

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

```
1 import javax.swing.*;
2 import java.util.Random;
3 import java.util.Scanner;
4
5
6 public class Main {
7     public static void playgame(){ //宣告playgame為method 1 usage
8         Scanner s1 = new Scanner(System.in);
9         System.out.println("請選擇難度:\n1. 簡單(範圍1-50,10次機會)\n2. 中等(範圍1-100,7次機會)\n3. 困難(範圍1-200,5次機會)");
10        int difficulty = s1.nextInt(); //讀取整數存於difficulty
11        int minNum = 1;
12        int maxNum = 50;
13        int maxAttempts = 10; //宣告猜測次數
14        switch (difficulty) {
15            case 1: //簡單難度
16                break;
17            case 2: //中等難度
18                maxNum = 100;
19                maxAttempts = 7;
20                break;
21            case 3: //困難難度
22                maxNum = 200;
23                maxAttempts = 5;
24                break;
```

```

25         default:
26             System.out.println("無法辨識您的輸入，預設為簡單難度");
27             break;
28     }
29     Scanner s2 = new Scanner(System.in);
30     Random r = new Random();
31     int answerNum = r.nextInt(bound: maxNum-minNum); //隨機在區間產生數字答案
32
33
34     while(true){
35         System.out.println("請輸入一個數字(介於" + minNum + "到" + maxNum + "): ");
36         int guessNum = s2.nextInt();
37         if(guessNum < minNum || guessNum > maxNum) {
38             System.out.println("請重新輸入，介於正確範圍內的數字");
39         }
40         maxAttempts--; //猜測次數-1
41         if(guessNum == answerNum){
42             System.out.println("你猜對了! 正確答案是" + answerNum);
43             break; //結束迴圈
44         }else if (guessNum < answerNum){
45             System.out.println("你猜的數字比正確答案小");
46             minNum = guessNum;

```

```

47         }else {
48             System.out.println("你猜的數字比正確答案大");
49             maxNum = guessNum;
50         }
51         if (maxAttempts == 0) {
52             System.out.println("你已經用完所有的猜測次數! 正確答案是 " + answerNum);
53             break;
54         }
55     }
56 }
57 ▶ public static void main(String[] args){
58     while(true){
59         playgame(); //執行method
60         String answer = JOptionPane.showInputDialog("是否再來一局?(y/n)");
61         //顯示彈跳對話視窗，提示使用者輸入y或n選擇再一局或結束遊戲
62         if (answer.equals("n")){
63             break;
64         }
65         else if(!answer.equals("y")){ //如果輸入不為y或n
66             System.out.println("無法辨識您的輸入，預設為結束遊戲");
67             break;
68         }
69     }

```

```
70         System.out.println("Game over");  
71     }  
72 }
```

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

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)

4

無法辨識您的輸入，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):

1

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):

1

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):

1

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):

1

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):

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):

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):

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):

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



```
無法辨識您的輸入，預設為結束遊戲  
Game over  
  
Process finished with exit code 0
```

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](#)