遊戲介紹及玩法

- 起源
  - 據說起源於中國，藉由被販賣到美洲的勞工外傳。
- 玩法
  - 常見的玩法為將硬幣分為為 三、四、五 3列。
  - 兩人輪流取硬幣，每次在同列取一個或以上的硬幣
  - 拿到最後一個硬幣的人就贏了

4

研究動機

- 高一上學期的數學研方老師所提到，當下就對它非常有興趣，也很想將它做成一個遊戲。
- 編寫出此程式的演算法
- 進一步美化此程式的使用者介面，設計不同版本及規則的遊戲。
- 選擇與 C++、Java相較之下比較好上手的 Python來完成此遊戲。

2

遊戲必勝原理

- 需使用二進位邏輯運算
- 若按照必勝原理，先手下必勝

5

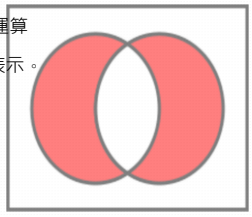
研究目標

- 學習Python的使用者介面程式架構
- 並用python程式語言設計出圖形介面的拈遊戲。
- 多媒體的功能

3

遊戲必勝原理

- 異或
  - 一種邏輯運算
  - 遊戲必勝原理的基本運算
  - 通常以XOR、⊕、^ 表示。



6

遊戲必勝原理

- 異或—真值表

XOR	True(1)	False(0)
True(1)	False(0)	True(1)
False(0)	True(1)	False(0)

遊戲必勝原理

- 若局面為安全殘局則下一次拿動必定為不安全殘局。
- 若玩家一直維持安全殘局就會勝利。

遊戲必勝原理

- 利用異或判斷進一步運算
- 舉例：3, 4, 5

棒子數	二進位
3	011
4	100
5	101
+ (xor)	010 不安全殘局

此時第一排拿走兩個

棒子數	二進位
1	001
4	100
5	101
+ (xor)	000 安全殘局

遊戲實作—演算法

- 利用迴圈進行每一排的異或判斷

```
xored= 0
for s in state:
    xored ^= s
```


遊戲必勝原理

- 安全與不安全殘局
  - 安全殘局：
    所有排數進行異或判斷的值為 0
  - 不安全殘局：
    所有排數進行異或判斷的值不為 0

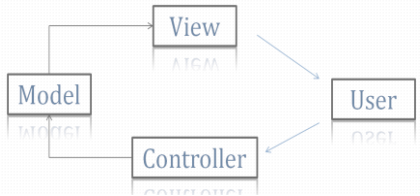
遊戲實作—演算法

- 利用異或判斷，決定該如何拿棒子。

```
if xored != 0:
for z in range(nRows):
    s = state[z] ^ xored
    if s <= state[z]:
        move= (z, s)
        break
```

遊戲實作—使用者介面

- MVC架構
  - 為使用者介面呈現方式
  - 分為模型、視圖、控制器三部分
  - 彼此互相連結，為一動態程式設計



13

遊戲實作—使用者介面

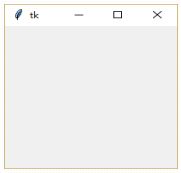
- 電腦移動棒子

```
def move(self, row, col):
        maxspalte = self.sticks[row]
        self.sticks[row] = col
        if self.game.mode == 0:
            if self.player == 0:
                self.player = 1
                self.move(row, col)
                self.player = 0
        elif self.game.mode == 1:
            if self.game_over():
                self.game.state =
NimGame.OVER
                self.winner = self.player
        if self.player == 0:
                self.player = 1
        else:
                self.player = 0
```

遊戲實作—使用者介面

- 建立視窗

```
self.screen =    turtle.Screen()
self.tk =        self.screen.getcanvas().master
```



14

遊戲實作—使用者介面

滑鼠點擊移動棒子

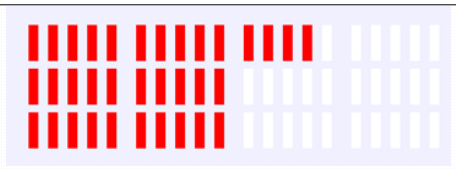
```
for stick in self.sticks.values():
    if self.game.mode == 0:
        stick.onclick(stick.makemove)
    elif self.game.mode == 1:
        stick.onclick(stick.playermove
```

