



2017 프로그래밍 실습II 프로젝트 최종 보고서

융합 교과	음악
-------	----

작성자 : 2학년 5반 16번 이름: 정우열

요약

요즘 'dancing line'이란 게임이 유행이다. 이 게임은 정해진 길을 탭을 통해 방향을 전환하면서 박자를 맞추는 게임이다. 누구나 쉽게 할 수 있으나 혼자만 할 수 있는 단점이 있다. 그래서 우리는 서로 누가 더 박자를 잘 맞추는지 경쟁을 할 수 있는 리듬게임을 만들어 보았다. 이 프로그램은 크게 3개의 소스코드로 나눌 수 있다. 각각은 그림을 그리는 프로그램, 그림을 출력하는 프로그램, 그리고 박자 블록이 내려오는 프로그램이다. 핵심적인 아이디어는 나의 움직임을 연속촬영으로 찍어 이를 그림 그리는 프로그램을 통해 좌표를 모두 찍어 저장하는 것이다. 저장된 좌표를 다시 그림 출력하는 프로그램으로 출력한다. 그러면 아까 그림 그림이 똑같이 출력된다. 이를 0.2초 마다 한 장씩 출력하면 아주 짧은 애니메이션이 만들어진다. 이 프로그램을 통해 캐릭터가 싸움을 하는 것을 애니메이션처럼 보여주게 하였다. 가장 어려운 것은 블록이 선에 도달하는 시간과 사람이 누르는 시간의 차이를 출력하는 함수를 구현하는 것이었다. 한 블록 당 한 번씩만 출력되어야 해서 블록의 번호를 의미하는 변수를 통해 이 변수가 바뀌면 while문을 들어가게 하여 이 문제를 해결하였다.

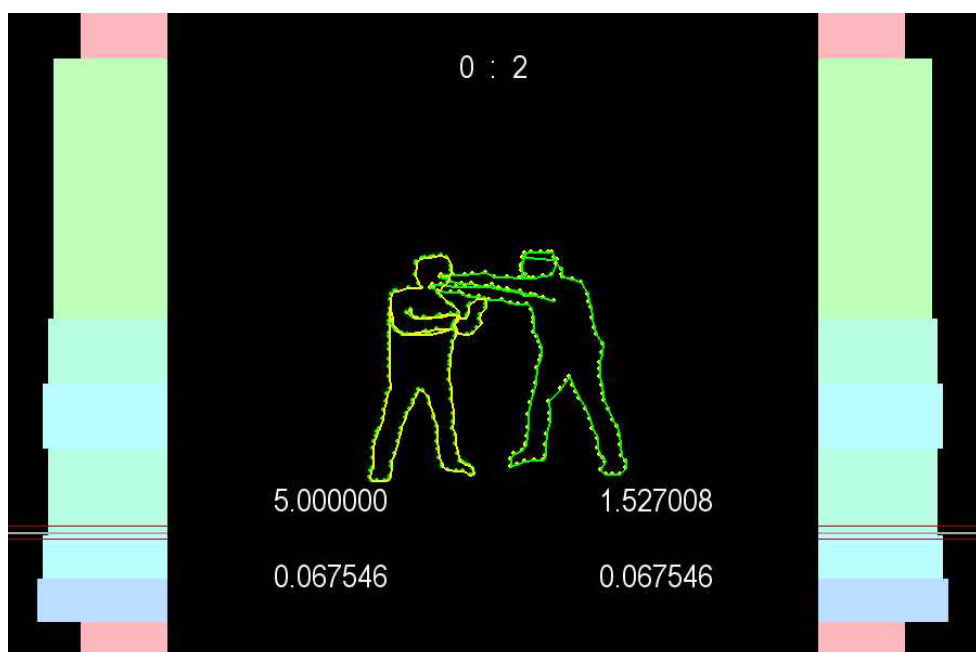


I. 프로그램 개발의 동기 및 목적

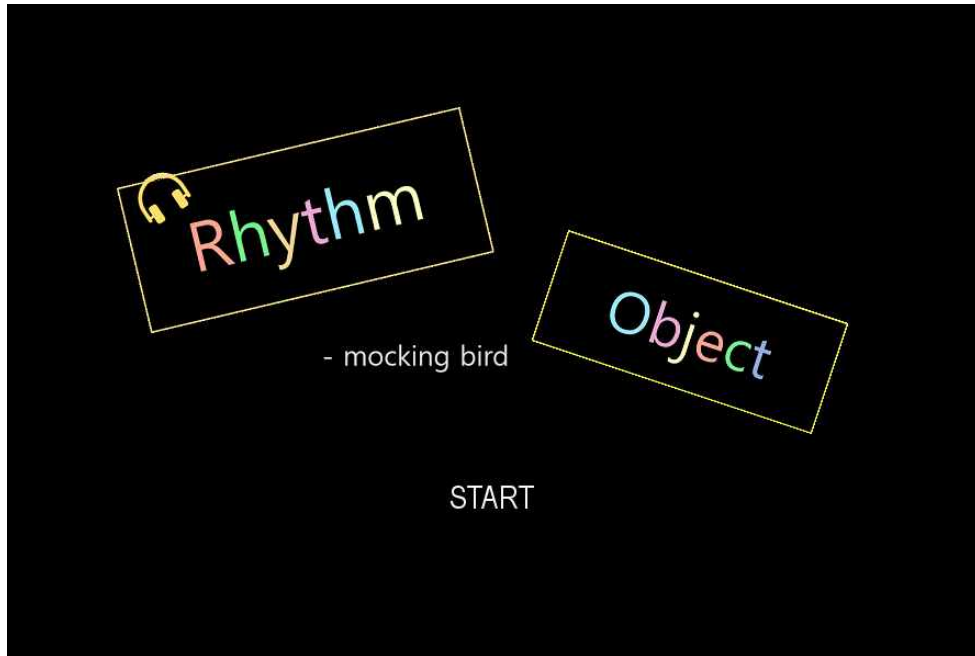
- 댄싱라인을 하던 도중 경준이의 실력이 너무 허접하여 조금 더 쉬운 버전을 만들어 주기로 하였다.
- 스스로 리듬게임을 잘한다고 착각하고 있는 경준이를 위해 실력을 비교할 수 있게 2인용 리듬게임을 제작해보기로 하였다.

II. 프로그램 소개

- 그림을 그리는 프로그램.
 - > 사진을 화면에 보여주고 마우스를 클릭하면 가장 최근에 누른 점과 클릭한 지점에 점이 생성, 연결된다.
 - > r을 누르면 가장 최근에 누른 점과 그 점과 연결된 선이 사라진다.
 - > n을 누르면 다음 사진으로 넘어가고 이전에 만든 사진의 좌표가 txt의 형태로 저장된다. 이전의 점은 모두 지워진다.
- 그림을 출력하는 프로그램.
 - > 0.2초마다 이전에 저장한 파일을 불러온다.
 - > 불러온 좌표를 통해 새로운 그림을 그린다.
- 리듬게임 프로그램.
 - > 박자와 음이 저장된 파일을 불러온다.
 - > 이 파일을 읽어 상자의 높이와 길이를 정하고 만들어낸다.
 - > 상자가 박자에 맞춰 내려오고, 각 플레이어의 구간에 맞춰 플레이어가 누르면 박자와 얼마나 차이가 나는지 시간차를 출력한다.
 - > 이전에 저장한 좌표를 두 번째 프로그램을 이용해 두 캐릭터가 마주보고 있는 이미지를 출력한다.
 - > 두 사람 중 더 정확하게 맞춘 사람의 캐릭터가 공격을 한다.
 - > 위에는 점수가 출력된다.



