科技與機器人

Python製作桌遊《貓貓食堂》

作者：洪佑宣 劉家宇 邱俊凱 陳宥升 陳延禎 楊勛齊
服務單位:丹鳳高中
youxuan388@apps.ntpc.edu.tw
jerry062954@gmail.com

完整程式碼：https://github.com/youxuan388/kittys/blob/main/kittys.py

Abstract:

Our team has learned Data Structure.Therefore,we want to through a famous game called Kittys in order to deeply apply our knowledge.

1.前言：

我們這組有學習過資料結構!!!，因此我們想透過《貓貓食堂》這款遊戲，讓我們能深入應用我們所學的。我們想使用python來撰寫出可以自己一個人和三個電腦玩這個遊戲的程式。

Production Method:

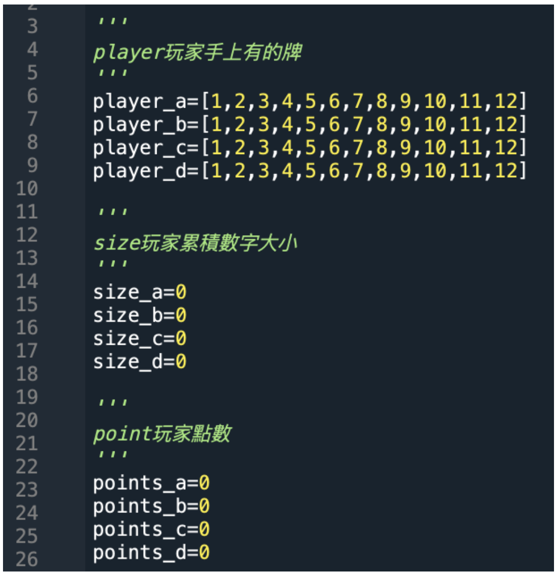
After playing the Kittys with classmates merely hundred times,we will understand more how the background of the game work,which helps us write programs smoothly.We will refer to many online tutorials and use the tools provided on the Internet to reduce the time we spend writing programs.When we encounter difficulties, we will seek help, and if possible, we will find ways to solve our confusion.

We decided to use a list data to store my and the computer's cards, and use variables to record the scores and points each cat got (Picture 1). Ask the player to enter the card to be played on the console and delete that card from the card library. The computer uses the random module to randomly draw cards from the card library(Picture 2)

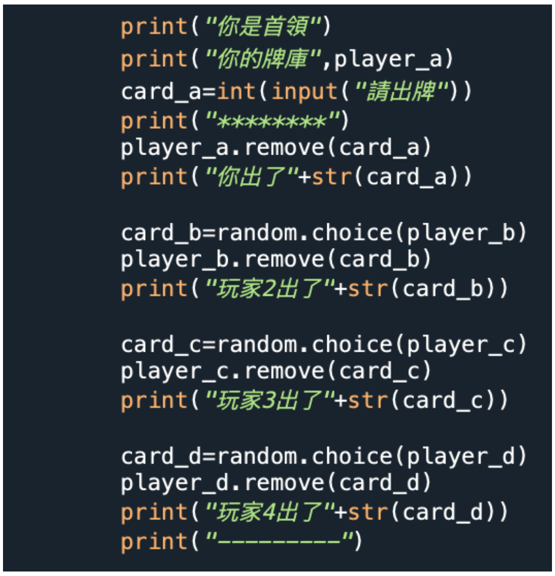
2.製作方法與過程：

認真遊玩過此遊戲後，與同學玩大約100次，更加了解遊戲的運作後，幫助我們撰寫程式順利。我們會許多網路上的教學和使用網路上所提供的工具，以減少我們撰寫程式的時程。遇到困難時，我們會去尋求幫助，可以的話，我們會想方設法將解決我們所困惑的。

我們決定使用串列來儲存我和電腦的卡牌，使用變數來紀錄每隻貓得到的分數和點數（如圖1）。請玩家在console上輸入要出的卡牌，並從牌庫裡把那張卡刪掉。電腦則是使用random模組從牌庫中隨意抽取卡牌（如圖2）。



```
'''
player玩家手上有的牌
'''
player_a=[1,2,3,4,5,6,7,8,9,10,11,12]
player_b=[1,2,3,4,5,6,7,8,9,10,11,12]
player_c=[1,2,3,4,5,6,7,8,9,10,11,12]
player_d=[1,2,3,4,5,6,7,8,9,10,11,12]

'''
size玩家累積數字大小
'''
size_a=0
size_b=0
size_c=0
size_d=0

'''
point玩家點數
'''
points_a=0
points_b=0
points_c=0
points_d=0
```

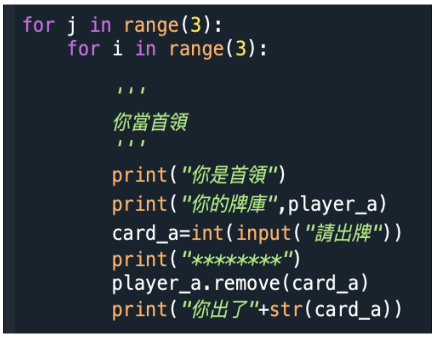


```
print("你是首領")
print("你的牌庫",player_a)
card_a=int(input("請出牌"))
print("********")
player_a.remove(card_a)
print("你出了"+str(card_a))

card_b=random.choice(player_b)
player_b.remove(card_b)
print("玩家2出了"+str(card_b))

card_c=random.choice(player_c)
player_c.remove(card_c)
print("玩家3出了"+str(card_c))

card_d=random.choice(player_d)
player_d.remove(card_d)
print("玩家4出了"+str(card_d))
print("---------")
```

