 오브젝트를 클릭했을 때 메트로놈이 작동하고, 숫자버튼 1과 키패드 1을 눌렀을 때 메트로놈이 작동합니다. 메트로놈이 작동하기 시작할 때 메트로놈 오브젝트 아래에 불이 켜지게 하였고 작동이 멈추면 불도 꺼지게 하였습니다. 방향키의 위와 아래버튼을 누르면 메트로놈의 속도가 조정되게 하였습니다.

[manual.cs]

: 신디사이저 조작법을 확인할 수 있게 만든 코드

: 신디사이저에 구현한 다양한 기능들을 한 눈에 볼 수 있도록 매뉴얼로 만들었습니다. 이 매뉴얼은 shift버튼을 눌러 확인할 수 있습니다. 모델링한 방의 벽 뒤에 매뉴얼 오브젝트를 위치해놓고 shift버튼을 누르면 해당 오브젝트가 신디사이저 위에 위치할 수 있게 구현하였습니다.

보고서 내용

1. 소스 코드 부가 설명

[camera.cs]

: 카메라의 위치를 조정할 수 있는 코드

: 숫자버튼 0이나 키패드 0을 눌러서 카메라의 위치를 조정할 수 있습니다. 신디사이저의 건반을 볼 수 있는 시점 (mainCam) 과, 신디사이저의 전반적인 외관을 확인할 수 있는 시점 (subCam), 총 두 가지의 시점으로 바라볼 수 있습니다. subCam으로 설정했을 때는 마우스를 활용하여 모델링한 방을 살펴볼 수 있습니다. 마우스 오른쪽 버튼을 누른 채로 마우스를 움직여 방을 둘러볼 수 있고, 마우스의 휠을 움직여 가까이 가거나 멀리서 바라볼 수 있습니다.

2. 팀원별 작성 코드

 mouseSou...	김예인 : 마우스 입력 및 전반적 코드 빌드 박준서 : 옥타브, 로테이트 및 색깔 변경 구현, 볼륨 기능 구현 구지연 : 사운드 출력 구현 고윤정 : 볼륨 기능 구현	 keyboard	구지연 : 옥타브 및 입력 코드 작성, 사운드 출력, 볼륨 기능 구현 김예인 : 로테이트 및 색깔 변경 구현 고윤정 : 볼륨 기능 구현
 metronome	고윤정 : 코드 작성, 버튼 기능 구현 박준서 : 버그 수정	 manual	박준서 : 코드 작성
 camera	김예인 : 코드 작성	 newrec	구지연 : 알고리즘 구현 및 기초 코드 작성 김예인 : 전반적 코드 빌드 박준서 : 버튼 기능 구현

Vscode 라이브 웨어, unity collaborate 등을 사용하여 실시간으로 서로의 코드를 보완하였습니다. 오류 수정 및 알고리즘 수정을 꾸준한 대면 및 비대면 만남을 통해 조원 모두 참여하여 프로젝트 진행을 효과적으로 할 수 있었습니다.

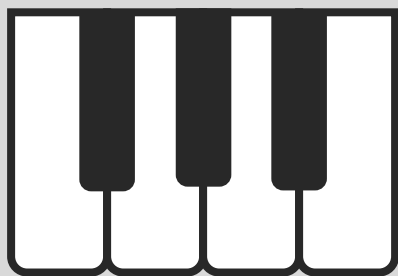
보고서 내용

3. 확인을 위한 컴파일 및 실행 방법

visual studio에서 C# 스크립트 파일을 작성하고, 저장하면 Unity에 자동 업로드가 됩니다.
unity에서 컴파일 후 플레이 버튼을 누르면 동작합니다.

4. 차용한 코드/SDK 등의 출처

Unity asset store에서 무료 asset 사용(발표자료 참조) 이외의 모든 모델링 및 코드는 팀원들이 직접 구현하였습니다.



감사합니다.

-군고구마-