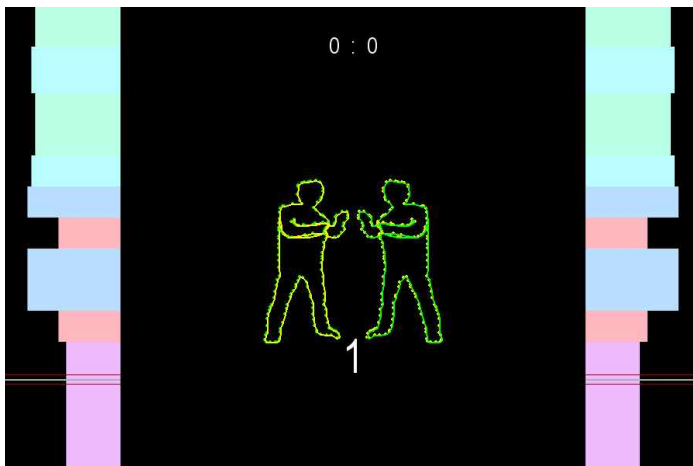
Ⅲ. 메뉴 구성도



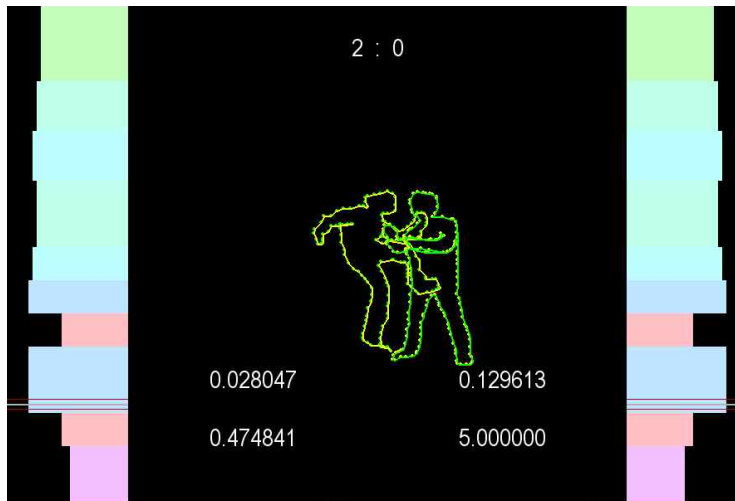
위에 보이는 바처럼 시작 버튼 하나 있다. 원래는 노래의 종류를 여러 개 하고 바자를 정하고 캐릭터 선택, 색깔 지정 등 많은 메뉴를 만들려 하였으나 시간의 부족으로 계획한 것으로 완성시키지 못하였다.

Ⅳ. 프로그램 사용법 및 주요 실행 결과

- 위에 보이는 메인 메뉴를 보면 이 게임은 'mocking bird' 라는 노래를 사용했음을 알려준다. 아래 보이는 START를 누르면 게임이 시작된다.



'START'를 누른 뒤 3초가 지나면 노래가 나오고 게임이 시작된다. 게임은 위처럼 오른쪽, 왼쪽 위에서 색깔별 블록이 내려오는데, 각 플레이어의 하단에 보이는 막대기에 도달하는 시간에 맞추어 'z', 'v'를 누르면 아래 그림처럼 숫자가 출력된다. 이 숫자는 실제 박자와 누른 시간 차이를 출력한 것이며 그 밑의 숫자는 이전에 출력되었던 숫자가 출력된다.



위에는 각 플레이어의 점수가 출력된다. 가끔 추가 점수를 준다. 캐릭터가 사용하는 기술에는 7가지 종류가 있다. 각 음마다 정해진 '발차기, 무릎 찌기', '눈 찌르기', '머리 박기', '어퍼컷', '넥 슬라이스', '펀치'가 존재한다. 상자의 길이는 박자와 비례하며 가로길이는 음에 비례한다.

V. 기술적 요소

1. 상자의 길이

상자의 길이는 악보를 다운받아 각각의 음, 박자의 길이를 txt파일에 적었다.



위는 한 예시이다. 이 메모장을 다시 불러와서 박자, 음을 받고 각 박자, 음에 해당하는 길이, 두께를 상자의 가로, 세로 길이로 지정한다.

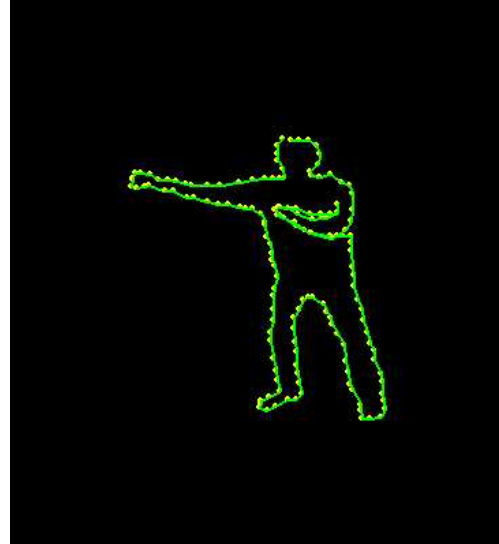
예를 들어 도를 10 이라고 하면 도#은 11, 레는 12이다. 이러면 12개의 음에 각각 숫자가 대입된다. 마찬가지로 8분 음표는 4, 16분 음표는 2와 같이 대입이 가능하다.

각각의 상자는 이 값들을 받고, 게임을 하면 고유의 음, 박자를 지니게 된다. 악보를 가져왔으니 박자도 당연히 노래와 일치할 것이다. 상자의 색도 같은 방식으로 지정하였다. 도부터 시까지 각각의 음은 고유의 색을 가진다. 아래는 실제로 사용한 색 값들이다.

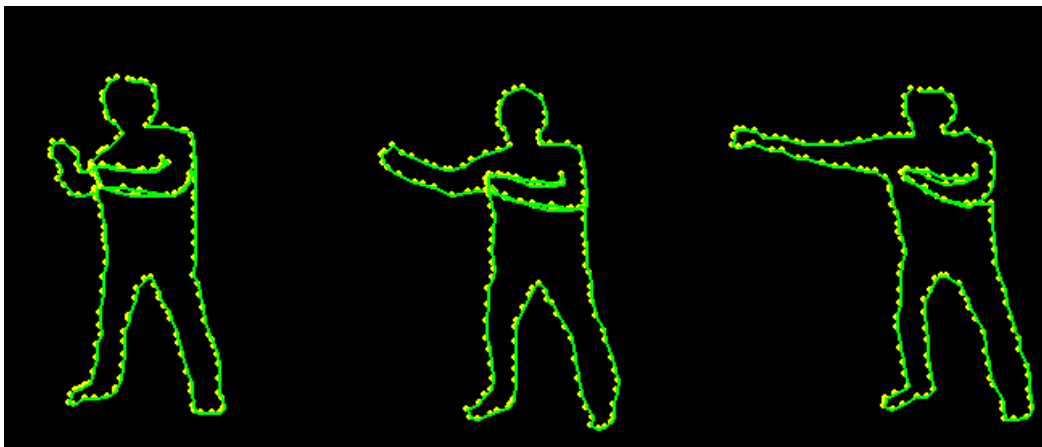
도 빨 (255, 185, 190) 미 주 (255, 223, 185) 파 노 (253, 255, 185) 솔 연초 (225, 255, 185)

2. 그림

> 나의 움직임을 아이폰의 연속촬영 기능을 이용해 모두 담았다. 그리고 각각의 사진을 첫 번째 프로그램을 이용해 오른쪽 이미지를 만들었다. 이 이미지를

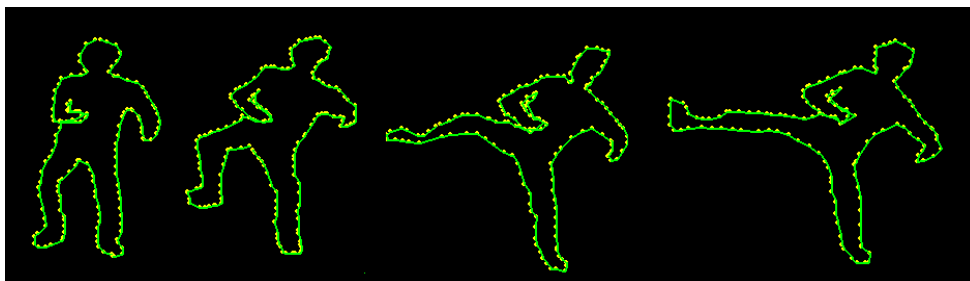


연속적으로 출력하면 모션이 애니메이션처럼 보인다.



