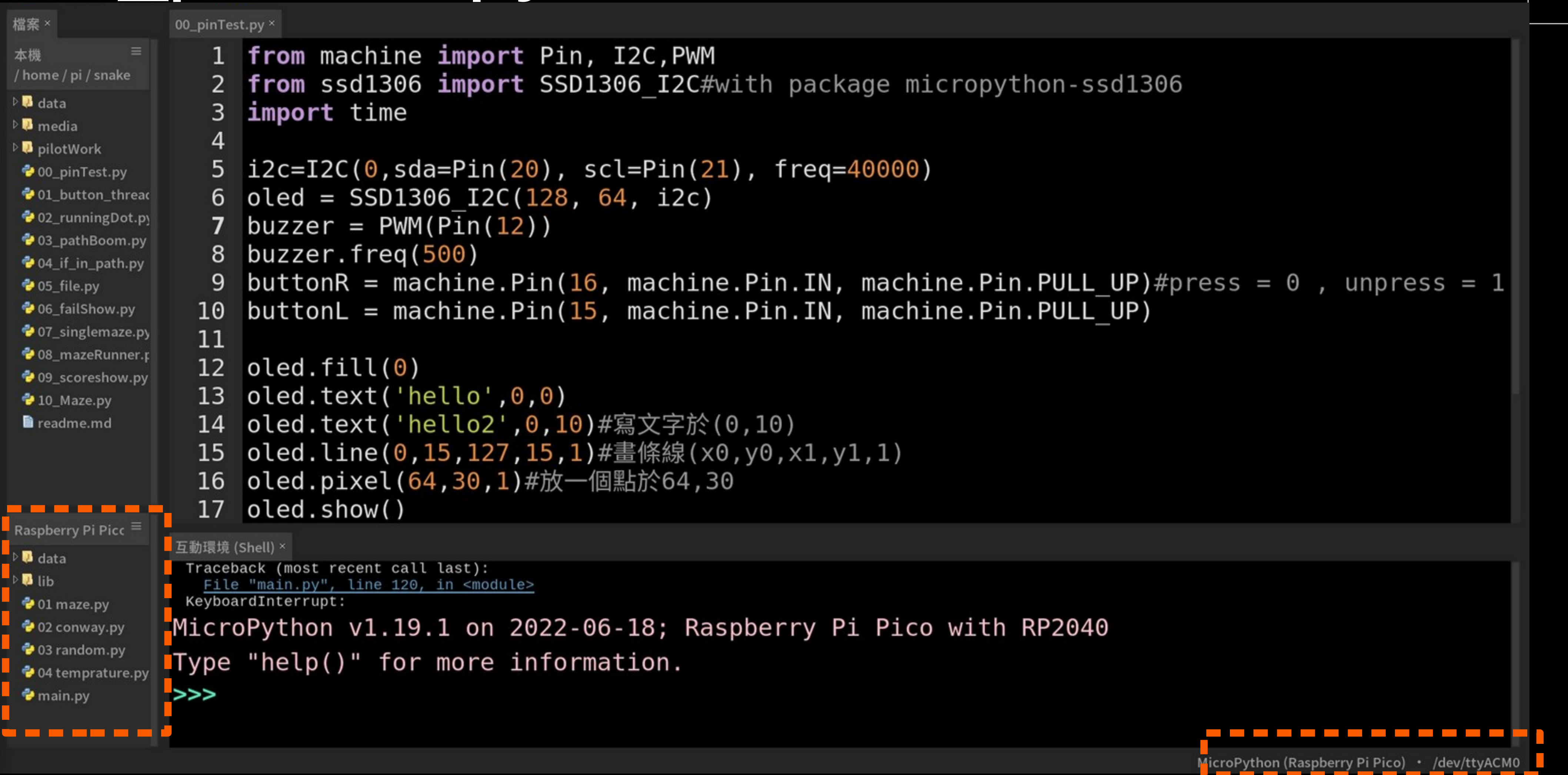
python 讀書會
資科專題迷你遊戲機
貪食蛇迷宮

chyijiunn

00_pinTest.py



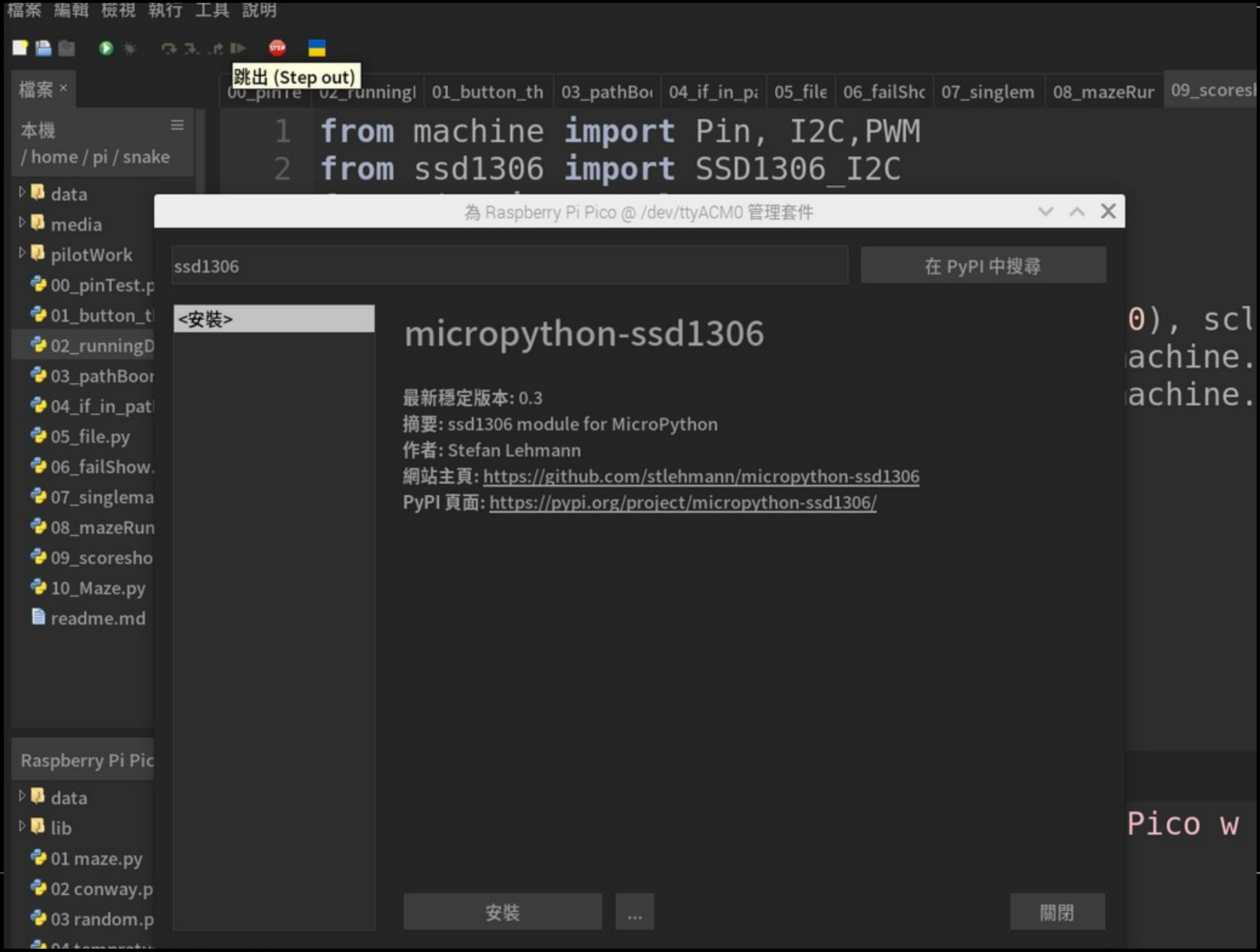
```python
from machine import Pin, I2C,PWM
from ssd1306 import SSD1306_I2C#with package micropython-ssd1306
import time

i2c=I2C(0,sda=Pin(20), scl=Pin(21), freq=40000)
oled = SSD1306_I2C(128, 64, i2c)
buzzer = PWM(Pin(12))
buzzer.freq(500)
buttonR = machine.Pin(16, machine.Pin.IN, machine.Pin.PULL_UP)#press = 0 , unpress = 1
buttonL = machine.Pin(15, machine.Pin.IN, machine.Pin.PULL_UP)

oled.fill(0)
oled.text('hello',0,0)
oled.text('hello2',0,10)#寫文字於(0,10)
oled.line(0,15,127,15,1)#畫條線(x0,y0,x1,y1,1)
oled.pixel(64,30,1)#放一個點於64,30
oled.show()
```

```
Traceback (most recent call last):
  File "main.py", line 120, in <module>
KeyboardInterrupt:
MicroPython v1.19.1 on 2022-06-18; Raspberry Pi Pico with RP2040
Type "help()" for more information.
>>>
```

