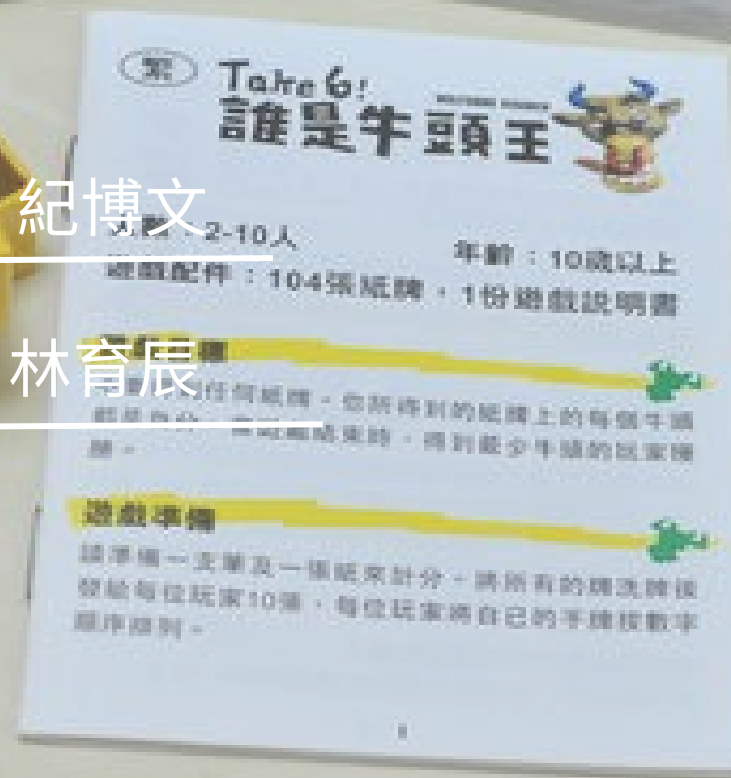
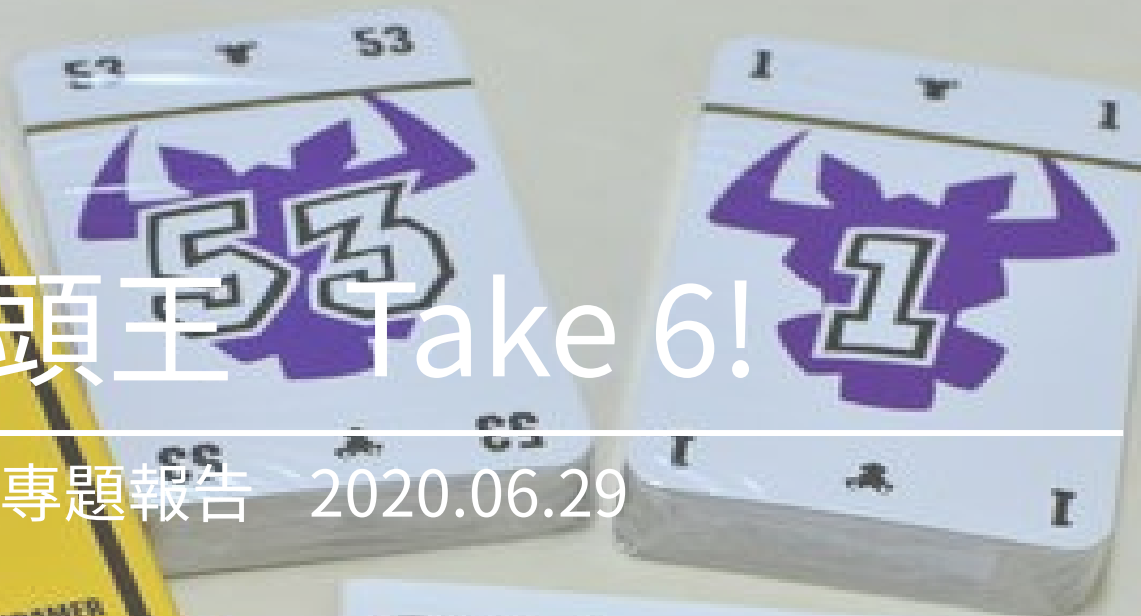


誰是牛頭王 Take 6!