사진을 첫 번째 프로그램으로 불러오면, 나는 프로그램으로 테두리에 적당히 많은 점을 찍는다. 그럼 자동으로 선이 이어진다. n을 누르면 각 좌표가 저장된다. 두 번째 프로그램은 다시 이를 위 이미지처럼 출력한다. 저장된 좌표 값을 받아 각 지점에 해당하는 좌표에 원을 그린 후 선으로 연결하는 작업을 내부에서 진행한다.

아래는 다른 동작의 모습이다.





3. 시간간격 출력

사실 가장 애는 많이 먹는 부분이다. 먼저 고려했던 부분들을 나열해 보자면

- 어떤 상자가 정해진 선에 충돌하는지를 알아야 한다.
- 만약 한 플레이어가 늦게 누른다면 숫자 출력은 언제 해야 하는가.

lineboxcollide() 라는 함수는 상자가 정해진 선과 충돌하면 시간이 배열에 추가된다. 그런데 이 상자는 박자를 맞추는 상자 간격 바로 위에 있어야 하는 상자여야 한다. 그래서 timeboxappend() 함수를 이용해 상자에 번호를 지정하고 상자가 지나가면 변수에 1을 더해 다음 상자의 아래 변으로 고려하도록 하였다.

그리고 선을 세 개 만들어서 가장 위의 선을 넘어가면 시간 차이를 출력하게 하였다. 만약 한 플레이어가 3개의 선 이내에 입력을 하지 않으면 5가 출력된다. 5는 가장 위와 가운데 선 사이 간격 시간보다 큰 수여서 항상 다른 수보다 크다. timer() 함수가 이를 조절하게 된다.

VI. 참고 문헌 및 웹 사이트

<https://www.jellynote.com/en/piano-sheet-music/eminem/mockingbird#tabs:A>

> 피아노 악보를 다운받은 곳

이외에는 우리가 저번에 만든 firstblood라는 게임의 소스코드를 참고하였다.

<https://wikidocs.net/12>

> 헛갈리는 개념들은 이 사이트에서 정리하였다.

VII. 개발 소감 (배운 점, 어려웠던 점과 해결했던 내용, 아쉬운 점, 느낀 점)

이 게임을 만들기 위해 생각, 시도했던 다른 것이 정말 많다. 가장 처음에 개획한 것은 첫 번째 계획서에 썼던 공굴리기 게임이다. 그러나 그건 저번 프로젝트와 상당히 비슷하였다. 그리고 만들기 매우 쉬워 보였다. 얼마 전 유행하는 리듬게임을 보고 리듬게임을 만들기로 결정하였다. 그리고 새롭고 특색 있는 리듬게임을 생각하기 위해 수 시간을 고민했다. 그리고는 만들기로 한 게 바운스 볼 스타일의 게임이다. 아래의 지면 위를 점프하며 박자를 맞추는 식이었다. 그러나 점프하는 시간 때문에 실패하고 이 프로젝트로 넘어갔다. 예술 리듬게임을 다음 목표로 잡고 게임을 구상하였다. 버튼을 누르면 지렁이나 나와서 잘 알려진 지렁이 게임을 하거나 아니면 버튼을 누르면 거품이 나와서 충돌하는 식의 게임을 계획했었다. 그러나 난이도와 시간의 문제로 리듬 전투 게임을 만들게 되었다. 이번 프로젝트로 창의적인 생각하는 법을 익혔던 거 같다. 오랜 시간 동안 가만히 앉아서 꼭 들어가야 하는 사항만을 앞에 적어두고 생각을 엄청 했다. 이런 방법의 생각으로 이렇게 재밌고 창의적인 프로젝트를 만들게 되었다. 혼자 머리 잡고 헤맸던 시간도 많다. 그러나 1학기에 수십 시간을 투자해 이런저런 오류들을 알았기 때문에 훨씬 적은 시간에 프로젝트를 만들 수 있었던 거 같다. 아직 완성하지 못한 개획했던 것들이 많아 너무 아쉽다.

VIII. 개발 일지

※ 일자별 개발 일지 모두 첨부

IX. 소스 코드



- 그림을 그리는 프로그램

```
import pygame, sys, random, time
from pygame.locals import *

# set up pygame
pygame.init()
mainClock = pygame.time.Clock()

WINDOWWIDTH = 900
WINDOWHEIGHT = 600
windowSurface = pygame.display.set_mode((WINDOWWIDTH, WINDOWHEIGHT), 0,
32)
pygame.display.set_caption('Rhy O')

BLACK = (0, 0, 0)
GREEN = (0, 255, 0)
WHITE = (255, 255, 255)
YELLOW = (255, 255, 0)

Image = []
num = 1
n = 0
mousexinput = []
mouseyinput = []
ok = False
mousex = 0
mousey = 0
mouseevent = False
what = 6
number = 0
ImageRect = []
ImageStretched = []
for i in range(4):
    Image.append(pygame.image.load('6.%d.png'%(i)))
#jumping=(pygame.image.load('jumping.jpg'))
    ImageStretched.append(pygame.transform.scale(Image[i], (300, 300)))
    ImageRect.append(ImageStretched[i].get_rect())
    ImageRect[i].centerx = 450
    ImageRect[i].centery = 300

SPEED = 500
while True:
```



```

windowSurface.fill(BLACK)
#windowSurface.blit(jumping,(0,0))
windowSurface.blit(ImageStretched[number], ImageRect[number])
for event in pygame.event.get():
    if event.type == QUIT:
        pygame.quit()
        sys.exit()
    if event.type == KEYUP:
        if event.key == ord('r'):
            if len(mousexinput) > 0:
                mouseXinput.remove(mouseXinput[len(mouseXinput)-1])
                mouseYinput.remove(mouseYinput[len(mouseYinput)-1])
                ok = False
            if event.key == ord('f'):
                ok = True
                print(mouseXinput)
                print(mouseYinput)
            if event.key == ord('n'):
                outfile=open("%d.%d.txt"%(what,number), "a")
                for num in mouseXinput:
                    outfile.write("%d\n"%(num))
                for num in mouseYinput:
                    outfile.write("%d\n"%(num))
                outfile.close()
                number += 1
                mouseXinput = []
                mouseYinput = []
                ok = False
        if event.type == MOUSEBUTTONDOWN:
            mouseX, mouseY = event.pos
            mouseXinput.append(mouseX)
            mouseYinput.append(mouseY)

    for i in range(len(mouseXinput)):
        pygame.draw.circle(windowSurface, YELLOW, (mouseXinput[i],
mouseYinput[i]), 2, 0)
        if i == len(mouseXinput)-1 and ok:
            pygame.draw.line(windowSurface, GREEN, (mouseXinput[i],
mouseYinput[i]), (mouseXinput[0], mouseYinput[0]), 2)
            if i > 0:
                pygame.draw.line(windowSurface, GREEN, (mouseXinput[i],
mouseYinput[i]), (mouseXinput[i-1], mouseYinput[i-1]), 2)

```




```
#windowSurface.blit(Image[num],(0,0))
pygame.display.update()
mainClock.tick(SPEED)
```

