

# 心理與神經資訊學

## (Psychoinformatics & Neuroinformatics)

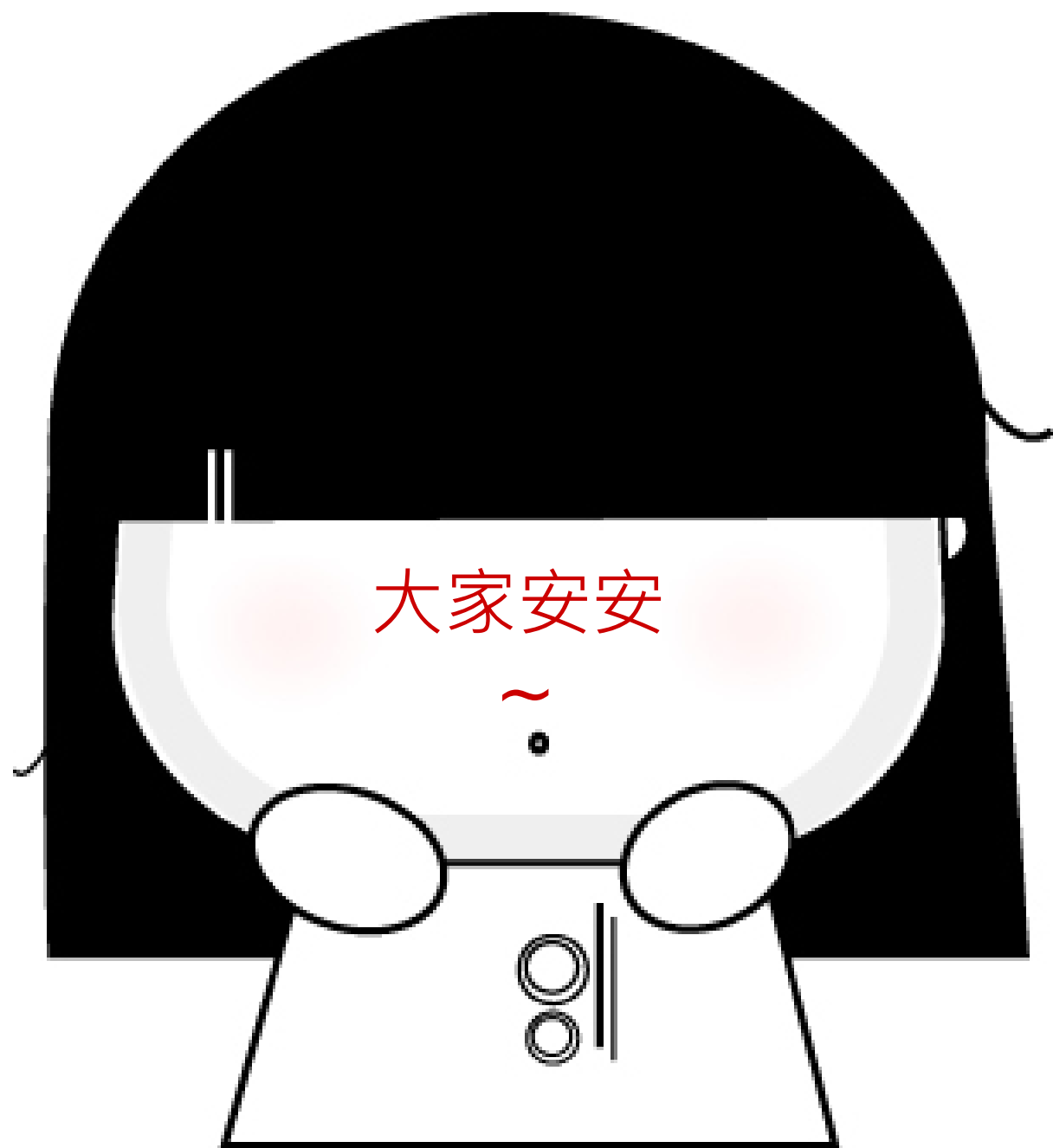
課號: Psy5261

教室: 綜合302

識別碼: 227U9340

時間: 五234





# 基本程式設計 (Python 3.X)

# 心理學案例研究(1/4)

實驗室有研究助理專責寫實驗程式

研究者來回請助理改程式小細節  
(如圖片在電腦螢幕上的大小, 位置, 與時間長短)

研究者的內心話: 怎麼改個東西這麼慢?

