通訊一 112503005 張采凝

112 /11/02

通訊一 112503505 蘇芸筠

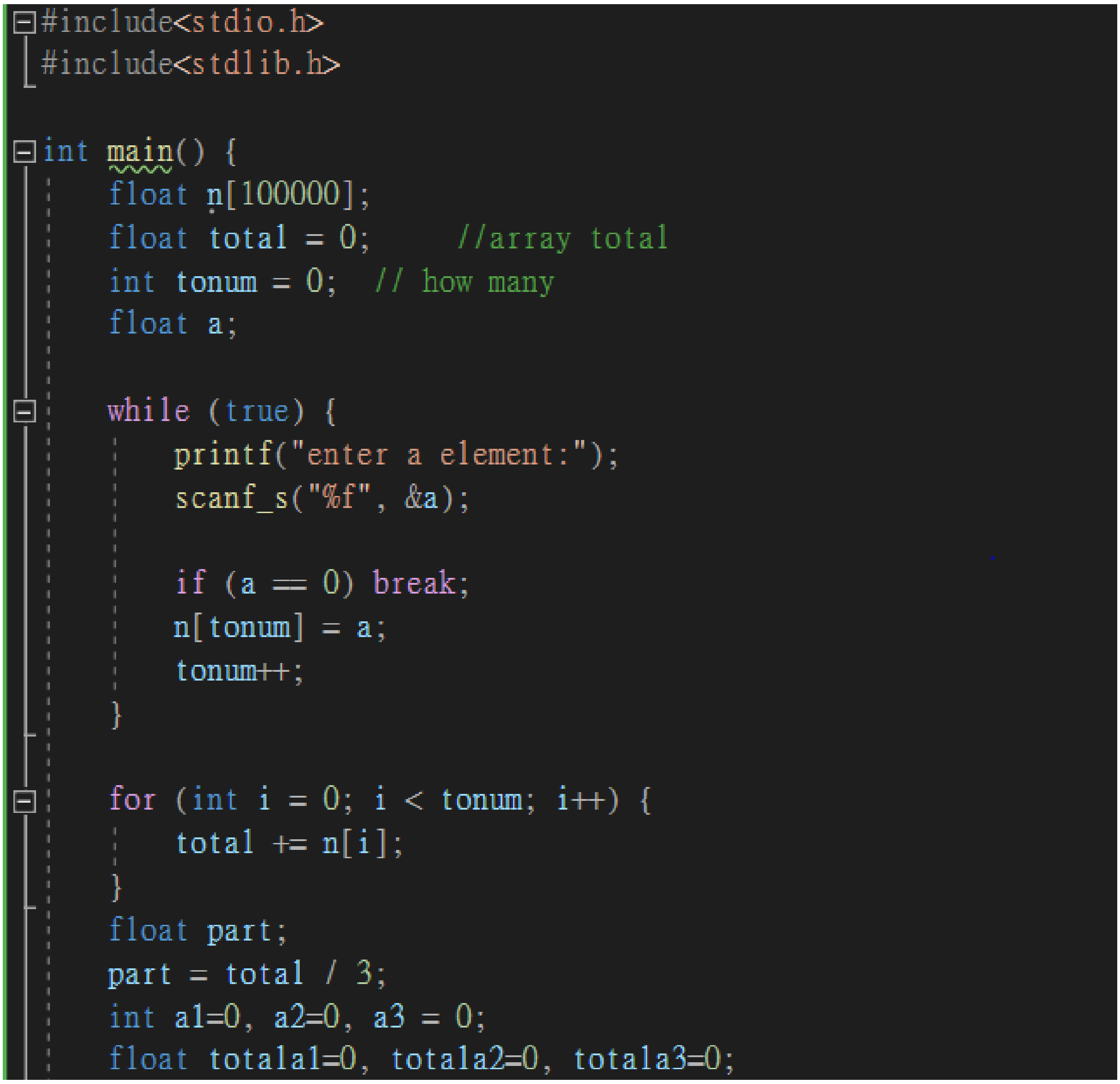
112 /11/02

5.1 C

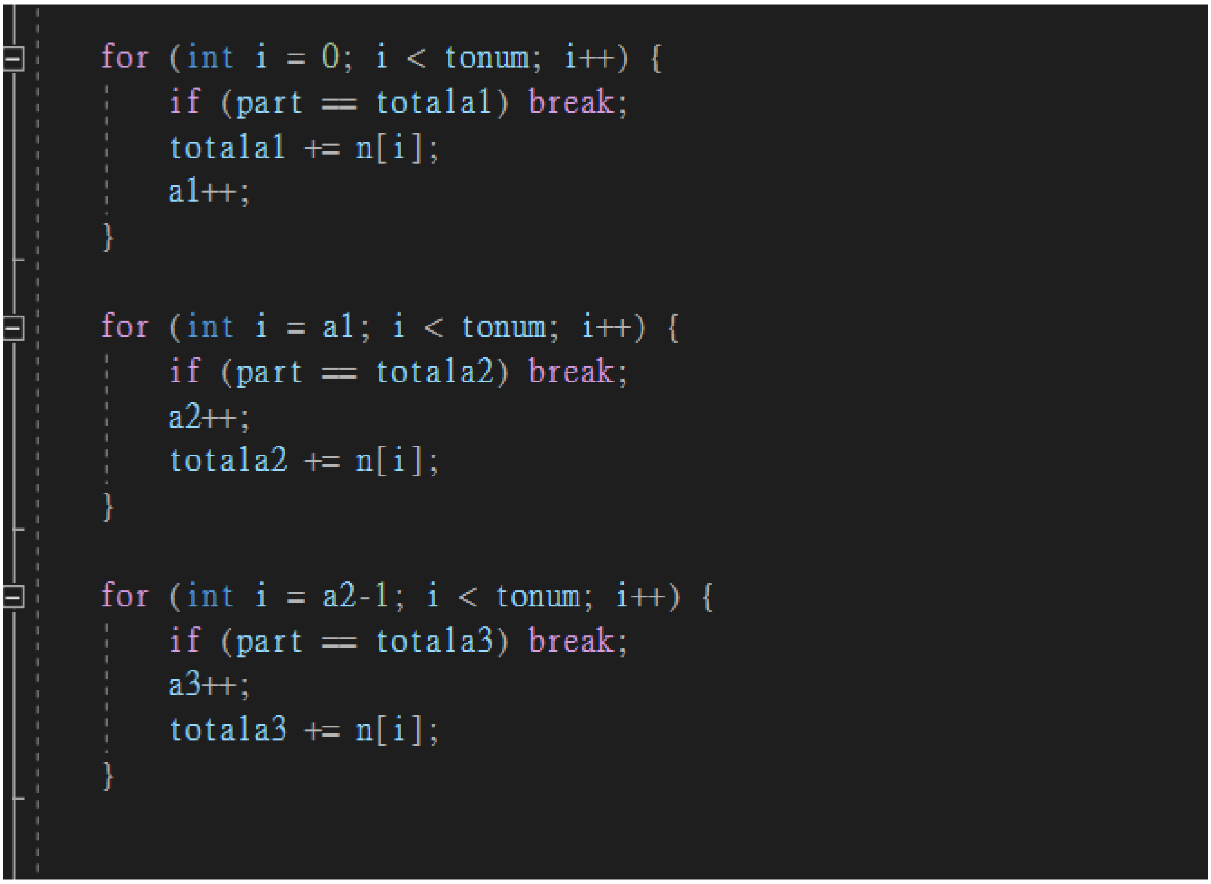
5.2.1 Python

5.2.2 Python

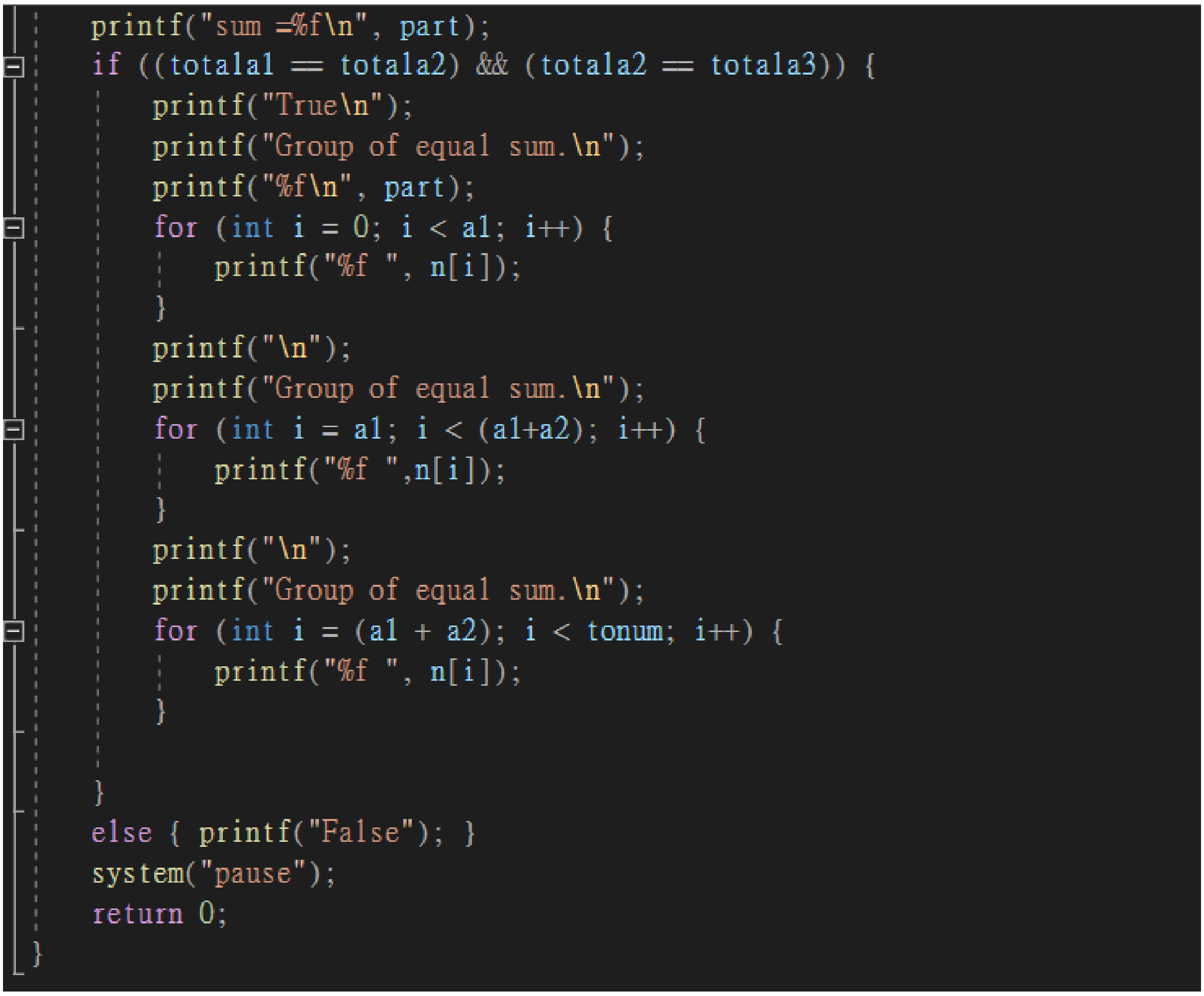
5.1 C demo



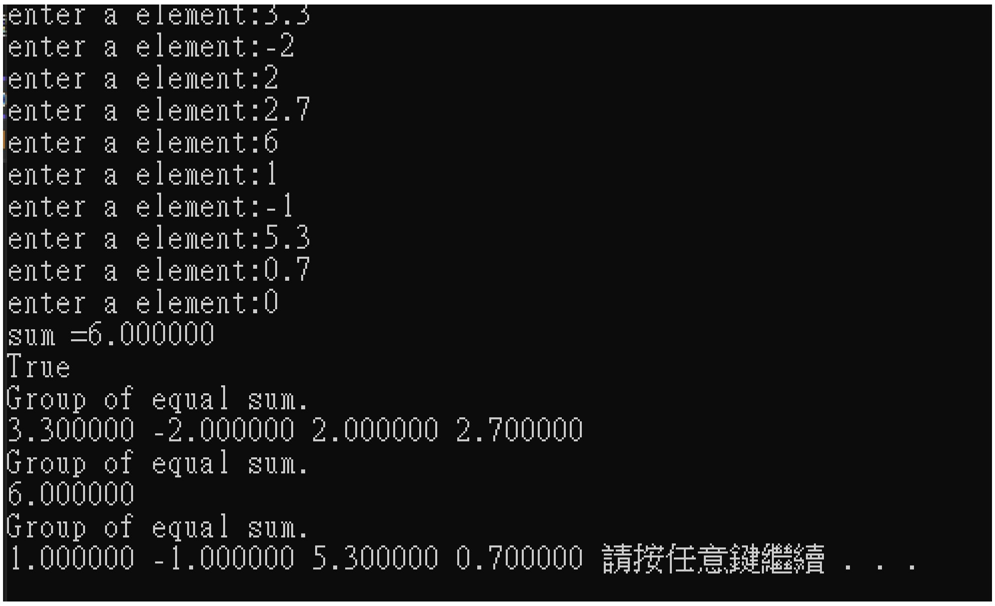
```
#include<stdio.h>
#include<stdlib.h>

int main() {
    float n[100000];
    float total = 0;      //array total
    int tonum = 0;   // how many
    float a;

    while (true) {
        printf("enter a element:");
        scanf_s("%f", &a);

        if (a == 0) break;
        n[tonum] = a;
        tonum++;
    }

    for (int i = 0; i < tonum; i++) {
        total += n[i];
    }
    float part;
    part = total / 3;
    int a1=0, a2=0, a3 = 0;
    float totala1=0, totala2=0, totala3=0;
```



```
for (int i = 0; i < tonum; i++) {
    if (part == totala1) break;
    totala1 += n[i];
    a1++;
}

for (int i = a1; i < tonum; i++) {
    if (part == totala2) break;
    a2++;
    totala2 += n[i];
}

for (int i = a2-1; i < tonum; i++) {
    if (part == totala3) break;
    a3++;
    totala3 += n[i];
}
```