- 그림을 출력하는 프로그램

```
import pygame, sys, random, time
from pygame.locals import *

# set up pygame
pygame.init()
mainClock = pygame.time.Clock()

WINDOWWIDTH = 900
WINDOWHEIGHT = 600
windowSurface = pygame.display.set_mode((WINDOWWIDTH, WINDOWHEIGHT), 0,
32)
pygame.display.set_caption('Rhy O')

BLACK = (0, 0, 0)
GREEN = (0, 255, 0)
WHITE = (255, 255, 255)
YELLOW = (255, 255, 0)

Image = []
ok = False
picx = [[]for row in range(5)]
picy = [[]for row in range(5)]
pic = [[]for row in range(5)]
picture=[]

for num in range(4):
    picture.append(open('6.%d.txt'%(num),"r"))
    for line in picture[num]:
        pictur = line.split(',')
        for word in pictur:
            pic[num].append(int(word))
    longg = len(pic[num])
    picy[num] = pic[num][(longg)//2:]
    picx[num] = pic[num][: (longg)//2]

SPEED = 500
onc = 1
```

```

i=0
j=1

def printpicture(num):
    num=num//100
    for i in range(len(picx[num])):
        pygame.draw.circle(windowSurface, YELLOW, (picx[num][i], picy[num][i]), 2,
0)
        if i == len(picx[num])-1 and ok:
            pygame.draw.line(windowSurface, GREEN, (picx[num][i], picy[num][i]),
(picx[num][0], picy[num][0]), 2)
        if i > 0:
            pygame.draw.line(windowSurface, GREEN, (picx[num][i], picy[num][i]),
(picx[num][i-1], picy[num][i-1]), 2)

while True:
    windowSurface.fill(BLACK)
    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()
            sys.exit()
    while onc:
        printpicture(i)
        onc = 0
        i += 2*j
        if i == 398 or i==0:
            j=j*(-1)
    onc = 1
    pygame.display.update()
    mainClock.tick(SPEED)

```

```

import pygame, sys, random, time
from pygame.locals import *

```

- 리듬게임 프로그램

```

# set up pygame
pygame.init()
mainClock = pygame.time.Clock()

```

```

WINDOWWIDTH = 900
WINDOWHEIGHT = 600
windowSurface = pygame.display.set_mode((WINDOWWIDTH, WINDOWHEIGHT), 0,

```



32)

```
pygame.display.set_caption('Rhy O')
```

```
BLACK = (0, 0, 0)
```

```
GREEN = (0, 255, 0)
```

```
WHITE = (255, 255, 255)
```

```
YELLOW = (255, 255, 0)
```

```
basicFont2 = pygame.font.SysFont('Arial', 60)
```

```
basicFont1 = pygame.font.SysFont('Arial', 30)
```

```
SPEED = 80
```

```
Movespeed = SPEED/80
```

```
Once = True
```

```
Oncemusic = True
```

```
player1_click = False
```

```
player2_click = False
```

```
player1box = []
```

```
player2box = []
```

```
player1 = []
```

```
player2 = []
```

```
top1line=[]
```

```
mid1line=[]
```

```
bot1line=[]
```

```
top2line=[]
```

```
mid2line=[]
```

```
bot2line=[]
```

```
timebox1 = []
```

```
timebox2 = []
```

```
heightlist = []
```

```
sizelist = []
```

```
firsttime1 = 0
```

```
firsttime2 = 0
```

```
lasttime = 0
```

```
first_constant1 = 1
```

```
first_constant2 = 1
```

```
lastclickupdate1 = 0
```

```
lastclickupdate2 = 0
```

```
lasttimeupdate = 0
```

```
player1boxnumber = 1
```

```
player2boxnumber = 1
```

```
timeboxnumber = 1
```



```
oncemeet = 1
onceclick1 = 0
onceclick2 = 0
main = 1
mockingbird = 1
text1Rect = 0
timeing = 0
winner = 0
Image = []
onc0 = 1
onc1 = 1
onc2 = 1
onc3 = 1
onc4 = 1
onc5 = 1
onc6 = 1
ii=0
jj=1
ok = False
end = False
on = False
onepoint = 0
twopoint = 0

picx0 = [[]for row in range(5)]
picy0 = [[]for row in range(5)]
pic0 = [[]for row in range(5)]
picture0=[]
for num in range(4):
    picture0.append(open('0.%d.txt'%(num),"r"))
    for line in picture0[num]:
        pictur = line.split(',')
        for word in pictur:
            pic0[num].append(int(word))
    longg = len(pic0[num])
    picy0[num] = pic0[num][:(longg)//2:]
    picx0[num] = pic0[num][:(longg)//2]
picx1 = [[]for row in range(5)]
picy1 = [[]for row in range(5)]
pic1 = [[]for row in range(5)]
picture1=[]
for num in range(4):
```



```
picture1.append(open('1.%d.txt'%(num),"r"))
for line in picture1[num]:
    pictur = line.split(',')
    for word in pictur:
        pic1[num].append(int(word))
    longg = len(pic1[num])
    picy1[num] = pic1[num][(longg)//2:]
    picx1[num] = pic1[num][: (longg)//2]
picx2 = [[]for row in range(5)]
picy2 = [[]for row in range(5)]
pic2 = [[]for row in range(5)]
picture2=[]
for num in range(4):
    picture2.append(open('2.%d.txt'%(num),"r"))
    for line in picture2[num]:
        pictur = line.split(',')
        for word in pictur:
            pic2[num].append(int(word))
        longg = len(pic2[num])
        picy2[num] = pic2[num][(longg)//2:]
        picx2[num] = pic2[num][: (longg)//2]
picx3 = [[]for row in range(5)]
picy3 = [[]for row in range(5)]
pic3 = [[]for row in range(5)]
picture3=[]
for num in range(5):
    picture3.append(open('3.%d.txt'%(num),"r"))
    for line in picture3[num]:
        pictur = line.split(',')
        for word in pictur:
            pic3[num].append(int(word))
        longg = len(pic3[num])
        picy3[num] = pic3[num][(longg)//2:]
        picx3[num] = pic3[num][: (longg)//2]
picx4 = [[]for row in range(5)]
picy4 = [[]for row in range(5)]
pic4 = [[]for row in range(5)]
picture4=[]
for num in range(4):
    picture4.append(open('4.%d.txt'%(num),"r"))
    for line in picture4[num]:
        pictur = line.split(',')