助理的內心話: 怎麼改來改去這麼囉唆?

怎樣讓這兩個人都快樂?

# 心理學案例研究(2/4)

老師請學生去做個問卷調查

學生蒐集完一堆問卷後打開Excel默默地key in data

老師接著請學生用SPSS做資料分析

學生發現SPSS要某種資料格式，  
只好又打開Excel默默地複製/貼上

老師的內心話:怎麼做個小研究手腳這麼慢?

學生的內心話:人生太無趣，我要休學了!

**怎樣讓這兩個人都快樂?**

# 心理學案例研究(3/4)

小美聽說人可以在0.1秒內辨識場景，決定研究看看



● 圖庫共有1000張圖

她用E-Prime製作實驗程式時選取了1000次檔名!

**如果老師說要換成另一組圖片怎麼辦?**

# 心理學案例研究(4/4)

小美實驗了**10**個人，其中每個受試者被測了**3**種條件

統計分析後，發現實驗操弄似乎有效果，但卻不顯著

老師建議再做**40**個人，增加統計檢定力(power)

但每新收一筆資料，就要**手動**匯入一次並重新做統計

**有沒有辦法將此過程自動化？**

# 為何要學程式設計？

## 對於高重複性的運算/動作：

- 你會累但電腦不會累(但兩者都會壞)
- 你會錯但電腦不會錯(除非你誤導電腦)



## 自動化例行公事後就有時間去玩了：

- 程式幫同學讀論文(真心不騙)
- 程式幫老師寫論文(神之作)



## 邊玩的時候電腦還繼續幫你收資料和賺錢：

- 網頁實驗不打烊(我在墾丁天氣晴)
- 程式交易(還需要工作嗎)

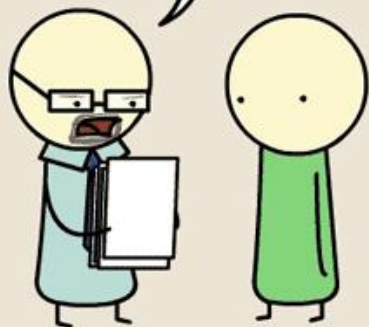




# 程式語言何其多

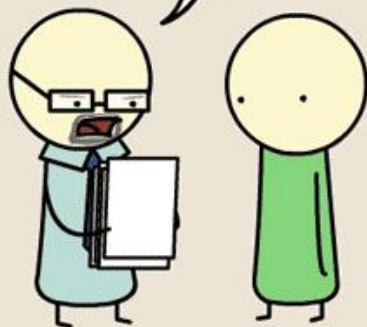
## PYTHON

THIS IS PLAGIARISM.  
YOU CAN'T JUST "IMPORT ESSAY."



## JAVA

I'M TWO PAGES IN AND I STILL  
HAVE NO IDEA WHAT YOU'RE SAYING.



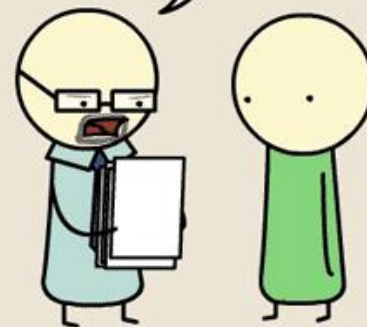
## C++

I ASKED FOR ONE COPY,  
NOT FOUR HUNDRED.



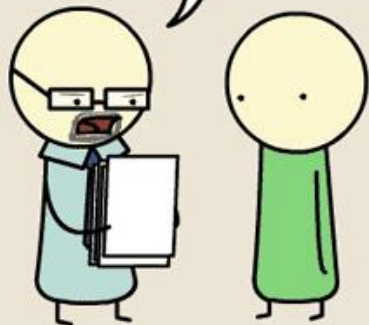
## UNIX SHELL

I DON'T HAVE PERMISSION TO  
READ THIS.



