MEG304_MID :: 基於Python的資訊素養入門

> Date: 2021-Nov-24 R00.
>
> \>

MEG304 Stuent.ID: _ Name: _ PC seat Number :_

>

TOC :: MEG304:::

- Level 1 -QUIZ - - Level 2 -GOAL -- END-Of-SHEET

LEVEL 1 : TWO-DICES GAME

運行（最多）**10**次遊戲以獲得兩個骰子的總和。如果第**n**次試驗的總和小於**6**，則遊戲結束（退出）

> Note: Define some functions to achive the LEVEL-1 goal.

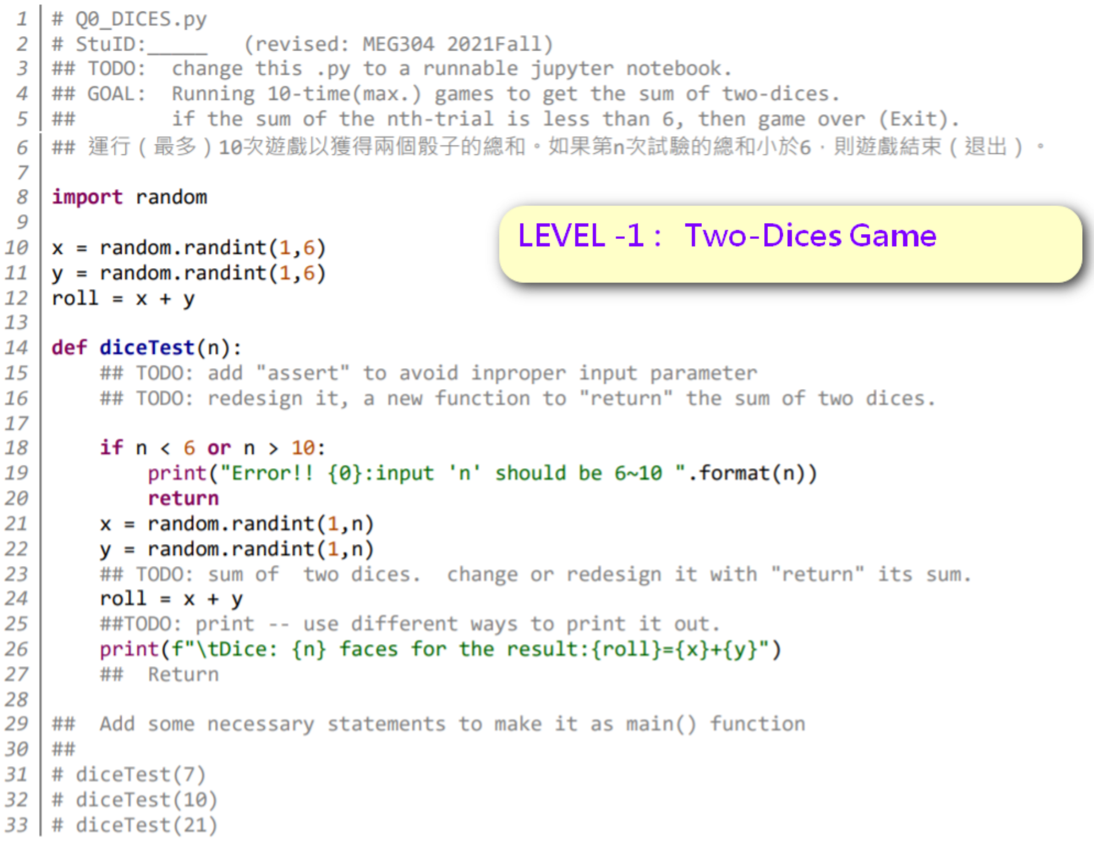
GAME GOAL: 兩個骰子的總和

In [1]:

```
## GOAL - template ( change the outfile name if needed)
# A sample short code to display this marked image file (at local subfolder )
## MEG304::  You need to use the existed Q0_DICES.py as the starting point.
##          The Expected DELIVERABLE (Answer sheet) is a jupyter notebook file.

from IPython import display
#use this ##  display.Image("./img/FIG1-Addimage.png")
display.Image("./img/MEGLevel1.png")
```

