DEFEND YOUR LAND

2100093020 朴宣禹

# 一、创意介绍

从敌方的攻击守护你的领域的，快速简单的领地防御游戏。

# 二、设计方案和硬件连接

一个点（敌方单位）从上面往下移动，玩者需要**用 A B 按钮移动角色**，跟敌方单位冲突。

在敌方单位落地之前**冲突成功**，**得1分**；**冲突失败**，**失去一个生命**，显示‘爆炸’效果。

游戏开始角色**有三个生命**。

游戏在**水平9个空间中**进行，其中**显示玩者角色为中心的5个范围**。

如果敌方单位在**显示范围之外**，**箭头**出现，**指向存在敌方单位的方向**。

操作方法：

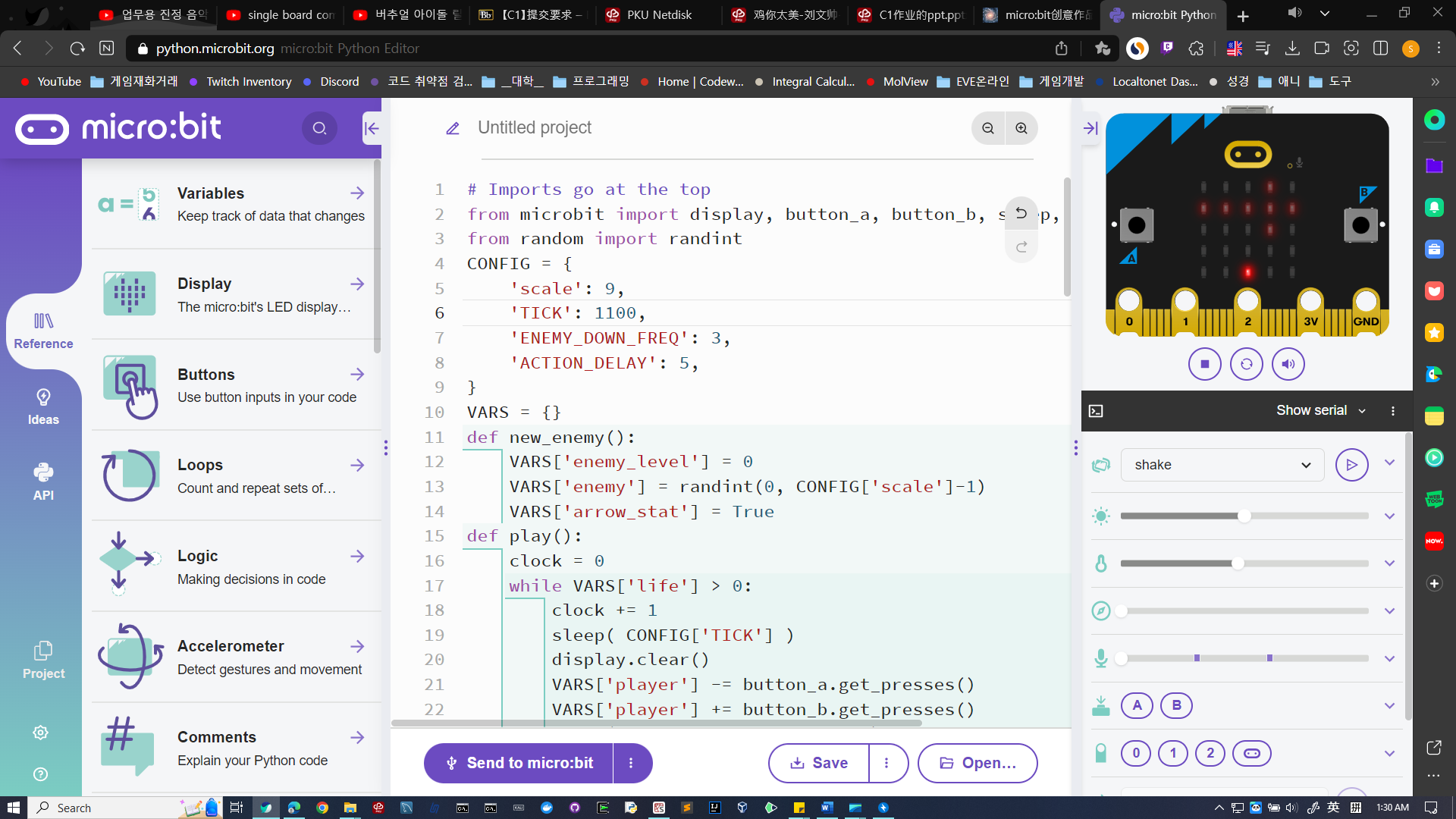
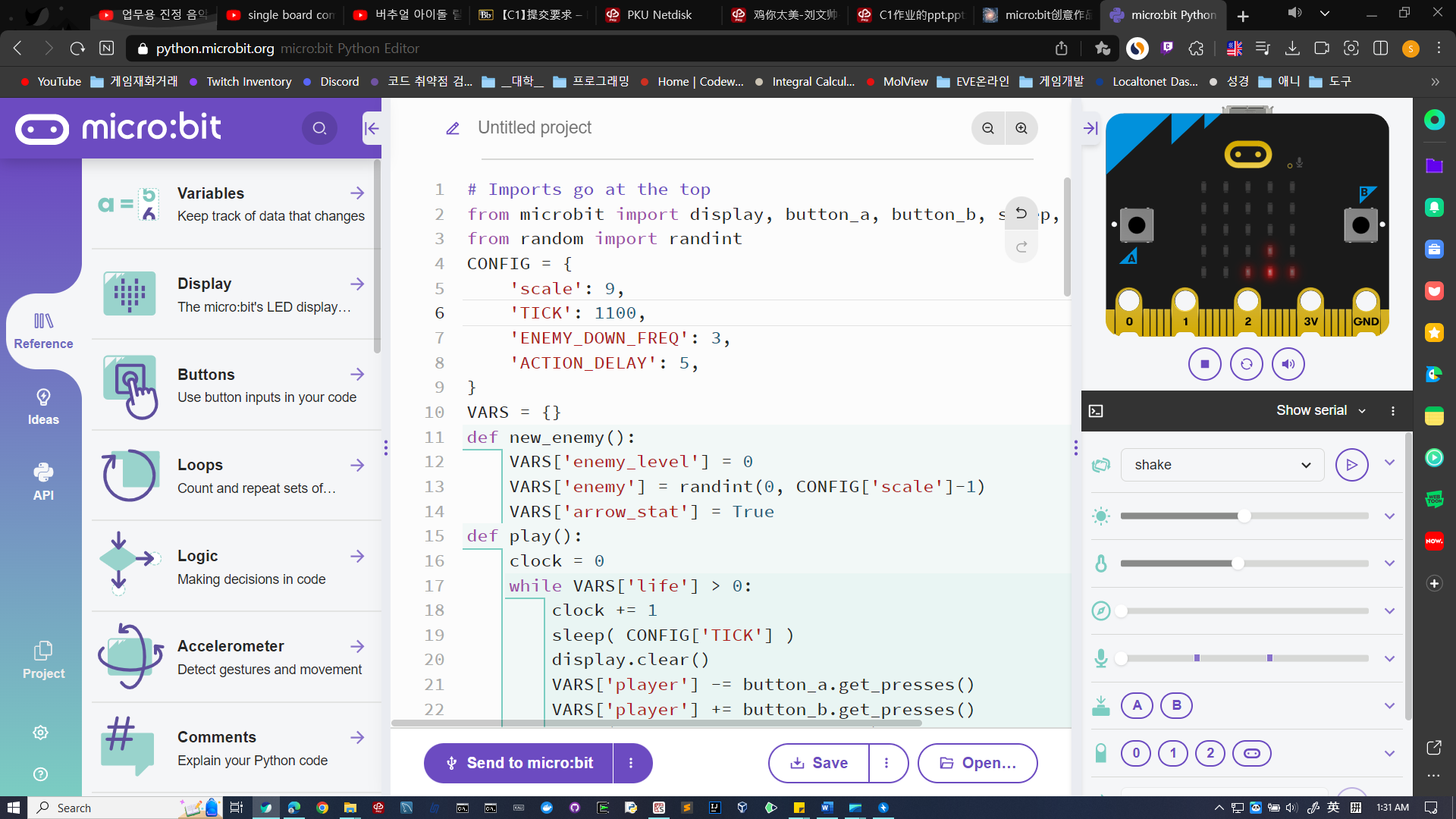
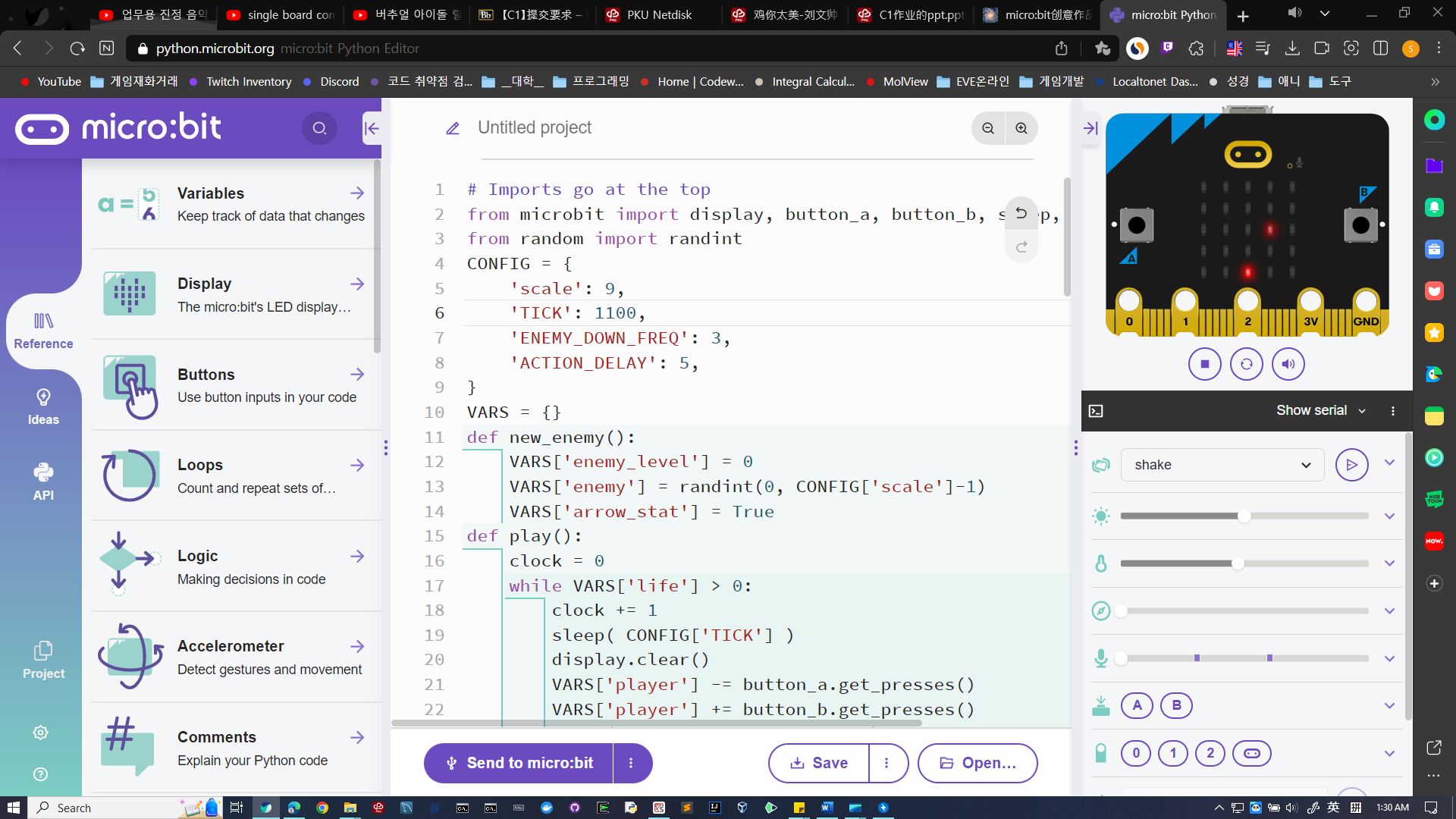
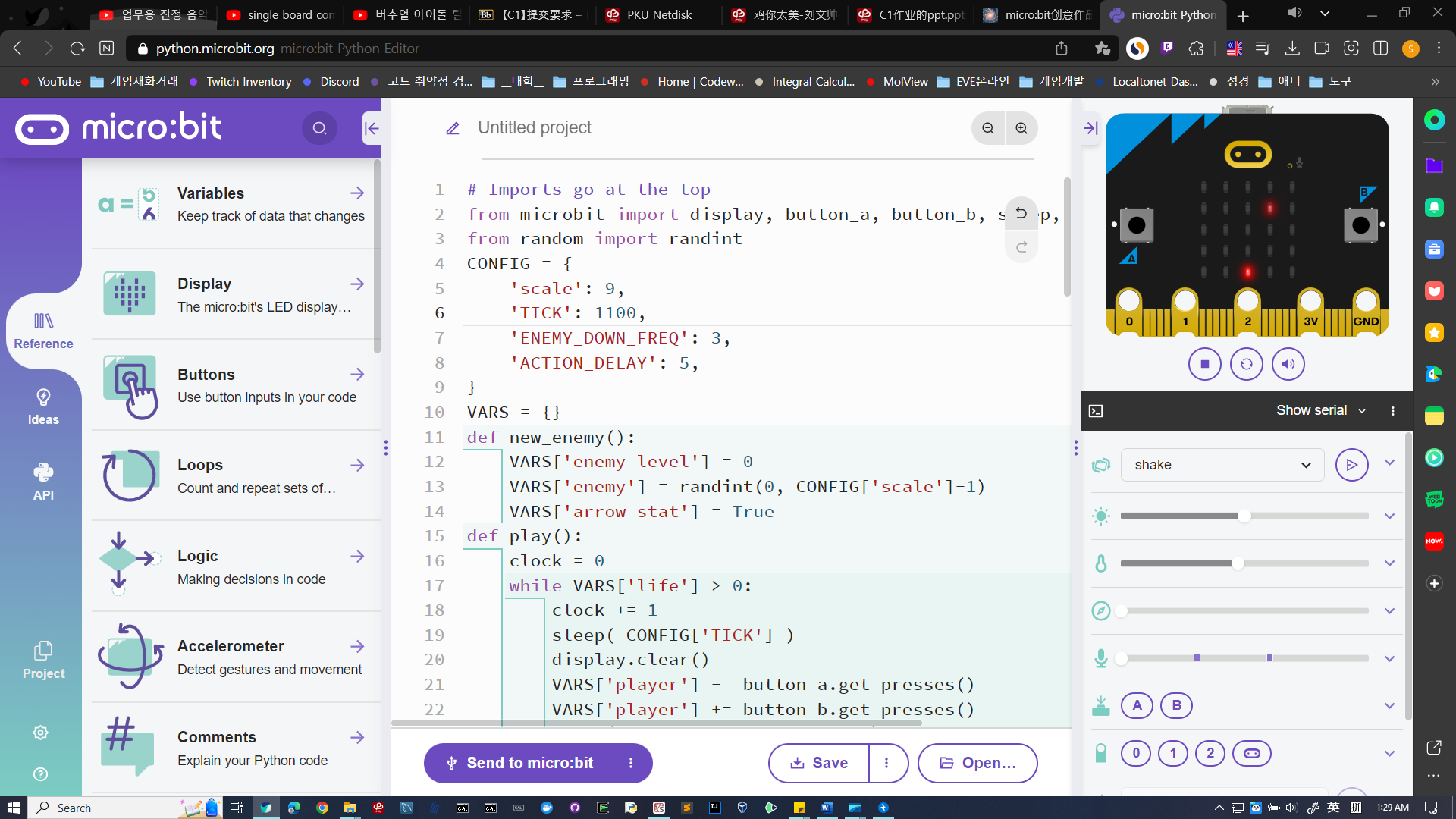
A：往左移动