## ASSEMBLY

DID YOU REALLY HAVE TO REDEFINE EVERY  
WORD IN THE ENGLISH LANGUAGE?



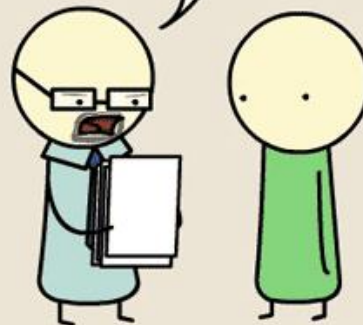
## C

THIS IS GREAT, BUT YOU FORGOT TO ADD  
A NULL TERMINATOR. NOW I'M JUST READING  
GARBAGE.



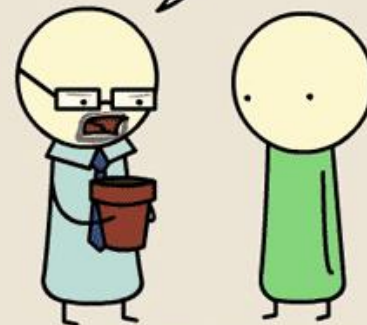
## LATEX

YOUR PAPER MAKES NO GODDAMN SENSE,  
BUT IT'S THE MOST BEAUTIFUL THING  
I HAVE EVER LAID EYES ON.



## HTML

THIS IS A FLOWER POT.



# 低階vs.高階語言

## JAVA

```
public class HelloWorld
{
    public static void main (String[] args)
    {
        System.out.println("Hello, world!");
    }
}
```

```
import java.io.*;
...
BufferedReader myFile =
    new BufferedReader(
        new FileReader(argFilename));
```

## PYTHON

```
print("Hello, world!")
```

```
MyFile = open(argFilename)
```

[這](#)有更多比較

# Python程式設計懶人包

我們需要的幾個核心元素是：

1. 基本數學/邏輯運算與變數存取

2. 內/自建函數(functions)

3. 條件判斷(if和else)

4. 迴圈敘述(for或while)

**for:** 知道要重複幾次或具體項目

**while:** 不知道要重複幾次，但知道停止的條件



# 基本數學/邏輯運算與變數存取

Try:

$a = 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10$

$b = (1 + 10) * 10 / 2$

$a$

$b$

$a > b$

$a == b$

$(a > 10) \text{ and } (b > 20)$



好文共享:[我的數學夢之告解數學](#) by 臺大心理江怡廷

# 資料的形態與轉換

```
type(100) #int
type(100.0) #float
type(float(100)) #float
type(int(100.0)) #int
type('Hello') #str
type("Hello") #str
type(range(101)) #range
float(range(101)) #TypeError
a=[33,7,18]
a[0],'Hello'+str(a[2]),a[1]+3 #33 Hello18 10
a+[13] #[33, 7, 18, 13]
a+13 #TypeError
```



**Python**還有集合，不重複隨機選取的實驗裡常用！

# Python中的大、中、小括號

- 大括號:定義dictionary(即[hash table](#))

```
tel={'suling': 63097, 'tren': 63104}
```

```
list(tel.keys()) #['suling', 'tren']
```

```
list(tel.values()) #[63097,63104]
```

```
tel['tren'] #63104
```

- 中括號:存取List

```
a=range(5,0,-1)
```

```
a[1] #4
```



- 小括號:決定運算順序或呼叫函數



# 什麼是函數？

**Input**



這傢伙不見得會吐錢

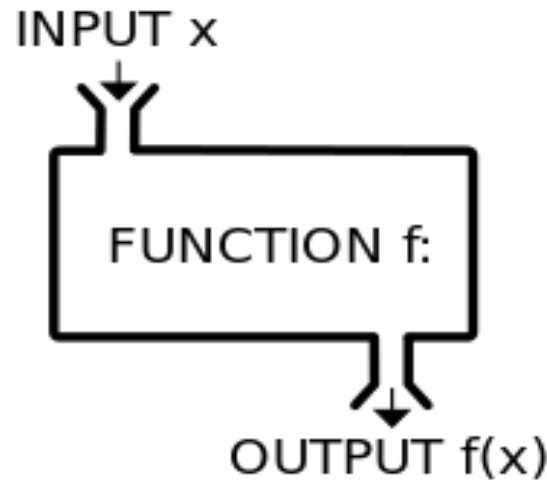
**Output**



# 內建函數

**Try:**

```
a=range(1,11)
b=sum(a)/len(a)
c=1
c=c+1
print(a,b,c)
import math
math.factorial(5)
math.pi
import random
round(10*random.random())
```