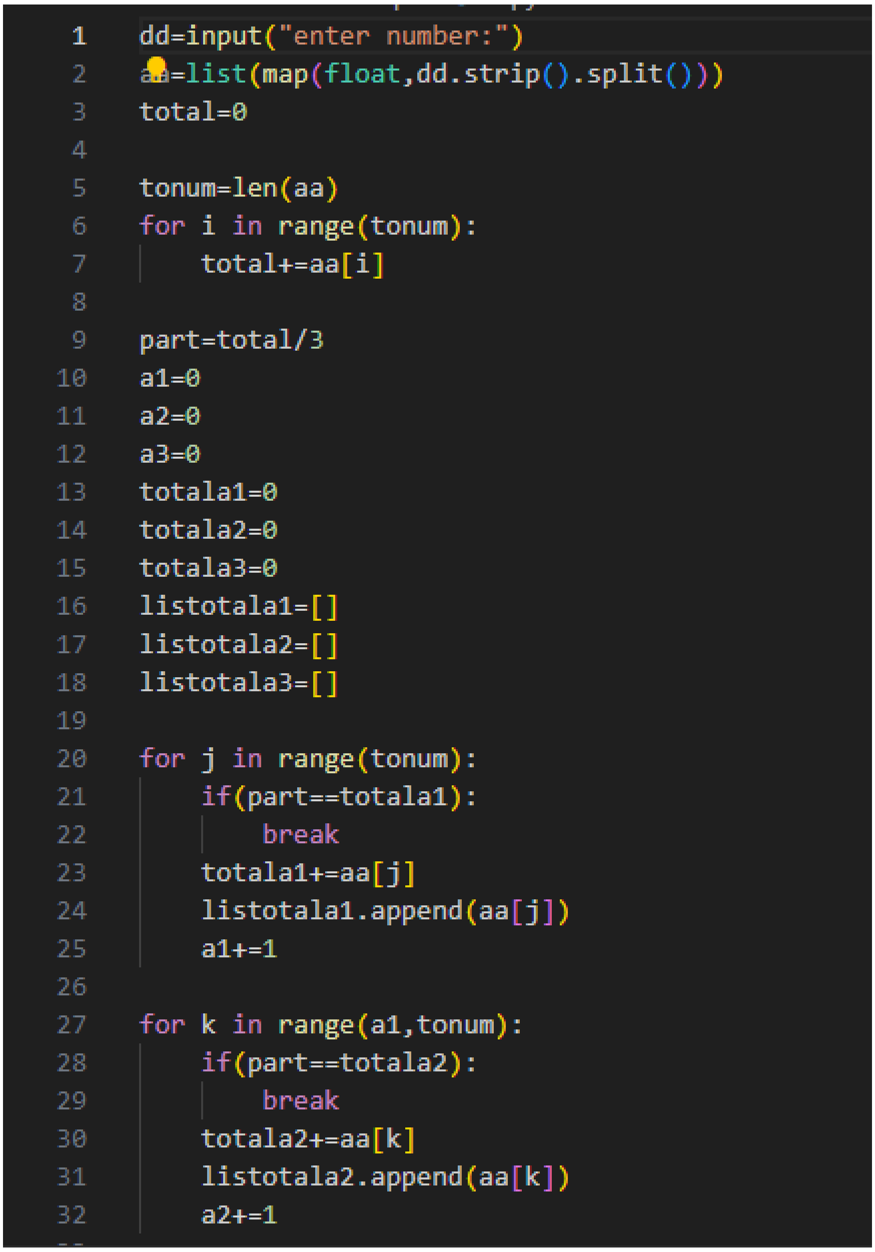


```c
    printf("sum =%f\n", part);
    if ((totala1 == totala2) && (totala2 == totala3)) {
        printf("True\n");
        printf("Group of equal sum.\n");
        printf("%f\n", part);
        for (int i = 0; i < a1; i++) {
            printf("%f ", n[i]);
        }
        printf("\n");
        printf("Group of equal sum.\n");
        for (int i = a1; i < (a1+a2); i++) {
            printf("%f ",n[i]);
        }
        printf("\n");
        printf("Group of equal sum.\n");
        for (int i = (a1 + a2); i < tonum; i++) {
            printf("%f ", n[i]);
        }

    }
    else { printf("False"); }
    system("pause");
    return 0;
}
```