```



```
        for word in pictur:
            pic4[num].append(int(word))
    longg = len(pic4[num])
    picy4[num] = pic4[num][(longg)//2:]
    picx4[num] = pic4[num][: (longg)//2]
    picx5 = [[]for row in range(5)]
    picy5 = [[]for row in range(5)]
    pic5 = [[]for row in range(5)]
    picture5=[]
    for num in range(3):
        picture5.append(open('5.%d.txt'%(num),"r"))
        for line in picture5[num]:
            pictur = line.split(',')
            for word in pictur:
                pic5[num].append(int(word))
        longg = len(pic5[num])
        picy5[num] = pic5[num][(longg)//2:]
        picx5[num] = pic5[num][: (longg)//2]
    picx6 = [[]for row in range(5)]
    picy6 = [[]for row in range(5)]
    pic6 = [[]for row in range(5)]
    picture6=[]
    for num in range(4):
        picture6.append(open('6.%d.txt'%(num),"r"))
        for line in picture6[num]:
            pictur = line.split(',')
            for word in pictur:
                pic6[num].append(int(word))
        longg = len(pic6[num])
        picy6[num] = pic6[num][(longg)//2:]
        picx6[num] = pic6[num][: (longg)//2]
    picxm = picx5[0]
    picym = picy5[0]

    size1=open('mockingbird size.txt',"r")
    for line in size1:
        sizelis = line.split()
        for word in sizelis:
            sizelist.append(word)
    print(len(sizelist))
    height1=open('mockingbird height.txt','r')
    for line in height1:
```



```
    heightlis = line.split()
    for word in heightlis:
        heightlist.append(word)
print(len(heightlist))
red1=open('red.txt','r')
for line in red1:
    redlist = line.split()
green1=open('green.txt','r')
for line in green1:
    greenlist = line.split()
blue1=open('blue.txt','r')
for line in blue1:
    bluelist = line.split()

class box1:
    def __init__(self):
        self.height = 0
        self.right = WINDOWWIDTH/6
        self.size = 0
        self.bottom = WINDOWHEIGHT
        self.red = 0
        self.blue = 0
        self.green = 0

class box2:
    def __init__(self):
        self.height = 0
        self.left = WINDOWWIDTH/6*5
        self.size = 0
        self.bottom = WINDOWHEIGHT
        self.red = 0
        self.blue = 0
        self.green = 0

def eventt():
    global player1_click
    global player2_click
    global first_constant1
    global first_constant2
    global firsttime1
    global firsttime2
    global onceclick1
```



```
global onceclick2
for event in pygame.event.get():
    if event.type == QUIT:
        pygame.quit()
        sys.exit()
    if event.type == KEYDOWN:
        if event.key == ord('z'):
            player1_click = True
            onceclick1 = 1
            while first_constant1:
                firsttime1 = time.time()
                first_constant1 = 0
        if event.key == ord('/'):
            player2_click = True
            onceclick2 = 1
            while first_constant2:
                firsttime2 = time.time()
                first_constant2 = 0
    if event.type == KEYUP:
        if event.key == K_ESCAPE:
            pygame.quit()
        if event.key == ord('z'):
            player1_click = False
            first_constant1 = 1
            onceclick1 = 1
        if event.key == ord('/'):
            player2_click = False
            first_constant2 = 1
            onceclick2 = 1

def makebox():
    global sizelist
    global player1box
    global player1
    global player2box
    global player2
    global sizelist
    global heightlist
    for i in range(len(sizelist)):
        BOX=box1()
        player1box.append(BOX)
        player1box[i].height = int(heightlist[i])*5+20
```




```
    player1box[i].size = int(sizelist[i])*10
    player1box[i].red = redlist[int(heightlist[i])%12]
    player1box[i].green = greenlist[int(heightlist[i])%12]
    player1box[i].blue = bluelist[int(heightlist[i])%12]
    if i > 0:
        player1box[i].bottom = player1box[i-1].bottom -
int(player1box[i-1].size)
    for i in range(len(sizelist)):
        BOX=box2()
        player2box.append(BOX)
        player2box[i].height = int(heightlist[i])*5+20
        player2box[i].size = int(sizelist[i])*10
        player2box[i].red = redlist[int(heightlist[i])%12]
        player2box[i].green = greenlist[int(heightlist[i])%12]
        player2box[i].blue = bluelist[int(heightlist[i])%12]
        if i > 0:
            player2box[i].bottom = player2box[i-1].bottom -
int(player2box[i-1].size)

def appendbox():
    global sizelist
    global player1box
    global player1
    global player2
    global player2box
    player1 = []
    player2 = []
    for i in range(len(sizelist)):
        player1.append(pygame.Rect(player1box[i].right - int(player1box[i].height),
player1box[i].bottom - int(player1box[i].size), int(player1box[i].height),
int(player1box[i].size)))
        player2.append(pygame.Rect(player2box[i].left, player2box[i].bottom -
int(player2box[i].size), int(player2box[i].height), int(player2box[i].size)))

def movebox():
    global Movespeed
    global sizelist
    global player1box
    global player2box
    for i in range(len(sizelist)):
        player1box[i].bottom += Movespeed
        player2box[i].bottom += Movespeed
```

```

def drawbox():
    global windowSurface
    global player1box
    global player2box
    global player1
    global player2
    for i in range(len(player1box)):
        pygame.draw.rect(windowSurface, (int(player1box[i].red),
int(player1box[i].green), int(player1box[i].blue)), player1[i])
        pygame.draw.rect(windowSurface, (int(player2box[i].red),
int(player2box[i].green), int(player2box[i].blue)), player2[i])

def drawline():
    global player1_click
    global player2_click
    global windowSurface
    global top1line
    global mid1line
    global bot1line
    global top2line
    global mid2line
    global bot2line
    top1line=[]
    mid1line=[]
    bot1line=[]
    top2line=[]
    mid2line=[]
    bot2line=[]
    top1line.append(pygame.Rect(0, WINDOWHEIGHT/100*79, WINDOWWIDTH/6,
1))
    mid1line.append(pygame.Rect(0, WINDOWHEIGHT/100*80, WINDOWWIDTH/6,
2))
    bot1line.append(pygame.Rect(0, WINDOWHEIGHT/100*81, WINDOWWIDTH/6,
1))
    top2line.append(pygame.Rect(WINDOWWIDTH/6*5, WINDOWHEIGHT/100*79,
WINDOWWIDTH/6, 1))
    mid2line.append(pygame.Rect(WINDOWWIDTH/6*5, WINDOWHEIGHT/100*80,
WINDOWWIDTH/6, 2))
    bot2line.append(pygame.Rect(WINDOWWIDTH/6*5, WINDOWHEIGHT/100*81,
WINDOWWIDTH/6, 1))
    if player2_click:

```



```
pygame.draw.rect(windowSurface,(0, 0, 0), mid2line[0])
pygame.draw.rect(windowSurface,(255, 0, 0), top2line[0])
pygame.draw.rect(windowSurface,(255, 0, 0), bot2line[0])
pygame.draw.rect(windowSurface,(174, 174, 174), mid1line[0])
pygame.draw.rect(windowSurface,(174, 0, 0), top1line[0])
pygame.draw.rect(windowSurface,(174, 0, 0), bot1line[0])
if player1_click:
    pygame.draw.rect(windowSurface,(0, 0, 0), mid1line[0])
    pygame.draw.rect(windowSurface,(255, 0, 0), top1line[0])
    pygame.draw.rect(windowSurface,(255, 0, 0), bot1line[0])
    pygame.draw.rect(windowSurface,(174, 174, 174), mid2line[0])
    pygame.draw.rect(windowSurface,(174, 0, 0), top2line[0])
    pygame.draw.rect(windowSurface,(174, 0, 0), bot2line[0])
if player1_click and player2_click:
    pygame.draw.rect(windowSurface,(0, 0, 0), mid1line[0])
    pygame.draw.rect(windowSurface,(255, 0, 0), top1line[0])
    pygame.draw.rect(windowSurface,(255, 0, 0), bot1line[0])
    pygame.draw.rect(windowSurface,(0, 0, 0), mid2line[0])
    pygame.draw.rect(windowSurface,(255, 0, 0), top2line[0])
    pygame.draw.rect(windowSurface,(255, 0, 0), bot2line[0])
if not player1_click and not player2_click:
    pygame.draw.rect(windowSurface,(174, 174, 174), mid1line[0])
    pygame.draw.rect(windowSurface,(174, 0, 0), top1line[0])
    pygame.draw.rect(windowSurface,(174, 0, 0), bot1line[0])
    pygame.draw.rect(windowSurface,(174, 174, 174), mid2line[0])
    pygame.draw.rect(windowSurface,(174, 0, 0), top2line[0])
    pygame.draw.rect(windowSurface,(174, 0, 0), bot2line[0])

def lineboxcollide():
    global lasttime
    global timeboxnumber
    global player1box
    global midline
    global lasttimeupdate
    global on
    if timeboxnumber < len(sizelist):
        if player1box[timeboxnumber].bottom > mid1line[0].top + 1:
            lasttime = time.time()
            timeboxnumber += 1
            lasttimeupdate = 1
            on = True
            #print(timeboxnumber)
```

```

def timer():
    global timebox1
    global timebox2
    global player1_click
    global player2_click
    global player1boxnumber
    global player2boxnumber
    global player1box
    global onceclick1
    global onceclick2
    global lastclickupdate1
    global lastclickupdate2
    #print(player1_click, onceclick1)
    if timeboxnumber < len(sizelist):
        if player1_click and player1box[player1boxnumber].bottom <
bot1line[0].bottom and player1box[player1boxnumber].bottom > top1line[0].top and
onceclick1:
            lastclickupdate1 = 1
            onceclick1 = 0
            #print(1)
        if timeboxnumber < len(sizelist):
            if player2_click and player1box[player2boxnumber].bottom <
bot1line[0].bottom and player1box[player2boxnumber].bottom > top1line[0].top and
onceclick2:
                lastclickupdate2 = 1
                onceclick2 = 0

def timeboxappend():
    global lastclickupdate1
    global lastclickupdate2
    global lasttimeupdate
    global player1boxnumber
    global player2boxnumber
    global lasttimeupdate
    global timeboxnumber
    global timebox1
    global timebox2
    global player1box
    global player2box
    global firsttime1
    global firsttime2