c=c+1

~~c=c+1~~

0=1???

$$\frac{1}{n} \sin x = ?$$

$$\frac{1}{\cancel{n}} \cancel{\sin} x = ?$$

$$\text{six} = 6 ?$$



# 怎麼知道有神馬函數可用？

長大自然就懂了!?

例1: [python 亂數](#)

例2: [python 畫圖](#)

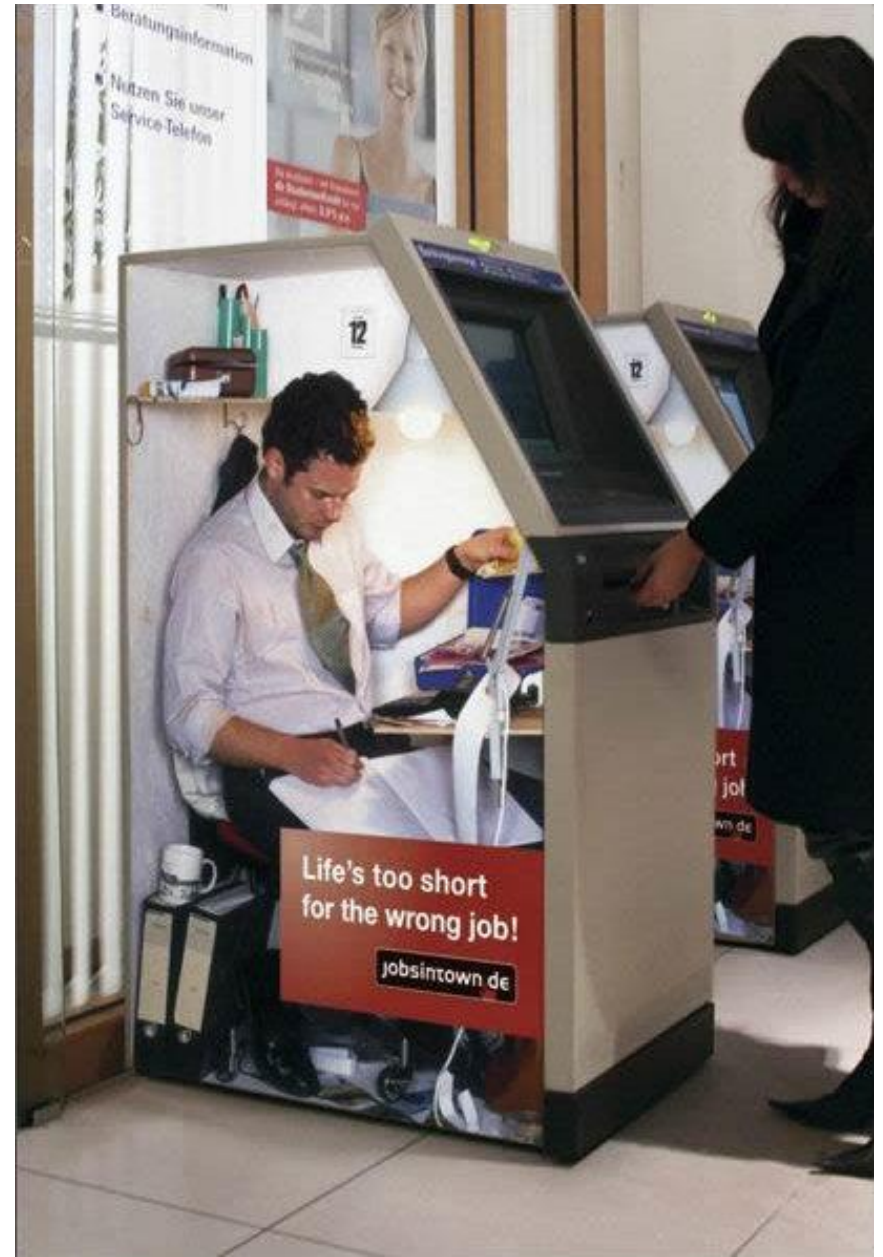
例3: [python statistics](#)