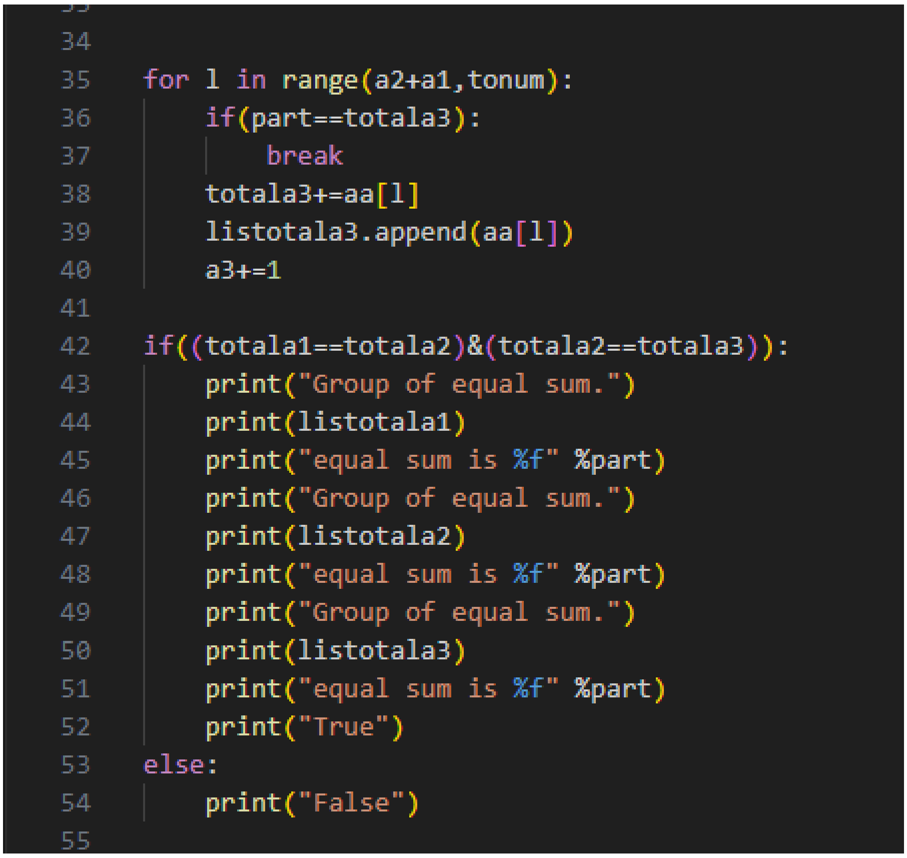


```
enter a element:3.3
enter a element:-2
enter a element:2
enter a element:2.7
enter a element:6
enter a element:1
enter a element:-1
enter a element:5.3
enter a element:0.7
enter a element:0
sum =6.000000
True
Group of equal sum.
3.300000 -2.000000 2.000000 2.700000
Group of equal sum.
6.000000
Group of equal sum.
1.000000 -1.000000 5.300000 0.700000 請按任意鍵繼續 . . .
```

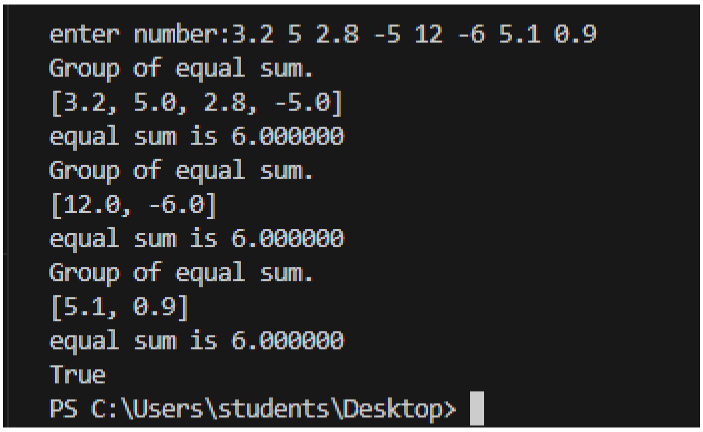
5.2.1 Python demo



```python
dd=input("enter number:")
aa=list(map(float,dd.strip().split()))
total=0

tonum=len(aa)
for i in range(tonum):
    total+=aa[i]

part=total/3
a1=0
a2=0
a3=0
totala1=0
totala2=0
totala3=0
listotala1=[]
listotala2=[]
listotala3=[]

for j in range(tonum):
    if(part==totala1):
        break
    totala1+=aa[j]
    listotala1.append(aa[j])
    a1+=1

for k in range(a1,tonum):
    if(part==totala2):
        break
    totala2+=aa[k]
    listotala2.append(aa[k])
    a2+=1
```