17

遊戲實作—使用者介面

- 製作棒子

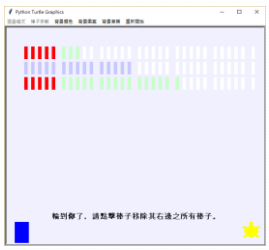
```
self.nRows= game.nRows
for row in range(self.nRows):
    for col in range(MAXSTICKS):
        self.sticks[(row, col)] = Stick(row, col, game
```



15

遊戲實作—使用者介面

- 電腦與滑鼠點擊移動棒子(藍色為電腦，綠色為滑鼠點擊)



18

遊戲實作—使用者介面

- 鍵盤控制

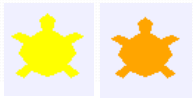
```
# 用 "space" 來 開始遊戲
self.game.screen.onkey(self.game.model.setup, "space")

# 用 "Esc" 來 清除 畫面
self.game.screen.onkey(self.game.view.clear, "Escape")
```

19

遊戲實作—使用者介面

- Level按鈕



```
class Level(Turtle):
    def __init__(self, game=None, x=0, y=0):
    def changevalue(self, x, y):
        if(self.value==True):
            self.value = False
            self.color('orange')
        else:
            self.value = True
            self.color('yellow')
```

22

遊戲實作—使用者介面

- 文字訊息

```
def set_msg(self):
        msg = '''
                    Python 拈遊戲 (NimGame)

以下是遊戲規則:
1. 玩家與電腦輪流撿棒子，撿到最後一個的就贏了。
2. 點擊任一棒子則玩家將會撿走該棒子右側的所有棒子。
3. 點擊左下角的方塊會給予提示。
4. 點擊右下角的烏龜可選擇難度(黃色代表難，橘色代表簡單)
5. 遊戲開始前請選擇排數
        '''
        self.t = Turtle()
        self.t.pu()
        self.t.ht()
        self.t.goto(0,-50)
        self.t.write(...)
```

20

遊戲實作—使用者介面

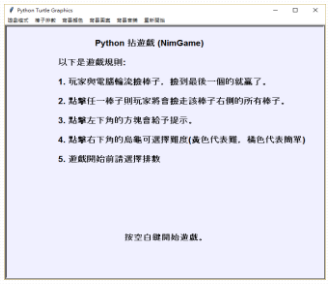
- Hint按鈕

```
class Hint(Turtle):
    def __init__(self, game, x=0, y=0):
        ...
        self.onclick(self.changevalue)
        self.onrelease(self.changevalue02)
    def changevalue(self, x, y):
        ...
    def changevalue02(self, x, y):
        ...
```

23

遊戲實作—使用者介面

- 文字訊息



21

遊戲實作—使用者介面

- Hint按鈕點擊前後

24

遊戲實作—使用者介面

- 選單列

25

遊戲實作—使用者介面

- 選單列—更改排數

```
cMenu = Menu(aMenu, tearoff=0)
    cLabel = ['3', '4', '5', '6',
              '7', '8', '9', '10']
        for i in cLabel:
            def cmd(x=i):
                nRows = int(x)
                mode = game.mode
                aNimGame= NimGame(nRows, mode)
            cMenu.add_command(label=i, command=cmd)
```

28

遊戲實作—使用者介面

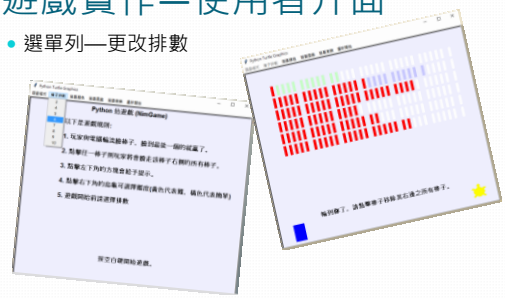
- 選單列—雙人模式

```
bMenu = Menu(aMenu, tearoff=0)
        #bLabel = ['單人模式', '雙人模式']
        bLabel = {'單人模式': 0,
                  '雙人模式': 1}
        keyL = sorted(bLabel)
        for k in keyL:
            def cmd(x=k):
                nRows = game.nRows
                mode = bLabel[x]
                aNimGame= NimGame(nRows, mode)
            bMenu.add_command(label=k, command=cmd)
```