例4: [python tetris](#)

例5: [python mario](#)

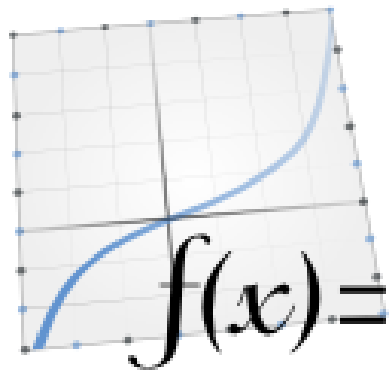


# 自建函數



# 自建函數

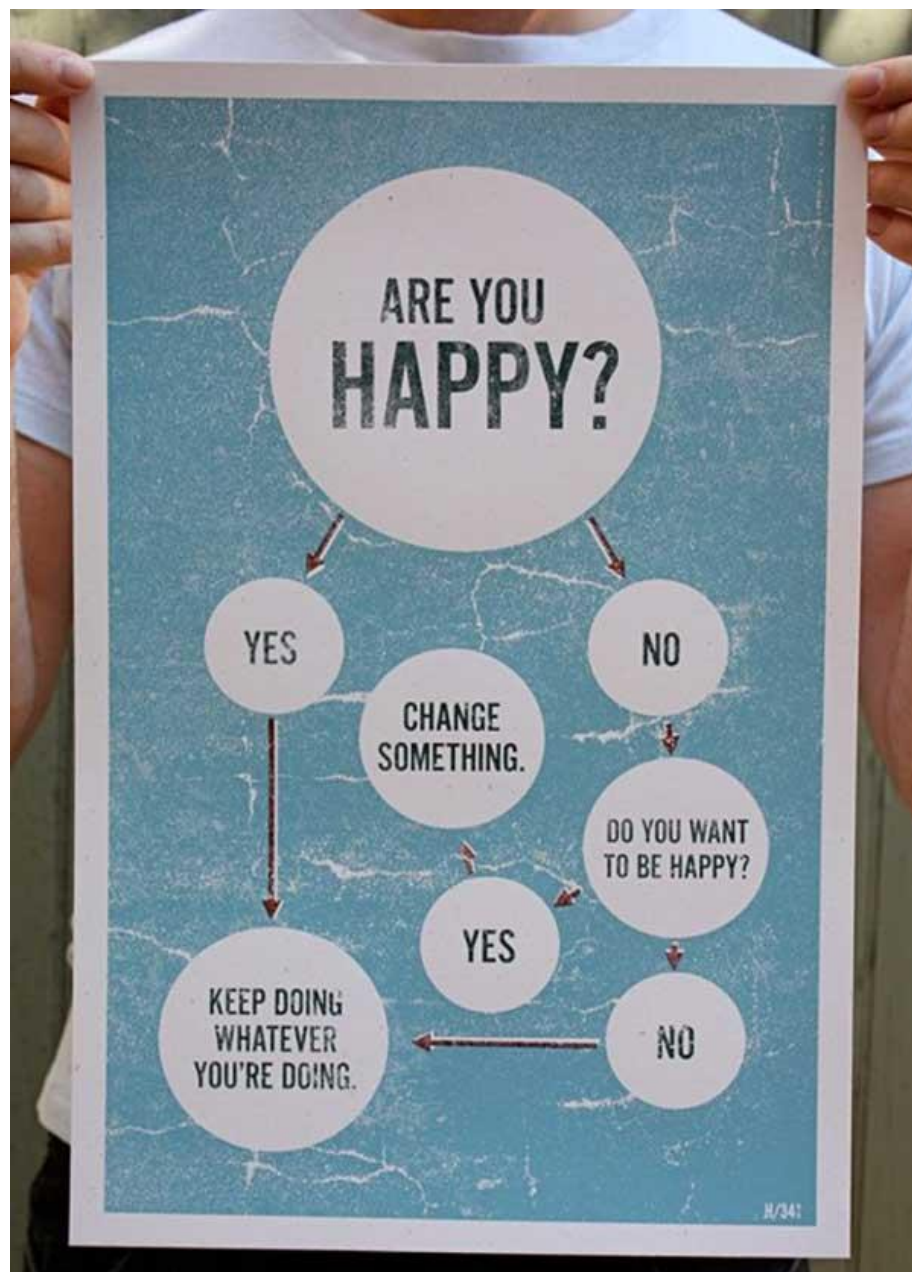
**Try:** (注意縮排用來告訴**Python**從屬關係)



```
import math
def adjust_score(old):
    new=math.sqrt(old)*10
    return new
```

```
a=adjust_score(0)
b=adjust_score(60)
print(a,b)
```