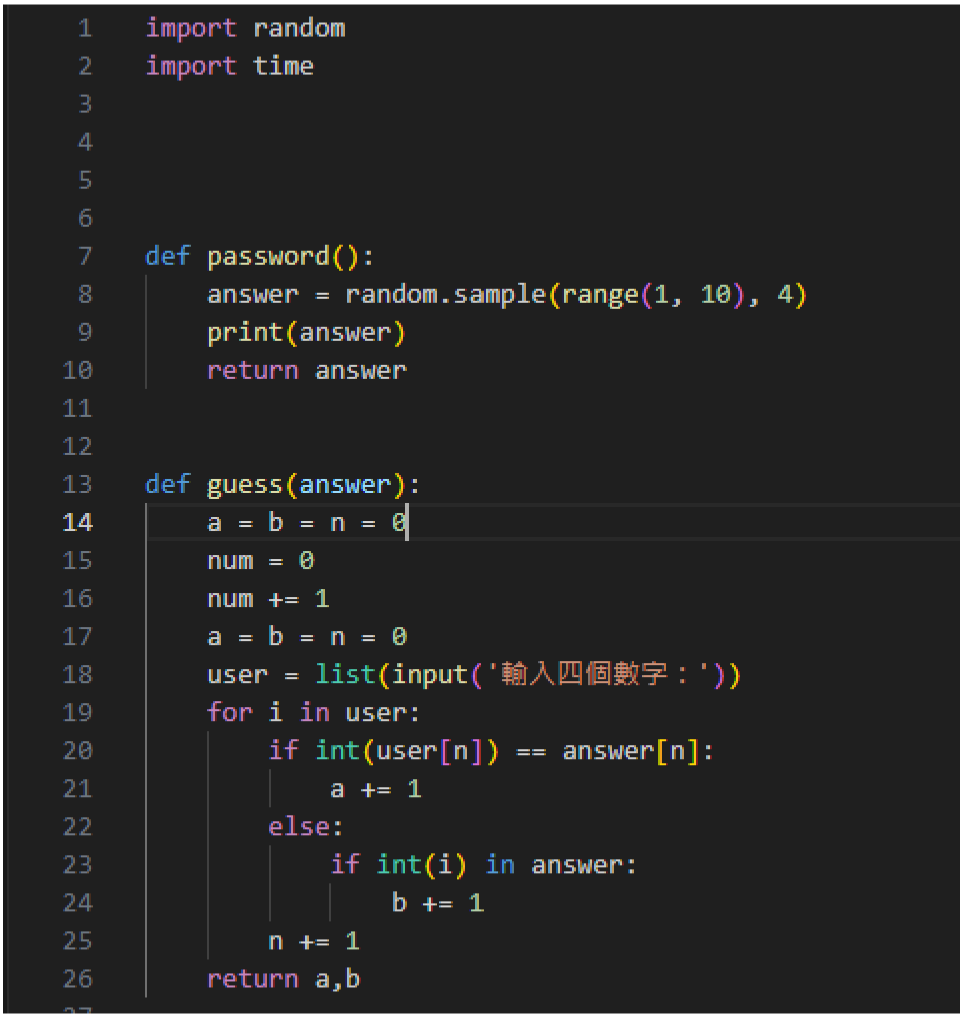


```python

for l in range(a2+a1,tonum):
    if(part==totala3):
        break
    totala3+=aa[l]
    listotala3.append(aa[l])
    a3+=1

if((totala1==totala2)&(totala2==totala3)):
    print("Group of equal sum.")
    print(listotala1)
    print("equal sum is %f" %part)
    print("Group of equal sum.")
    print(listotala2)
    print("equal sum is %f" %part)
    print("Group of equal sum.")
    print(listotala3)
    print("equal sum is %f" %part)
    print("True")
else:
    print("False")