B：往右移动

# 三、小组分工合作

全部工作由本人单独完成。

# 四、实现方案及代码分析



（游戏的简单示例图）

### 反复玩游戏以及结束以后的处理

**while** **True**:

VARS['score'] = 0

VARS['life'] = 3

VARS['player'] = CONFIG['scale']//2

new\_enemy()

play()

**if** **not** restart():

**break**

display.scroll('Good bye ~~', delay=75)

display.show(Image.HAPPY)

### 游戏一轮结束以后，问问角色是否再玩

**def** restart():

**while** **True**:

display.scroll("A:retry ", delay=50)

display.scroll("B:quit ", delay=50)

**for** i **in** range(12):

i = 2 - ( i % 3 )

l = ['0', '0']

l.insert(i, '9')

l = ''.join(l)

display.show(Image('00000:00000:00000:0**{}**0:00000'.format(l)))

sleep(200)

**if** button\_a.was\_pressed():

**return** **True**

**if** button\_b.was\_pressed():

**return** **False**

### microbit 包 import 和游戏初始化

**from** **microbit** **import** display, button\_a, button\_b, sleep, Image

**from** **random** **import** randint

CONFIG = {

'scale': 9,

'TICK': 110,

'ENEMY\_DOWN\_FREQ': 3,

'ACTION\_DELAY': 5,

}

VARS = {}

### 生成敌方单位

**def** new\_enemy():

VARS['enemy\_level'] = 0

VARS['enemy'] = randint(0, CONFIG['scale']-1)

VARS['arrow\_stat'] = **True**

### 包含主循环的 play 函数。为了节约存储空间，代码合并到一块了。

**def** play():

clock = 0

**while** VARS['life'] > 0:

clock += 1

sleep( CONFIG['TICK'] )

display.clear()

VARS['player'] -= button\_a.get\_presses()

VARS['player'] += button\_b.get\_presses()

sleep( CONFIG['ACTION\_DELAY'] )

relpos = VARS['enemy'] - VARS['player']