程式設計 專題報告 2020.06.29

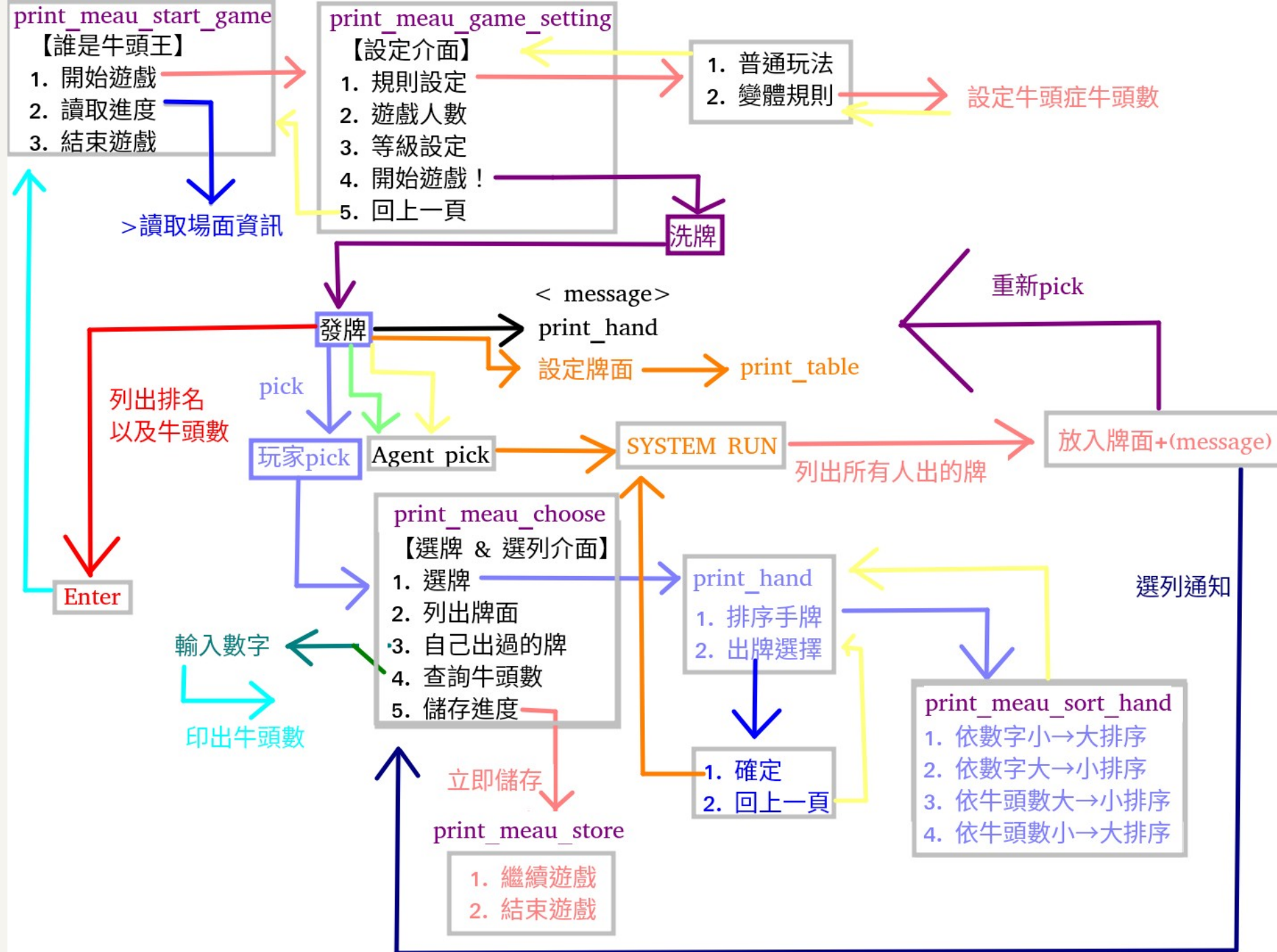
課堂教授：紀博文

資工 111 林育辰





流程圖





遊戲特色

遊戲特色

Take 6!