26

遊戲實作—使用者介面

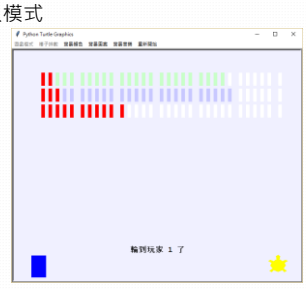
- 選單列—更改排數



29

遊戲實作—使用者介面

- 選單列—雙人模式



27

遊戲實作—使用者介面

- 選單列—背景顏色及圖案

```
dMenu = Menu(aMenu, tearoff=0)
        dLabel= {'顏色 1','顏色 2','顏色 3','顏色 4'}
        keyL = sorted(dLabel)
        for k in keyL:
            def cmd(x=k):
                game.screen.bgcolor(dLabel[x])
            dMenu.add_command(label= k, command= cmd)
        #-----------------------------------------
        eMenu = Menu(aMenu, tearoff=0)
        eLabel = {'無圖案','圖案 1','圖案 2','圖案 3'}
        keyL = sorted(eLabel)
        for k in keyL:
            def cmd(x=k):
                game.screen.bgpic(eLabel[x])
            eMenu.add_command(label = k, command=cmd)
```

30

遊戲實作—使用者介面

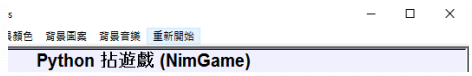
- 選單列—背景顏色及圖案

遊戲實作—使用者介面

- 選單列—重新開始

```
def cmd():
        aNimGame= NimGame(game.nRows, game.mode)
    aMenu.add_command(label= '重新開始', command= cmd)

    game.tk.config(menu= aMenu)
```

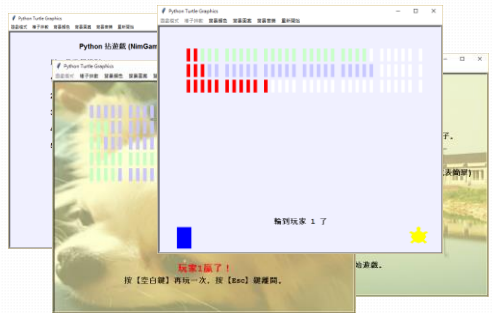


遊戲實作—使用者介面

- 選單列—背景音樂

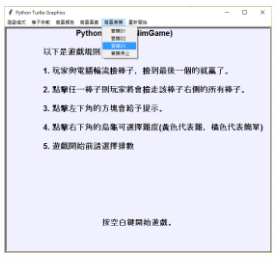
```
fMenu = Menu(aMenu, tearoff=0)
        fLabel= {'音樂停止': '__0__.mid',
                 '音樂01': 'musics/background.mid',
                 '音樂02': 'musics/canon.mid',
                 '音樂03': 'musics/John_Cena.mp3'}
        keyL = sorted(fLabel)
        for k in keyL:
            def cmd(x=k):
                musicFile= fLabel[x]
                if musicFile != '__0__.mid':
                    pygame.mixer.music.load(musicFile)
                    pygame.mixer.music.play(-1, 0.0)
                else:
                    pygame.mixer.music.stop()
            fMenu.add_command(label= k, command= cmd)
```


遊戲實作—成果展現



遊戲實作—使用者介面

- 選單列—背景音樂



總結

- 遊戲必勝法則
- 演算法
- MVC程式架構
- 三大 Python 模組
  - Turtle, Tk, Pygame
- Debug 能力

心得

- 經過了這一年，相信自己成長了許多，不管是寫程式、做簡報、寫小論文、以及面對一些問題的處理能力，都有所提升。也很感謝老師耐心的指導，給了我這樣的一個機會製作這個遊戲與大家分享。希望在未來，能夠將這一年所學到的，在相關的科系、領域中繼續發展下去，讓自己為社會盡一份力。

37

參考資料

- C:\Python35-32\Lib\turtledemo\nim.py
- https://zh.wikipedia.org/wiki/MVC
- http://rykids.blogspot.tw/
- http://episte.math.ntu.edu.tw/articles/mm/mm_03_2_02/
- file:///C:/Users/Calvin/Dropbox/cvry/ryNim/tc_%20nim.pdf

38