Out[1]:



LEVEL -1 : Two-Dices Game

```python
# Q0_DICES.py
# StuID:_____    (revised: MEG304 2021Fall)
## TODO:  change this .py to a runnable jupyter notebook.
## GOAL:  Running 10-time(max.) games to get the sum of two-dices.
##         if the sum of the nth-trial is less than 6, then game over (Exit).
## 運行（最多）10次遊戲以獲得兩個骰子的總和。如果第n次試驗的總和小於6，則遊戲結束（退出）。

import random

x = random.randint(1,6)
y = random.randint(1,6)
roll = x + y

def diceTest(n):
    ## TODO: add "assert" to avoid inproper input parameter
    ## TODO: redesign it, a new function to "return" the sum of two dices.

    if n < 6 or n > 10:
        print("Error!! {0}:input 'n' should be 6~10 ".format(n))
        return
    x = random.randint(1,n)
    y = random.randint(1,n)
    ## TODO: sum of  two dices.  change or redesign it with "return" its sum.
    roll = x + y
    ##TODO: print -- use different ways to print it out.
    print(f"\tDice: {n} faces for the result:{roll}={x}+{y}")
    ##  Return

##  Add some necessary statements to make it as main() function
##
# diceTest(7)
# diceTest(10)
# diceTest(21)
```

In [2]:

```python
### Your LEVEL-1 Code is here ...
```

In [ ]:

In [ ]:

In [ ]:

End of Level-1 Block

LEVEL 2 : 曼哈頓距離

In [3]:

```python
##  Note: Two differnt data sets are needed to prove your skill level
## ( including the Predefined-set1 as FIG1)
```

In [4]:

```python
##  Level2- FIG1 is here ...
# display.Image("./img/FIG1-Addimage.png")
display.Image("./img/Level2FIG1.png")
```

Out[4]:

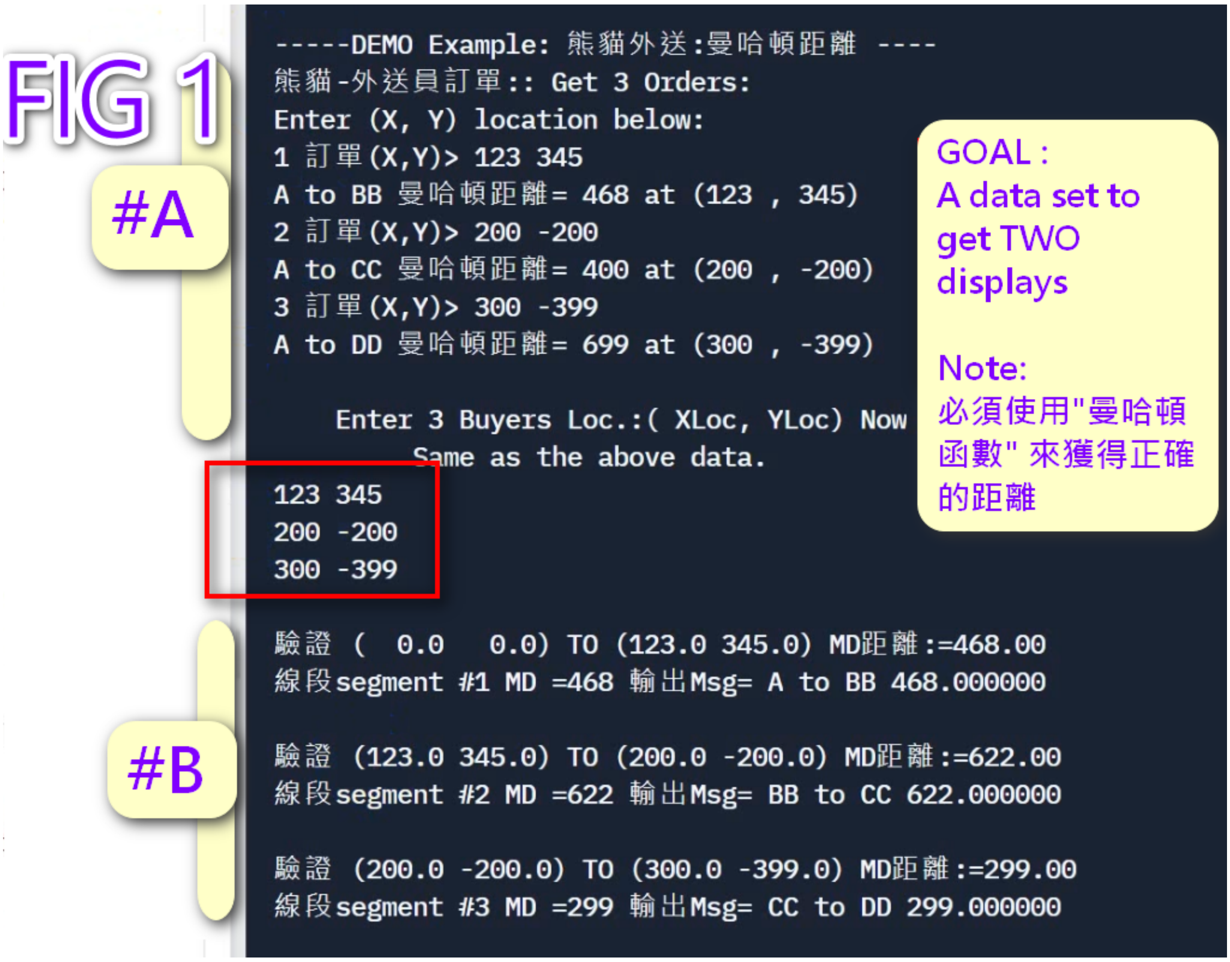


FIG 1

#A

```
-----DEMO Example: 熊貓外送:曼哈頓距離 ----
熊貓-外送員訂單:: Get 3 Orders:
Enter (X, Y) location below:
1 訂單(X,Y)> 123 345
A to BB 曼哈頓距離= 468 at (123 , 345)
2 訂單(X,Y)> 200 -200
A to CC 曼哈頓距離= 400 at (200 , -200)
3 訂單(X,Y)> 300 -399
A to DD 曼哈頓距離= 699 at (300 , -399)

    Enter 3 Buyers Loc.:( XLoc, YLoc) Now
          Same as the above data.
123 345
200 -200
300 -399
```

#B

```
驗證 (   0.0    0.0) TO (123.0 345.0) MD距離:=468.00
線段segment #1 MD =468 輸出Msg= A to BB 468.000000

驗證 (123.0 345.0) TO (200.0 -200.0) MD距離:=622.00
線段segment #2 MD =622 輸出Msg= BB to CC 622.000000

驗證 (200.0 -200.0) TO (300.0 -399.0) MD距離:=299.00
線段segment #3 MD =299 輸出Msg= CC to DD 299.000000
```

GOAL :
A data set to get TWO displays

Note:
必須使用"曼哈頓函數" 來獲得正確的距離

In [5]:

```
## 基於Python的資訊素養入門 --- your code /  brief  text-block (MD-Style) is here...
```

In [6]:

```
#  GOAL:  Level 2 Figure 2
# display.Image("./img/FIG1-Addimage.png")
display.Image("./img/Level2FIG2.png")
```

Out[6]:

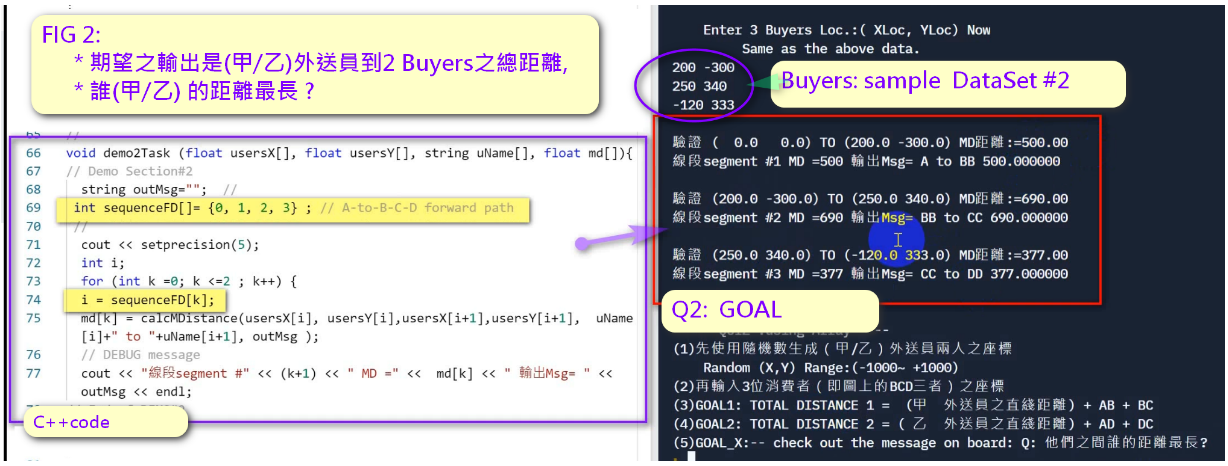


FIG 2:
* 期望之輸出是(甲/乙)外送員到2 Buyers之總距離,
* 誰(甲/乙) 的距離最長？

```
void demo2Task (float usersX[], float usersY[], string uName[], float md[]){
// Demo Section#2
  string outMsg="";  //
 int sequenceFD[]= {0, 1, 2, 3} ; // A-to-B-C-D forward path
 //
  cout << setprecision(5);
  int i;
  for (int k =0; k <=2 ; k++) {
  i = sequenceFD[k];
  md[k] = calcMDistance(usersX[i], usersY[i],usersX[i+1],usersY[i+1],  uName
  [i]+" to "+uName[i+1], outMsg );
  // DEBUG message
  cout << "線段segment #" << (k+1) << " MD =" <<  md[k] << " 輸出Msg= " <<
  outMsg << endl;
```

C++code

```
    Enter 3 Buyers Loc.:( XLoc, YLoc) Now
        Same as the above data.
200 -300
250 340
-120 333

驗證 (  0.0   0.0) TO (200.0 -300.0) MD距離:=500.00
線段segment #1 MD =500 輸出Msg= A to BB 500.000000

驗證 (200.0 -300.0) TO (250.0 340.0) MD距離:=690.00
線段segment #2 MD =690 輸出Msg= BB to CC 690.000000

驗證 (250.0 340.0) TO (-120.0 333.0) MD距離:=377.00
線段segment #3 MD =377 輸出Msg= CC to DD 377.000000

(1)先使用隨機數生成 ( 甲/乙 ) 外送員兩人之座標
    Random (X,Y) Range:(-1000~ +1000)
(2)再輸入3位消費者 ( 即圖上的BCD三者 ) 之座標
(3)GOAL1: TOTAL DISTANCE 1 =  (甲  外送員之直綫距離) + AB + BC
(4)GOAL2: TOTAL DISTANCE 2 = ( 乙  外送員之直綫距離) + AD + DC
(5)GOAL_X:-- check out the message on board: Q: 他們之間誰的距離最長?
```

Buyers: sample DataSet #2

Q2: GOAL

In [7]:

```
## --- your code /  brief  text-block (MD-Style) is here...
```

In [8]:

```
##  add your own code here ...
```

Back to :: Level 1 -QUIZ or - Level 2 -GOAL

In [9]:

```
## --- your code /  brief  text-block (MD-Style) is here...
```