Package - ssd1306



```
from machine import Pin, I2C,PWM
from ssd1306 import SSD1306_I2C
```

為 Raspberry Pi Pico @ /dev/ttyACM0 管理套件

ssd1306

micropython-ssd1306

最新穩定版本: 0.3
摘要: ssd1306 module for MicroPython
作者: Stefan Lehmann
網站主頁: https://github.com/stlehmann/micropython-ssd1306
PyPI 頁面: https://pypi.org/project/micropython-ssd1306/

貪食蛇需求

- 吃東西會變長
- 一直跑
- 可轉彎
- 撞到自己 **gameOver**

滿足~~貪食蛇~~課程需求

- ~~吃東西會變長~~
- 一直跑
- 可轉彎
- 撞到自己 **gameOver**

- ~~不只變長、還得縮短~~
- 新增像素點
- 搭配按鈕，至少兩顆
- 紀錄座標

滿足~~貪食蛇~~課程需求

- ~~吃東西會變長~~
- 跑：新增像素點
- 轉彎：兩顆按鈕
- 撞到：紀錄座標

- oled.pixel(x,y,1)
oled.show()

滿足跑需求

- 跑 : 新增像素點

- While True:

```
    oled.pixel(x,y,1)
    oled.show()
```

滿足跑需求

- 跑：新增像素點
- 不會跑

- **While True:**

```
    oled.pixel(x,y,1)
    oled.show()
```

滿足跑 - 方向需求

- 跑 : 新增像素點
- 會跑、跑去哪？

- While True:

```
    oled.pixel(x,y,1)
    oled.show()
    x+1
    y+1
```

設計的直覺 vs 使用的直覺

- **Direction ← button**
    - **While True:**

      ```
      oled.pixel(x,y,1)
      oled.show()
      if button1 == 0 AND  button2==0 : x = x+1
      ```

滿足跑 - 方向需求

- 跑：新增像素點
- 會跑、跑去哪？
  - Direction ← button

- While True:

```
oled.pixel(x,y,1)
oled.show()
if button1 == 0:x+1
if button2==0:y+1
```

滿足跑 - 方向需求

- 跑：新增像素點
- 不會跑、跑去哪？
- **Direction ← button**

- **While True:**

```
    button1 == 0 : direction +=1
    button2== 0 : direction += 1
```

- **While True:**

```
    oled.pixel(x,y,1)
    oled.show()
    if direction == ?:x+1
    if direction==?:y+1
```

滿足跑 - 方向需求

- 跑：新增像素點
- 不會跑、跑去哪？
- **Direction ← button**
  - **Derection == 0**
    1
    2
    3

- **While True:**
  button1 == 0 : direction +=1
  button2== 0 : direction -= 1
- **While True:**
  oled.pixel(x,y,1)
  oled.show()
  if direction == ?:x+1
  if direction==?:y+1