# 條件判斷





# 條件判斷

**Try:** (注意縮排)

```
import random
r=random.random()
if(r>0.5):
    print("Version A")
else:
    print("Version B")
```



這就是業界常說的 **A/B testing**

# 迴圈敘述for

計算 $1+2+3...+k$

凡人的寫法：(注意縮排)

```
k=int(input("k="))  
s=0  
for i in range(1,k+1):  
    s=s+i  
    print(i,s)
```

高斯的寫法：(難理解)

```
k=int(input("k="))  
print((1+k)*k/2)
```



電腦是什麼?  
能吃嗎?

# 解惑時間：i 深陷輪迴要投胎

```
for i in [5566,'never dies',range(3)]:  
    print(i)
```

```
5566  
never dies  
range(0,3)
```



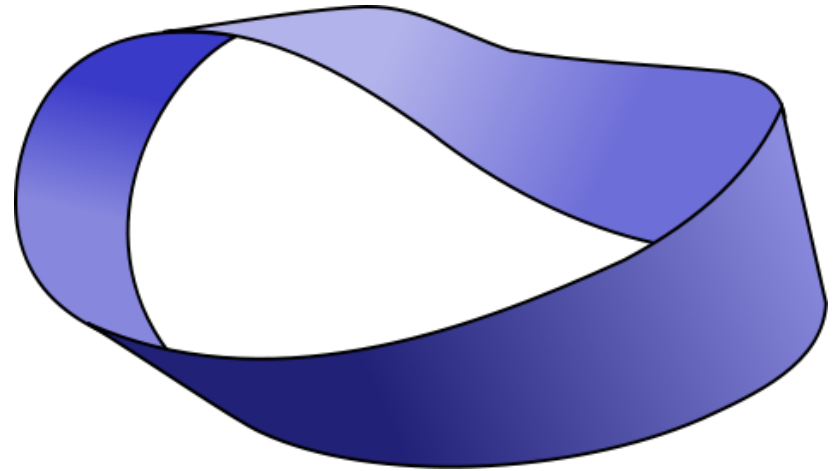
```
for i in range(3):  
    print("I don't care i")
```

```
I don't care i  
I don't care i  
I don't care i
```

# 迴圈敘述while

**Try:** (注意縮排)

```
import random
c=0 #counter
r=0
while(r<0.9):
    c=c+1
    r=random.random()
    print(c,r)
```



**應用:** 讓受試者練習到正確率90%以上才正式實驗



# 反應時間測量



```
import time, random
print("Get ready...", flush=True)
time.sleep(10 * random.random())
t0 = time.time()
input("Press [Enter] now!")
print(time.time() - t0)
```

# 本週作業

## 修改反應時間測量程式

1. 將隨機時間從[0,10]改成[5,10]。
2. 將測量次數從1次改成3次;  
全部測量都結束後才印出平均反應時間，如:

Get ready...

Press [Enter] now!

Get ready...

Press [Enter] now!

Get ready...

Press [Enter] now!

0.3725



GAME Over