```



```
global lasttime
#print(lastclickupdate1, lasttimeupdate)
if lastclickupdate1 and lasttimeupdate:
    timebox1.append(abs(lasttime-firsttime1))
    lastclickupdate1 = 0
    player1boxnumber += 1
    #print(1, lasttime, firsttime1)
if lastclickupdate2 and lasttimeupdate:
    timebox2.append(abs(lasttime-firsttime2))
    lastclickupdate2 = 0
    player2boxnumber += 1
    #print(2, lasttime, firsttime2)
if player1box[timeboxnumber-1].bottom > bot1line[0].bottom and
player1boxnumber != player2boxnumber and lasttimeupdate:
    lasttimeupdate = 0
    if player1boxnumber > player2boxnumber:
        timebox2.append(5)
        player2boxnumber += 1
    if player1boxnumber < player2boxnumber:
        timebox1.append(5)
        player1boxnumber += 1
if player1box[timeboxnumber-1].bottom > bot1line[0].bottom and
player1boxnumber < timeboxnumber and player2boxnumber < timeboxnumber:
    player1boxnumber += 1
    player2boxnumber += 1
    lasttimeupdate = 0

def drawtimer():
    global timebox1
    global timebox2
    #print(timebox1)
    #print(timebox2)
    if len(timebox1) > 0:
        text1 = basicFont1.render('%f'%(timebox1[len(timebox1)-1]), True, WHITE)
        text1Rect = text1.get_rect()
        text1Rect.centerx = 300
        text1Rect.centery = 450
        windowSurface.blit(text1, text1Rect)
    if len(timebox2) > 0:
        text2 = basicFont1.render('%f'%(timebox2[len(timebox2)-1]), True, WHITE)
        text2Rect = text2.get_rect()
        text2Rect.centerx = 600
```

```

        text2Rect.centery = 450
        windowSurface.blit(text2, text2Rect)
    if len(timebox1) > 1:
        text1 = basicFont1.render('%f%(timebox1[len(timebox1)-2]), True, WHITE)
        text1Rect = text1.get_rect()
        text1Rect.centerx = 300
        text1Rect.centery = 520
        windowSurface.blit(text1, text1Rect)
    if len(timebox2) > 1:
        text2 = basicFont1.render('%f%(timebox2[len(timebox2)-2]), True, WHITE)
        text2Rect = text2.get_rect()
        text2Rect.centerx = 600
        text2Rect.centery = 520
        windowSurface.blit(text2, text2Rect)

def music():
    pygame.mixer.music.load('Mockingbird.mp3')
    pygame.mixer.music.play(-1,0.0)

def design():
    global text1Rect
    text1 = basicFont1.render('START', True, WHITE)
    text1Rect = text1.get_rect()
    text1Rect.centerx = 450
    text1Rect.centery = 450
    windowSurface.blit(text1, text1Rect)
    Rhythmimage = pygame.image.load('R.png')
    text2Rect = Rhythmimage.get_rect()
    text2Rect.centerx = 280
    text2Rect.centery = 200
    Objectimage = pygame.image.load('object.png')
    text3Rect = Objectimage.get_rect()
    text3Rect.centerx = 630
    text3Rect.centery = 300
    mockingimage = pygame.image.load('m.png')
    text4Rect = mockingimage.get_rect()
    text4Rect.centerx = 380
    text4Rect.centery = 320
    windowSurface.blit(Rhythmimage, text2Rect)
    windowSurface.blit(Objectimage, text3Rect)
    windowSurface.blit(mockingimage, text4Rect)

```



```
def design2():
    global onepoint
    global twopoint
    if onepoint > twopoint:
        text5 = basicFont1.render('2player win', True, WHITE)
        text5Rect = text5.get_rect()
        text5Rect.centerx = 450
        text5Rect.centery = 450
        windowSurface.blit(text5, text5Rect)
    if onepoint < twopoint:
        text5 = basicFont1.render('1player win', True, WHITE)
        text5Rect = text5.get_rect()
        text5Rect.centerx = 450
        text5Rect.centery = 450
        windowSurface.blit(text5, text5Rect)
    Rhythmimage = pygame.image.load('R.png')
    text2Rect = Rhythmimage.get_rect()
    text2Rect.centerx = 280
    text2Rect.centery = 200
    Objectimage = pygame.image.load('object.png')
    text3Rect = Objectimage.get_rect()
    text3Rect.centerx = 630
    text3Rect.centery = 300
    mockingimage = pygame.image.load('m.png')
    text4Rect = mockingimage.get_rect()
    text4Rect.centerx = 380
    text4Rect.centery = 320
    windowSurface.blit(Rhythmimage, text2Rect)
    windowSurface.blit(Objectimage, text3Rect)
    windowSurface.blit(mockingimage, text4Rect)
```

```
def win():
    global timebox1
    global timebox2
    global player1box
    global timeboxnumber
    global bot1line
    global winner
    global end
    global on
    global twopoint
    global onepoint
```

```

if timeboxnumber > 0 and len(timebox1) > 0 and len(timebox2) > 0 and on:
    if player1box[timeboxnumber-1].bottom > bot1line[0].bottom:
        if timebox1[len(timebox1)-1] > timebox2[len(timebox2)-1]:
            winner = 2
            twopoint += 1
            end = True
            on = False
        if timebox1[len(timebox1)-1] < timebox2[len(timebox2)-1]:
            winner = 1
            onepoint += 1
            end = True
            on = False

def printpicture0(num):
    num=num//10
    for i in range(len(picx0[num])):
        if winner == 2:
            pygame.draw.circle(windowSurface, YELLOW, (picx0[num][i],
picy0[num][i]), 2, 0)
        if winner == 1:
            pygame.draw.circle(windowSurface, GREEN, (900-picx0[num][i],
picy0[num][i]), 2, 0)
        if i == len(picx0[num])-1 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx0[num][i],
picy0[num][i]), (picx0[num][0], picy0[num][0]), 2)
        if i == len(picx0[num])-1 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx0[num][i],
picy0[num][i]), (900-picx0[num][0], picy0[num][0]), 2)
        if i > 0 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx0[num][i],
picy0[num][i]), (picx0[num][i-1], picy0[num][i-1]), 2)
        if i > 0 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx0[num][i],
picy0[num][i]), (900-picx0[num][i-1], picy0[num][i-1]), 2)
    for i in range(len(picxm)):
        if winner == 2:
            pygame.draw.circle(windowSurface, GREEN, (900-picxm[i], picym[i]), 2,
0)
        if winner == 1:
            pygame.draw.circle(windowSurface, YELLOW, (picxm[i], picym[i]), 2, 0)
        if i == len(picxm)-1 and winner == 2:
            pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),