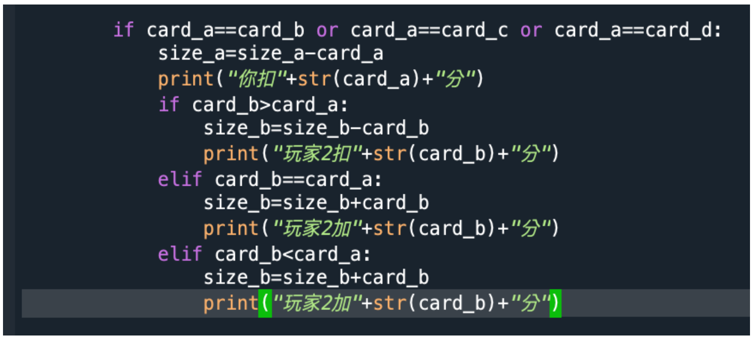
圖1 圖2

Next, we use the else if syntax to determine the card size of each cat as the leader cat and the normal cat, and record the bonus points and deductions in variables (Picture 3).

接下來我們用else if語法來判斷每隻貓當首領貓跟普通貓的卡牌大小情況並將加分扣分的情況記錄在變數裡（如圖3）。



```
for j in range(3):
    for i in range(3):

        '''
        你當首領
        '''
        print("你是首領")
        print("你的牌庫",player_a)
        card_a=int(input("請出牌"))
        print("********")
        player_a.remove(card_a)
        print("你出了"+str(card_a))
```



```
if card_a==card_b or card_a==card_c or card_a==card_d:
    size_a=size_a-card_a
    print("你扣"+str(card_a)+"分")
    if card_b>card_a:
        size_b=size_b-card_b
        print("玩家2扣"+str(card_b)+"分")
    elif card_b==card_a:
        size_b=size_b+card_b
        print("玩家2加"+str(card_b)+"分")
    elif card_b<card_a:
        size_b=size_b+card_b
        print("玩家2加"+str(card_b)+"分")
```

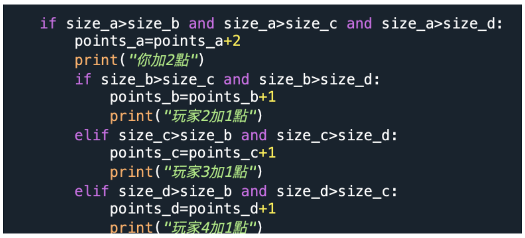
圖3 圖4

As there are 4 cats, it takes 4 rounds for each cat to take a turn as the leader. Each cat has 12 cards, so each cat will take a turn as the leader 3 times in each round, and there are 3 rounds in a game. Thus, we use two for loops that loop three times (Picture 4).

因為有4隻貓，每隻貓都輪過1次首領要4個回合，每隻貓有12張卡片，所以每個貓在每一輪會輪到3次首領，而且一局遊戲有3輪，於是我們使用兩個循環3次的for迴圈（如圖4）。

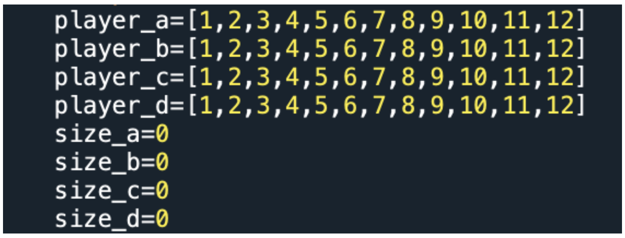
After each round, the points gained and lost need to be settled and points distributed, so I use the else if syntax at the end of the first for loop to settle and distribute the points (Picture 5) and refresh the card library and return it to zero. Score to start the next round (Picture 6).

在每一輪結束後需要結算拿到跟失去的分數和分配點數，所以我在第一個for迴圈的最後用else if語法結算與分配點數（如圖5）並刷新了牌庫跟歸零分數以便下一輪的開始（如圖6）。



```
if size_a>size_b and size_a>size_c and size_a>size_d:
    points_a=points_a+2
    print("你加2點")
    if size_b>size_c and size_b>size_d:
        points_b=points_b+1
        print("玩家2加1點")
    elif size_c>size_b and size_c>size_d:
        points_c=points_c+1
        print("玩家3加1點")
    elif size_d>size_b and size_d>size_c:
        points_d=points_d+1
        print("玩家4加1點")
```

圖5



```
player_a=[1,2,3,4,5,6,7,8,9,10,11,12]
player_b=[1,2,3,4,5,6,7,8,9,10,11,12]
player_c=[1,2,3,4,5,6,7,8,9,10,11,12]
player_d=[1,2,3,4,5,6,7,8,9,10,11,12]
size_a=0
size_b=0
size_c=0
size_d=0
```

圖6

Thoughts and conclusions：

We encountered some difficulties when researching and writing code, which allowed all of us to develop better team division of labor capabilities. We also strengthened the use of data structures when writing code and became more familiar with for loops and learning. More uses for the rondom module. We look forward to learning the use of tkinter modules together and creating more beautiful works in our next cooperation.

3.心得與結論：

在研究和撰寫程式碼的時候我們遇到了一些困難，這讓我們這組成員都培養了更好的團隊分工能力，在寫程式碼的時候也加強了資料結構的運用且更加熟悉for迴圈和學習了更多rondom模組的用法。我們期望在下次合作中一起學習tkinter模組的使用並做出更加精美的作品。