程式設計實作 終極密碼戰

系級:醫資一學號:413570012 學生:林宜庭

壹、整體程式碼

```
default:
       System.out.println("無法辨識您的輸入,預設為簡單難度");
       break;
Scanner s2 = new Scanner(System.in);
Random r = new Random();
int answerNum = r.nextInt(bound: maxNum-minNum); //隨機在區間產生數字答案
while(true){
    System.out.println("請輸入一個數字(介於" + minNum + "到" + maxNum + "): ");
    int guessNum = s2.nextInt();
    if(guessNum < minNum || guessNum > maxNum) {
       System.out.println("請重新輸入,介於正確範圍內的數字");
    maxAttempts--; //猜測次數-1
    if(guessNum == answerNum){
       System.out.println("你猜對了! 正確答案是" + answerNum);
       break; //結束迴圈
    }else if (guessNum < answerNum){</pre>
       System.out.println("你猜的數字比正確答案小");
       \underline{minNum} = guessNum;
     System.out.println("你猜的數字比正確答案大");
    maxNum = guessNum;
 if (maxAttempts == 0) {
```

貳、執行結果

(一)遊戲流程

1.遊戲開始輸入1選擇簡單難度

```
請選擇難度:
1.簡單(範圍1-50,10次機會)
2.中等(範圍1-100,7次機會)
3.困難(範圍1-200,5次機會)
1
請輸入一個數字(介於1到50):
33
你猜的數字比正確答案小
請輸入一個數字(介於33到50):
45
你猜的數字比正確答案大
請輸入一個數字(介於33到45):
35
你猜對了! 正確答案是35
```

2.回答正確跳出對話視窗選擇是否再來一局,選是



3.遊戲重新開始,選擇中等難度

請選擇難度: 1. 簡單(範圍1-50,10次機會) 2.中等(範圍1-100,7次機會) 3.困難(範圍1-200,5次機會) 2 請輸入一個數字(介於1到100): 55 你猜的數字比正確答案小 請輸入一個數字(介於55到100) 77 你猜的數字比正確答案大 請輸入一個數字(介於55到77): 66 你猜的數字比正確答案小 請輸入一個數字(介於66到77): 69 你猜的數字比正確答案小 請輸入一個數字(介於69到77): 74 你猜的數字比正確答案小 請輸入一個數字(介於74到77): 76 你猜的數字比正確答案大 請輸入一個數字(介於74到76): 75 你猜對了! 正確答案是75

4.回答正確選再來一局,選擇困難難度,五次未猜得答案後公佈答案



請選擇難度: 1. 簡單(範圍1-50,10次機會) 2.中等(範圍1-100,7次機會) 3.困難(範圍1-200,5次機會) 3 請輸入一個數字(介於1到200): 100 你猜的數字比正確答案大 請輸入一個數字(介於1到100): 55 你猜的數字比正確答案小 請輸入一個數字(介於55到100): 77 你猜的數字比正確答案大 請輸入一個數字(介於55到77): 66 你猜的數字比正確答案大 請輸入一個數字(介於55到66): 60 你猜的數字比正確答案小 你已經用完所有的猜測次數! 正確答案是 62

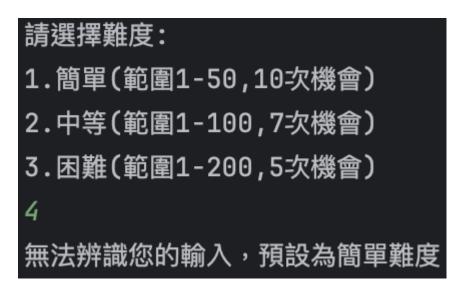
5.輸入n結束遊戲



Game over
Process finished with exit code 0

(二)其他測試與debug

1.輸入1至3以外的數字直接預設為簡單難度



2.簡單難度10次未猜得後公佈答案

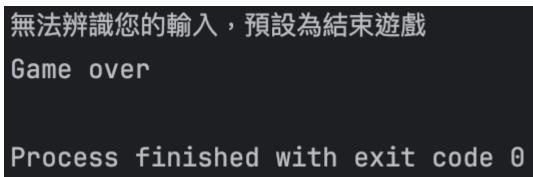
```
請輸入一個數字(介於1到50):
1
你猜的數字比正確答案小
請輸入一個數字(介於1到50):
你猜的數字比正確答案小
請輸入一個數字(介於1到50):
1
你猜的數字比正確答案小
請輸入一個數字(介於1到50):
你猜的數字比正確答案小
請輸入一個數字(介於1到50):
你猜的數字比正確答案小
請輸入一個數字(介於1到50):
你猜的數字比正確答案小
請輸入一個數字(介於1到50):
1
你猜的數字比正確答案小
請輸入一個數字(介於1到50):
你猜的數字比正確答案小
請輸入一個數字(介於1到50):
你猜的數字比正確答案小
請輸入一個數字(介於1到50):
1
你猜的數字比正確答案小
你已經用完所有的猜測次數! 正確答案是 31
```

3.中等難度7次未猜得後公佈答案

```
請選擇難度:
1. 簡單(範圍1-50,10次機會)
2.中等(範圍1-100,7次機會)
3.困難(範圍1-200,5次機會)
2
請輸入一個數字(介於1到100):
1
你猜的數字比正確答案小
請輸入一個數字(介於1到100):
1
你猜的數字比正確答案小
請輸入一個數字(介於1到100):
你猜的數字比正確答案小
請輸入一個數字(介於1到100):
1
你猜的數字比正確答案小
請輸入一個數字(介於1到100):
1
你猜的數字比正確答案小
請輸入一個數字(介於1到100):
你猜的數字比正確答案小
請輸入一個數字(介於1到100):
1
你猜的數字比正確答案小
你已經用完所有的猜測次數! 正確答案是 19
```

4.輸入y或n以外的字元預設結束遊戲





參、學習心得

(一)撰寫歷程

• 再來一局

詢問ChatGPT while迴圈的概念以正確了解原有程式 碼所需修改的部分,並利用while、彈跳視窗與if語法 進行撰寫。

● 困難度選擇

首先想到將遊戲內容設為method, 讓使用者選擇困難度後再挪用method, 於是輸入function請ChatGPT撰寫。而後指示ChatGPT撰寫困難選擇功能, 並追問其中的switch語法。

debug

應用ChatGPT所撰寫的「無效的選擇,預設為簡單難度」概念以及ICE提示的!answer.equals至再來一局功能。

(二)結果討論

該程式碼成功添加再來一局與困難選擇功能,可惜的是未提示猜測次數減少,使用ChatGPT所撰寫的會造成顯示錯誤,我未查訊到相關網路資源,斟酌過後放棄進行添加。

肆、參考資料

ChatGPT對話歷程