```




```
(900-picxm[0], picym[0]), 2)
    if i == len(picxm)-1 and winner == 1:
        pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[0], picym[0]), 2)
    if i > 0 and winner == 2:
        pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[i-1], picym[i-1]), 2)
    if i > 0 and winner == 1:
        pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[i-1], picym[i-1]), 2)

def printpicture1(num):
    num=num//10
    print(num)
    for i in range(len(picx1[num])):
        if winner == 2:
            pygame.draw.circle(windowSurface, YELLOW, (picx1[num][i],
picy1[num][i]), 2, 0)
        if winner == 1:
            pygame.draw.circle(windowSurface, GREEN, (900-picx1[num][i],
picy1[num][i]), 2, 0)
        if i == len(picx1[num])-1 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx1[num][i],
picy1[num][i]), (picx1[num][0], picy1[num][0]), 2)
        if i == len(picx1[num])-1 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx1[num][i],
picy1[num][i]), (900-picx1[num][0], picy1[num][0]), 2)
        if i > 0 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx1[num][i],
picy1[num][i]), (picx1[num][i-1], picy1[num][i-1]), 2)
        if i > 0 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx1[num][i],
picy1[num][i]), (900-picx1[num][i-1], picy1[num][i-1]), 2)
    for i in range(len(picxm)):
        if winner == 2:
            pygame.draw.circle(windowSurface, GREEN, (900-picxm[i], picym[i]), 2,
0)
        if winner == 1:
            pygame.draw.circle(windowSurface, YELLOW, (picxm[i], picym[i]), 2, 0)
        if i == len(picxm)-1 and winner == 2:
            pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[0], picym[0]), 2)
```

```

        if i == len(picxm)-1 and winner == 1:
            pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[0], picym[0]), 2)
        if i > 0 and winner == 2:
            pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[i-1], picym[i-1]), 2)
        if i > 0 and winner == 1:
            pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[i-1], picym[i-1]), 2)

def printpicture2(num):
    num=num//10
    print(num)
    for i in range(len(picx2[num])):
        if winner == 2:
            pygame.draw.circle(windowSurface, YELLOW, (picx2[num][i],
picy2[num][i]), 2, 0)
        if winner == 1:
            pygame.draw.circle(windowSurface, GREEN, (900-picx2[num][i],
picy2[num][i]), 2, 0)
        if i == len(picx2[num])-1 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx2[num][i],
picy2[num][i]), (picx2[num][0], picy2[num][0]), 2)
        if i == len(picx2[num])-1 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx2[num][i],
picy2[num][i]), (900-picx2[num][0], picy2[num][0]), 2)
        if i > 0 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx2[num][i],
picy2[num][i]), (picx2[num][i-1], picy2[num][i-1]), 2)
        if i > 0 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx2[num][i],
picy2[num][i]), (900-picx2[num][i-1], picy2[num][i-1]), 2)
    for i in range(len(picxm)):
        if winner == 2:
            pygame.draw.circle(windowSurface, GREEN, (900-picxm[i], picym[i]), 2,
0)
        if winner == 1:
            pygame.draw.circle(windowSurface, YELLOW, (picxm[i], picym[i]), 2, 0)
        if i == len(picxm)-1 and winner == 2:
            pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[0], picym[0]), 2)
        if i == len(picxm)-1 and winner == 1:

```



```
        pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[0], picym[0]), 2)
        if i > 0 and winner == 2:
            pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[i-1], picym[i-1]), 2)
        if i > 0 and winner == 1:
            pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[i-1], picym[i-1]), 2)

def printpicture3(num):
    num=num//10
    for i in range(len(picx3[num])):
        if winner == 2:
            pygame.draw.circle(windowSurface, YELLOW, (picx3[num][i],
picy3[num][i]), 2, 0)
        if winner == 1:
            pygame.draw.circle(windowSurface, GREEN, (900-picx3[num][i],
picy3[num][i]), 2, 0)
        if i == len(picx3[num])-1 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx3[num][i],
picy3[num][i]), (picx3[num][0], picy3[num][0]), 2)
        if i == len(picx3[num])-1 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx3[num][i],
picy3[num][i]), (900-picx3[num][0], picy3[num][0]), 2)
        if i > 0 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx3[num][i],
picy3[num][i]), (picx3[num][i-1], picy3[num][i-1]), 2)
        if i > 0 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx3[num][i],
picy3[num][i]), (900-picx3[num][i-1], picy3[num][i-1]), 2)
    for i in range(len(picxm)):
        if winner == 2:
            pygame.draw.circle(windowSurface, GREEN, (900-picxm[i], picym[i]), 2,
0)
        if winner == 1:
            pygame.draw.circle(windowSurface, YELLOW, (picxm[i], picym[i]), 2, 0)
        if i == len(picxm)-1 and winner == 2:
            pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[0], picym[0]), 2)
        if i == len(picxm)-1 and winner == 1:
            pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[0], picym[0]), 2)
```



```
        if i > 0 and winner == 2:
            pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[i-1], picym[i-1]), 2)
        if i > 0 and winner == 1:
            pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[i-1], picym[i-1]), 2)

def printpicture4(num):
    num=num//10
    for i in range(len(picx4[num])):
        if winner == 2:
            pygame.draw.circle(windowSurface, YELLOW, (picx4[num][i],
picy4[num][i]), 2, 0)
        if winner == 1:
            pygame.draw.circle(windowSurface, GREEN, (900-picx4[num][i],
picy4[num][i]), 2, 0)
        if i == len(picx4[num])-1 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx4[num][i],
picy4[num][i]), (picx4[num][0], picy4[num][0]), 2)
        if i == len(picx4[num])-1 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx4[num][i],
picy4[num][i]), (900-picx4[num][0], picy4[num][0]), 2)
        if i > 0 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx4[num][i],
picy4[num][i]), (picx4[num][i-1], picy4[num][i-1]), 2)
        if i > 0 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx4[num][i],
picy4[num][i]), (900-picx4[num][i-1], picy4[num][i-1]), 2)
    for i in range(len(picxm)):
        if winner == 2:
            pygame.draw.circle(windowSurface, GREEN, (900-picxm[i], picym[i]), 2,
0)
        if winner == 1:
            pygame.draw.circle(windowSurface, YELLOW, (picxm[i], picym[i]), 2, 0)
        if i == len(picxm)-1 and winner == 2:
            pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[0], picym[0]), 2)
        if i == len(picxm)-1 and winner == 1:
            pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[0], picym[0]), 2)
        if i > 0 and winner == 2:
            pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
```



```
(900-picxm[i-1], picym[i-1]), 2)
    if i > 0 and winner == 1:
        pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[i-1], picym[i-1]), 2)

def printpicture5(num):
    num=num//10
    for i in range(len(picx5[num])):
        if winner == 2:
            pygame.draw.circle(windowSurface, YELLOW, (picx5[num][i],
picy5[num][i]), 2, 0)
        if winner == 1:
            pygame.draw.circle(windowSurface, GREEN, (900-picx5[num][i],
picy5[num][i]), 2, 0)
        if i == len(picx5[num])-1 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx5[num][i],
picy5[num][i]), (picx5[num][0], picy5[num][0]), 2)
        if i == len(picx5[num])-1 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx5[num][i],
picy5[num][i]), (900-picx5[num][0], picy5[num][0]), 2)
        if i > 0 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx5[num][i],
picy5[num][i]), (picx5[num][i-1], picy5[num][i-1]), 2)
        if i > 0 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx5[num][i],
picy5[num][i]), (900-picx5[num][i-1], picy5[num][i-1]), 2)
    for i in range(len(picxm)):
        if winner == 2:
            pygame.draw.circle(windowSurface, GREEN, (900-picxm[i], picym[i]), 2,
0)
        if winner == 1:
            pygame.draw.circle(windowSurface, YELLOW, (picxm[i], picym[i]), 2, 0)
        if i == len(picxm)-1 and winner == 2:
            pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[0], picym[0]), 2)
        if i == len(picxm)-1 and winner == 1:
            pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[0], picym[0]), 2)
        if i > 0 and winner == 2:
            pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[i-1], picym[i-1]), 2)
        if i > 0 and winner == 1:
```

```

        pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[i-1], picym[i-1]), 2)

def printpicture6(num):
    num=num//10
    for i in range(len(picx6[num])):
        if winner == 2:
            pygame.draw.circle(windowSurface, YELLOW, (picx6[num][i],
picy6[num][i]), 2, 0)
        if winner == 1:
            pygame.draw.circle(windowSurface, GREEN, (900-picx6[num][i],
picy6[num][i]), 2, 0)
        if i == len(picx6[num])-1 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx6[num][i],
picy6[num][i]), (picx6[num][0], picy6[num][0]), 2)
        if i == len(picx6[num])-1 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx6[num][i],
picy6[num][i]), (900-picx6[num][0], picy6[num][0]), 2)
        if i > 0 and winner == 2:
            pygame.draw.line(windowSurface, GREEN, (picx6[num][i],
picy6[num][i]), (picx6[num][i-1], picy6[num][i-1]), 2)
        if i > 0 and winner == 1:
            pygame.draw.line(windowSurface, YELLOW, (900-picx6[num][i],
picy6[num][i]), (900-picx6[num][i-1], picy6[num][i-1]), 2)
    for i in range(len(picxm)):
        if winner == 2:
            pygame.draw.circle(windowSurface, GREEN, (900-picxm[i], picym[i]), 2,
0)
        if winner == 1:
            pygame.draw.circle(windowSurface, YELLOW, (picxm[i], picym[i]), 2, 0)
        if i == len(picxm)-1 and winner == 2:
            pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[0], picym[0]), 2)
        if i == len(picxm)-1 and winner == 1:
            pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[0], picym[0]), 2)
        if i > 0 and winner == 2:
            pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[i-1], picym[i-1]), 2)
        if i > 0 and winner == 1:
            pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[i-1], picym[i-1]), 2)