兩種
規則

- 牛頭數少→勝
- 牛頭數大於等於「標準」越多勝 + 原規則
- Random 不再強大，最笨的決策也會變聰明

存取
進度

- 單機遊戲必備功能
- 程式結束資料應該要保存
- 比起介面，此功能應是最優先的

```
【排名狀況——狂牛局】
| Rank 1 Player 4 Points: 59 | (感染者)
| Rank 2 Player 1 Points: 0 |
| Rank 2 Player 2 Points: 0 |
| Rank 2 Player 3 Points: 0 |
| Rank 2 Player 5 Points: 0 |
-----
>>>感染標準：30
Enter...
```

p.s 該局最後得分 59, 0, 0, 12, 29

【排名狀況——普通局】

	Rank 1	Player 1	Points: 8	
	Rank 2	Player 2	Points: 25	

Player1: 0.84				
Player2: 0.16				

p.s 1001 局 level 3 與 level 2 對戰

< 進度已儲存 >

【儲存介面】

1) 繼續遊戲

2) 結束遊戲

請輸入數字

Choose:

遊戲特色

Take 6!

排序
手牌

- 數字小→大
- 數字大→小
- 牛頭數大→小
- 牛頭數小→大

其它
功能

- 查詢功能
- 接受 2 人以上對局
- 只讓電腦對局
- 3 個等級 （可指定）

0) 玩家參賽: Yes

1) Agent level: 1

2) Agent level: 3

3) Agent level: 1

4) Agent level: 1

5) Agent level: 3

6) Agent level: 1

7) Agent level: 2

8) Agent level: 1

9) Agent level: 1

10) 回上一頁

請輸入想更動的編號

Choose:



0) 玩家參賽: Yes

1) Agent level: 1

2) Agent level: 3

3) Agent level: 1

4) Agent level: 1

5) Agent level: 3

6) Agent level: 1

7) Agent level: 2

8) Agent level: 1

9) Agent level: 1

10) 回上一頁

請輸入想更動的編號

Choose:

【手牌處理介面】

1) (主)手牌小→大

2) (主)手牌大→小

3) (主)牛頭數大→小／(次)手牌小→大

4) (主)牛頭數小→大／(次)手牌小→大

5) 回上一頁

請輸入數字

Choose:

程式結構 (等級)

Function pointer array

```
94 //function pointer
95 int32_t (*agent_pick[10])(int32_t[4][5], int32_t[10]);
96 int32_t (*agent_choose[10])(int32_t[4][5], int32_t);
97
```

程式結構 (等級)

```
862 //設定電腦級別
863 void set_agent_level(){
864     for (int32_t i = 0 ; i < 10 ; i++){
865         if (agent_level[i] == 0){
866             agent_pick[i] = User_pick;
867             agent_choose[i] = User_choose;
868         }else if (agent_level[i] == 1){
869             agent_pick[i] = Agent_pick_level_1;
870             agent_choose[i] = Agent_choose_level_1;
871         }else if (agent_level[i] == 2){
872             agent_pick[i] = Agent_pick_level_2;
873             agent_choose[i] = Agent_choose_level_2;
874         }else if (agent_level[i] == 3){
875             agent_pick[i] = Agent_pick_level_3;
876             agent_choose[i] = Agent_choose_level_3;
877         }
878         else
879             printf("error!\n");
880     }
881 }
```

```
94 //function pointer
95 int32_t (*agent_pick[10])(int32_t[4][5], int32_t[10]);
96 int32_t (*agent_choose[10])(int32_t[4][5], int32_t);
97
```

遇到的難題

- 存檔、讀檔 指標→ memory leak

遇到的難題

- 存檔、讀檔 指標→ memory leak
- 使用者介面撰寫→困難、很難寫的漂亮

遇到的難題

- 存檔、讀檔 指標→ memory leak
- 使用者介面撰寫→困難、很難寫的漂亮
- 手牌憑空消失、幽靈牌出沒

遇到的難題

- 存檔、讀檔 指標→ memory leak



指標問題細心即可，
多半是沒考量到某些情況才會出事

遇到的難題

- 使用者介面撰寫→困難、很難寫的漂亮



函式還是很醜，硬是寫出來
但很難維護，要 check 很多次

遇到的難題

- 手牌憑空消失、幽靈牌出沒



問題在於函式寫法不統一，
有些會清手牌有些沒有→搞混

未來方向

方向

- SDL
- 網頁→可供連線？
- 跟隨順喜老師持續發展 Agent

未來方向

方向

- SDL
- 網頁→可供連線？
- 跟隨順喜老師持續發展 Agent

專題能延伸的方向很多！
遠比當初構想時還不同



Thanks for your patience

誰是牛頭王 Take6!

資工 111 林育辰