**if** **not** clock % CONFIG['ENEMY\_DOWN\_FREQ']:

VARS['enemy\_level'] += 1

**if** VARS['enemy\_level'] > 3:

**if** VARS['enemy'] == VARS['player']:

VARS['score'] += 1

clock = 0

new\_enemy()

**elif** VARS['enemy\_level'] > 4:

VARS['life'] -= 1

new\_enemy()

**if** abs(relpos) < 2:

display.set\_pixel(1+relpos,4,4)

display.set\_pixel(2+relpos,4,6)

display.set\_pixel(3+relpos,4,4)

display.set\_pixel(2+relpos,3,4)

sleep( CONFIG['TICK'] )

**elif** abs(relpos) < 3:

display.set\_pixel(2+relpos,3,4)

display.set\_pixel(2+relpos,4,6)

**if** relpos > 0:

display.set\_pixel(1+relpos,4,4)

**else**:

display.set\_pixel(3+relpos,4,4)

sleep( CONFIG['TICK'] )

sleep( CONFIG['ACTION\_DELAY'] )

**for** i **in** range(5):

display.set\_pixel(i,1,0)

display.set\_pixel(1,0,0)

display.set\_pixel(1,2,0)

display.set\_pixel(3,0,0)

display.set\_pixel(3,2,0)

relpos = VARS['enemy'] - VARS['player']

**if** abs(relpos) < 3 **and** VARS['enemy\_level'] < 5:

display.set\_pixel(2+relpos, VARS['enemy\_level'], 6)

**elif** relpos < -2:

**for** i **in** range(5):

display.set\_pixel(i,1,4)

display.set\_pixel(1,0,4)

display.set\_pixel(1,2,4)

**elif** relpos > 2:

**for** i **in** range(5):

display.set\_pixel(i,1,4)

display.set\_pixel(3,0,4)

display.set\_pixel(3,2,4)

display.set\_pixel(2, 4, 9)

display.scroll("GAME OVER", delay=50)

display.scroll("SCORE: ", delay=100)

display.scroll(VARS['score'], delay=250)

button\_a.was\_pressed()

button\_b.was\_pressed()

### 更易人类理解，但不是最新版本的 play 函数

**def** leftArrow():

**for** i **in** range(5):

display.set\_pixel(i,1,4)

display.set\_pixel(1,0,4)

display.set\_pixel(1,2,4)

**def** rightArrow():

**for** i **in** range(5):

display.set\_pixel(i,1,4)

display.set\_pixel(3,0,4)

display.set\_pixel(3,2,4)

**def** hideArrow():

**for** i **in** range(5):

display.set\_pixel(i,1,0)

display.set\_pixel(1,0,0)

display.set\_pixel(1,2,0)

display.set\_pixel(3,0,0)

display.set\_pixel(3,2,0)

**def** new\_enemy():

VARS['enemy\_level'] = 0

VARS['enemy'] = randint(0, CONFIG['scale']-1)

VARS['arrow\_stat'] = **True**

**def** display\_enemy():

hideArrow()

relpos = VARS['enemy'] - VARS['player']

**if** abs(relpos) < 3 **and** VARS['enemy\_level'] < 5:

display.set\_pixel(relpos+2, VARS['enemy\_level'], 6)

**elif** relpos < -2:

leftArrow()

**elif** relpos > 2:

rightArrow()

**def** display\_player():

display.set\_pixel(2, 4, 9)

**def** process\_movement():

VARS['player'] -= button\_a.get\_presses()

VARS['player'] += button\_b.get\_presses()

**def** update(clock):

process\_movement()

sleep( CONFIG['ACTION\_DELAY'] )

**if** **not** clock % CONFIG['ENEMY\_DOWN\_FREQ']:

VARS['enemy\_level'] += 1

**if** VARS['enemy\_level'] > 4:

new\_enemy()

**elif** VARS['enemy\_level'] > 3:

**if** VARS['enemy'] == VARS['player']:

VARS['score\_up'] = **True**

new\_enemy()

**elif** VARS['enemy\_level'] > 4:

VARS['life'] -= 1

new\_enemy()

**else**:

VARS['life'] -= 1

new\_enemy()

**def** play():

clock = 0

**while** VARS['life'] > 0:

clock += 1

print(clock)

print(VARS)

sleep( CONFIG['TICK'] )

display.clear()

update(clock)

**if** VARS['score\_up']:

VARS['score'] += 1

VARS['score\_up'] = **False**

clock = 0

sleep( CONFIG['ACTION\_DELAY'] )

display\_enemy()

display\_player()

display.scroll("GAME OVER", delay=50)

display.scroll("SCORE: ", delay=100)

display.scroll(VARS['score'], delay=250)