```



```
def uploadpic0():
    global onc0
    global ii
    global jj
    global SPEED
    global end
    while onc0:
        printpicture0(ii)
        onc0 = 0
        ii += 5*jj
        if ii == 35:
            jj = -1
        if ii == 0:
            end = False
            jj = 1
    onc0 = 1
```

```
def uploadpic1():
    global onc1
    global ii
    global jj
    global SPEED
    global end
    while onc1:
        printpicture1(ii)
        onc1 = 0
        ii += 5*jj
        if ii == 35:
            jj = -1
        if ii == 0:
            end = False
            jj = 1
    onc1 = 1
```

```
def uploadpic2():
    global onc2
    global ii
    global jj
    global SPEED
    global end
    while onc2:
```



```
    printpicture2(ii)
    onc2 = 0
    ii += 5*jj
    if ii == 35:
        jj = -1
    if ii == 0:
        end = False
        jj = 1
    onc2 = 1
```

```
def uploadpic3():
    global onc3
    global ii
    global jj
    global SPEED
    global end
    while onc3:
        printpicture3(ii)
        onc3 = 0
        ii += 5*jj
        if ii == 45:
            jj = -1
        if ii == 0:
            end = False
            jj = 1
    onc3 = 1
```

```
def uploadpic4():
    global onc4
    global ii
    global jj
    global SPEED
    global end
    while onc4:
        printpicture4(ii)
        onc4 = 0
        ii += 5*jj
        if ii == 35:
            jj = -1
        if ii == 0:
            end = False
            jj = 1
```




```
onc4 = 1

def uploadpic5():
    global onc5
    global ii
    global jj
    global SPEED
    global end
    while onc5:
        printpicture5(ii)
        onc5 = 0
        ii += 5*jj
        if ii == 25:
            jj = -1
        if ii == 0:
            end = False
            jj = 1
    onc5 = 1

def uploadpic6():
    global onc6
    global ii
    global jj
    global SPEED
    global end
    while onc6:
        printpicture6(ii)
        onc6 = 0
        ii += 5*jj
        if ii == 35:
            jj = -1
        if ii == 0:
            end = False
            jj = 1
    onc6 = 1

def uploadpicm():
    for i in range (len(picxm)):
        pygame.draw.circle(windowSurface, YELLOW, (picxm[i], picym[i]), 2, 0)
        pygame.draw.circle(windowSurface, GREEN, (900-picxm[i], picym[i]), 2, 0)
        if i == len(picxm)-1:
            pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
```



```
(picxm[0], picym[0]), 2)
    pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[0], picym[0]), 2)
    if i > 0:
        pygame.draw.line(windowSurface, GREEN, (picxm[i], picym[i]),
(picxm[i-1], picym[i-1]), 2)
        pygame.draw.line(windowSurface, YELLOW, (900-picxm[i], picym[i]),
(900-picxm[i-1], picym[i-1]), 2)

while main:
    windowSurface.fill(BLACK)
    mousex = 0
    mousey = 0
    design()
    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()
            sys.exit()
        if event.type == MOUSEBUTTONDOWN:
            mousex, mousey = event.pos
            if mousex > text1Rect.left and mousex < text1Rect.right and mousey >
text1Rect.top and mousey < text1Rect.bottom:
                main = 0
    pygame.display.update()
    mainClock.tick(SPEED)

while mockingbird:
    windowSurface.fill(BLACK)
    eventtt()
    while timeing < SPEED*3:
        #print(timeing)
        timeing = timeing + 1
        text1 = basicFont2.render('%d'%(3-(timeing//SPEED)), True, WHITE)
        text1Rect = text1.get_rect()
        text1Rect.centerx = 450
        text1Rect.centery = 450
        windowSurface.blit(text1, text1Rect)
        break
    while Once:
        makebox()
        Once = False
    while Oncemusic and timeboxnumber == 2:
```



```
        music()
        Oncemusic = False
    appendbox()
    drawbox()
    movebox()
    drawline()
    lineboxcollide()
    timer()
    if timeboxnumber < len(sizelist):
        timeboxappend()
    drawtimer()
    win()
    if end:
        print(heightlist[timeboxnumber])
        if int(heightlist[timeboxnumber-1])%6 == 0:
            uploadpic0()
        if int(heightlist[timeboxnumber-1])%12 == 1:
            uploadpic1()
        if int(heightlist[timeboxnumber-1])%6 == 2:
            uploadpic2()
        if int(heightlist[timeboxnumber-1])%6 == 3:
            uploadpic3()
        if int(heightlist[timeboxnumber-1])%6 == 4:
            uploadpic4()
        if int(heightlist[timeboxnumber-1])%6 == 5:
            uploadpic5()
        if int(heightlist[timeboxnumber-1])%12 == 7:
            uploadpic6()
    if not end:
        uploadpicm()
    print(winner)
    text2 = basicFont1.render('%d : %d'%(onepoint, twopoint), True, WHITE)
    text2Rect = text2.get_rect()
    text2Rect.centerx = 450
    text2Rect.centery = 50
    windowSurface.blit(text2, text2Rect)
    pygame.display.update()
    if timeboxnumber == len(sizelist):
        mockingbird = 0

while main:
    windowSurface.fill(BLACK)
```



```
mousex = 0
mousey = 0
design2()
for event in pygame.event.get():
    if event.type == QUIT:
        pygame.quit()
        sys.exit()
    if event.type == MOUSEBUTTONDOWN:
        mousex, mousey = event.pos
        if mousex > text1Rect.left and mousex < text1Rect.right and mousey >
text1Rect.top and mousey < text1Rect.bottom:
            main = 0
pygame.display.update()
mainClock.tick(SPEED)
#print(firsttime1, firsttime2)
#print(timeboxnumber, player1boxnumber, player2boxnumber)
mainClock.tick(SPEED)
```