滿足跑 - 方向需求

- 跑：新增像素點
- 不會跑、跑去哪？
- **Direction ← button**
  - **Derection == 0**

    1

    2

    3

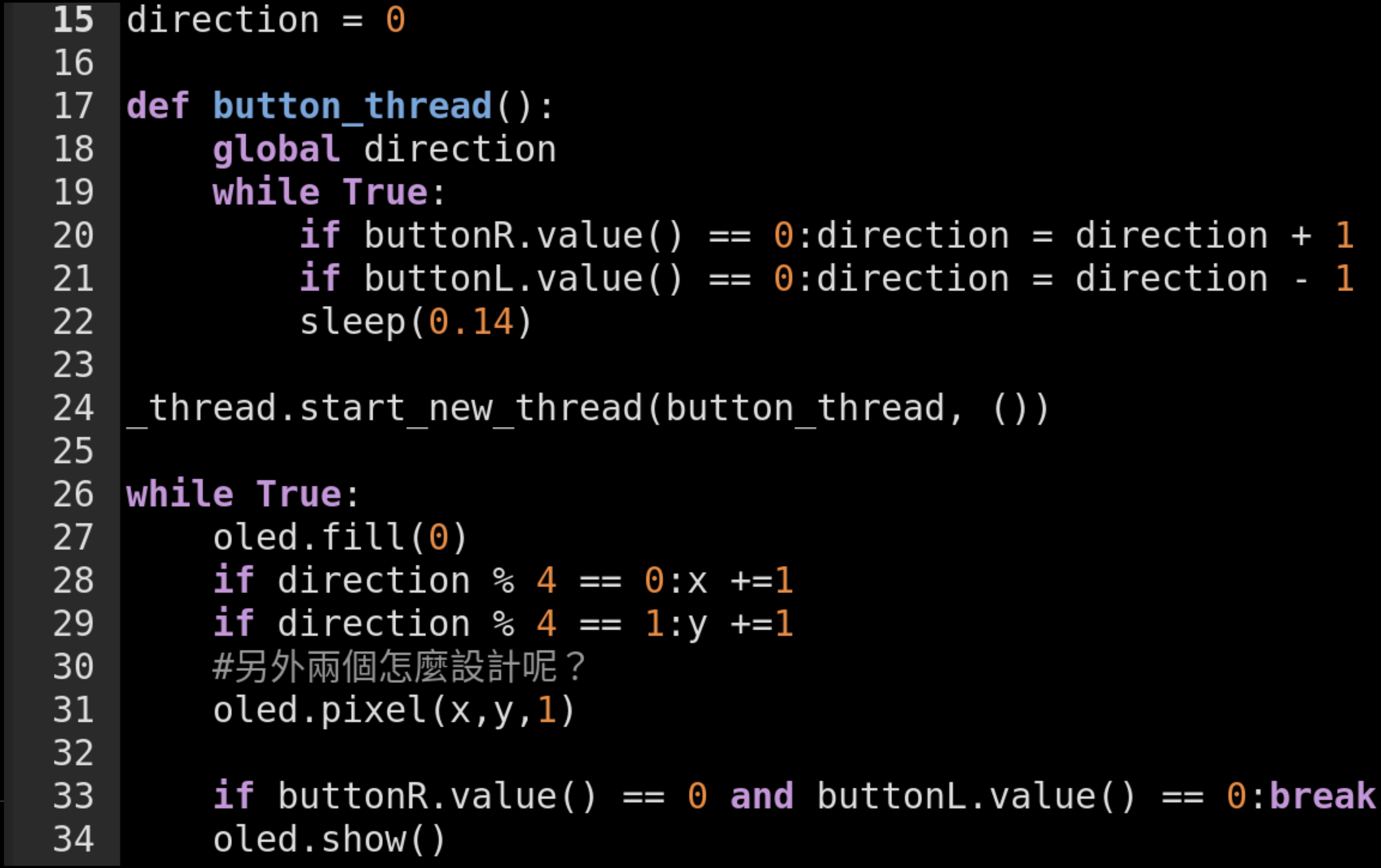
- **While True:**

```
    button1 == 0 : direction +=1
    button2== 0 : direction  -= 1
```

- **While True:**

```
    oled.pixel(x,y,1)
    oled.show()
    if direction%4 ==0 :x+1
    if direction%4 == 1:y+1
                then ?
```

02_runningDot.py



```python
direction = 0

def button_thread():
    global direction
    while True:
        if buttonR.value() == 0:direction = direction + 1
        if buttonL.value() == 0:direction = direction - 1
        sleep(0.14)

_thread.start_new_thread(button_thread, ())

while True:
    oled.fill(0)
    if direction % 4 == 0:x +=1
    if direction % 4 == 1:y +=1
    #另外兩個怎麼設計呢？
    oled.pixel(x,y,1)

    if buttonR.value() == 0 and buttonL.value() == 0:break
    oled.show()
```

滿足轉彎需求

- 跑 : 新增像素點
- 轉彎 : 兩顆按鈕
- 撞到 : 紀錄座標

- Oled.fill(x,y,1)
- If button ==0:turn()
- Path = []
  if in path:sys.exit

滿足紀錄需求

- 跑 : 新增像素點
- 轉彎 : 兩顆按鈕
- 撞到 : 紀錄座標

- Oled.fill(x,y,1)
- If button ==0:turn()
- Path = []
  if in path:sys.exit

滿足紀錄需求

- 撞到 : 紀錄座標

- Path = []
- Path.append()
- if in path:sys.exit