```

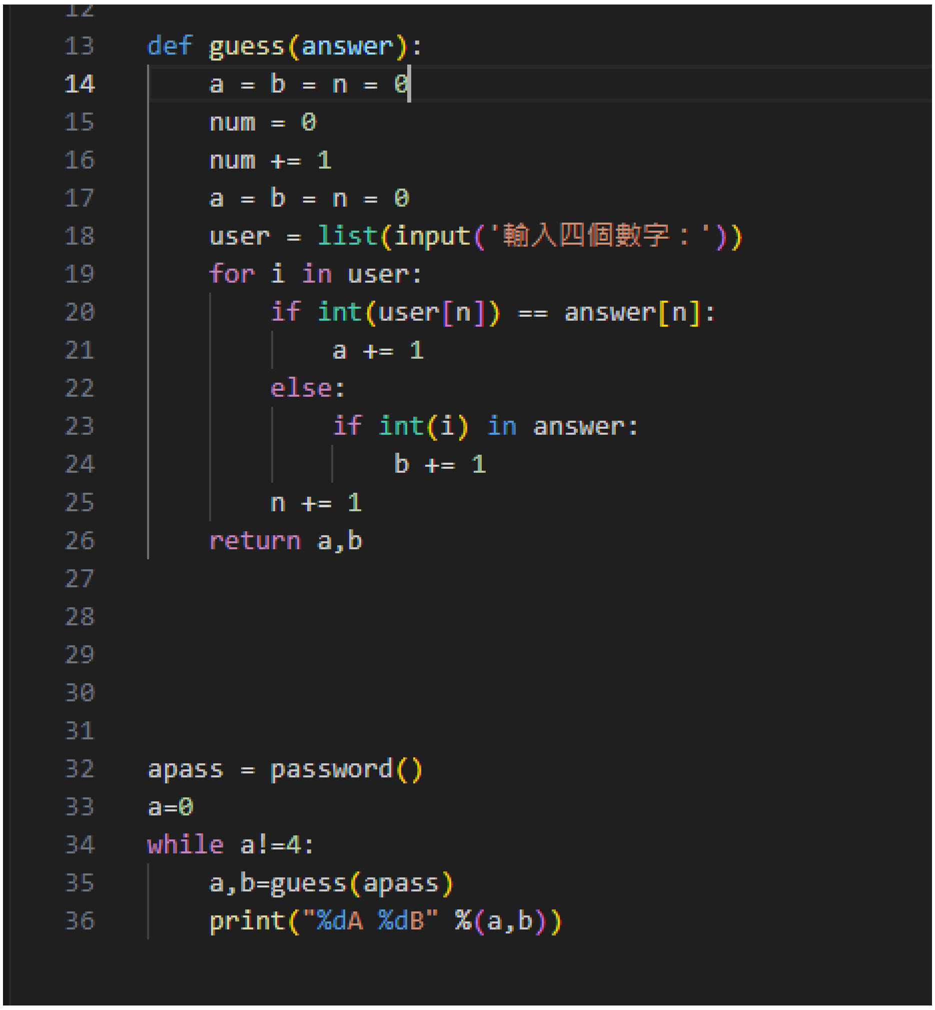


```
enter number:3.2 5 2.8 -5 12 -6 5.1 0.9
Group of equal sum.
[3.2, 5.0, 2.8, -5.0]
equal sum is 6.000000
Group of equal sum.
[12.0, -6.0]
equal sum is 6.000000
Group of equal sum.
[5.1, 0.9]
equal sum is 6.000000
True
PS C:\Users\students\Desktop>
```

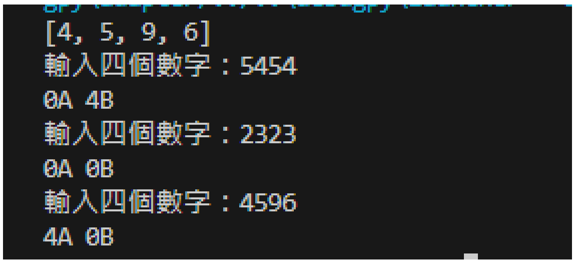
5.2.2 Python demo



```python
import random
import time




def password():
    answer = random.sample(range(1, 10), 4)
    print(answer)
    return answer


def guess(answer):
    a = b = n = 0
    num = 0
    num += 1
    a = b = n = 0
    user = list(input('輸入四個數字：'))
    for i in user:
        if int(user[n]) == answer[n]:
            a += 1
        else:
            if int(i) in answer:
                b += 1
        n += 1
    return a,b
```

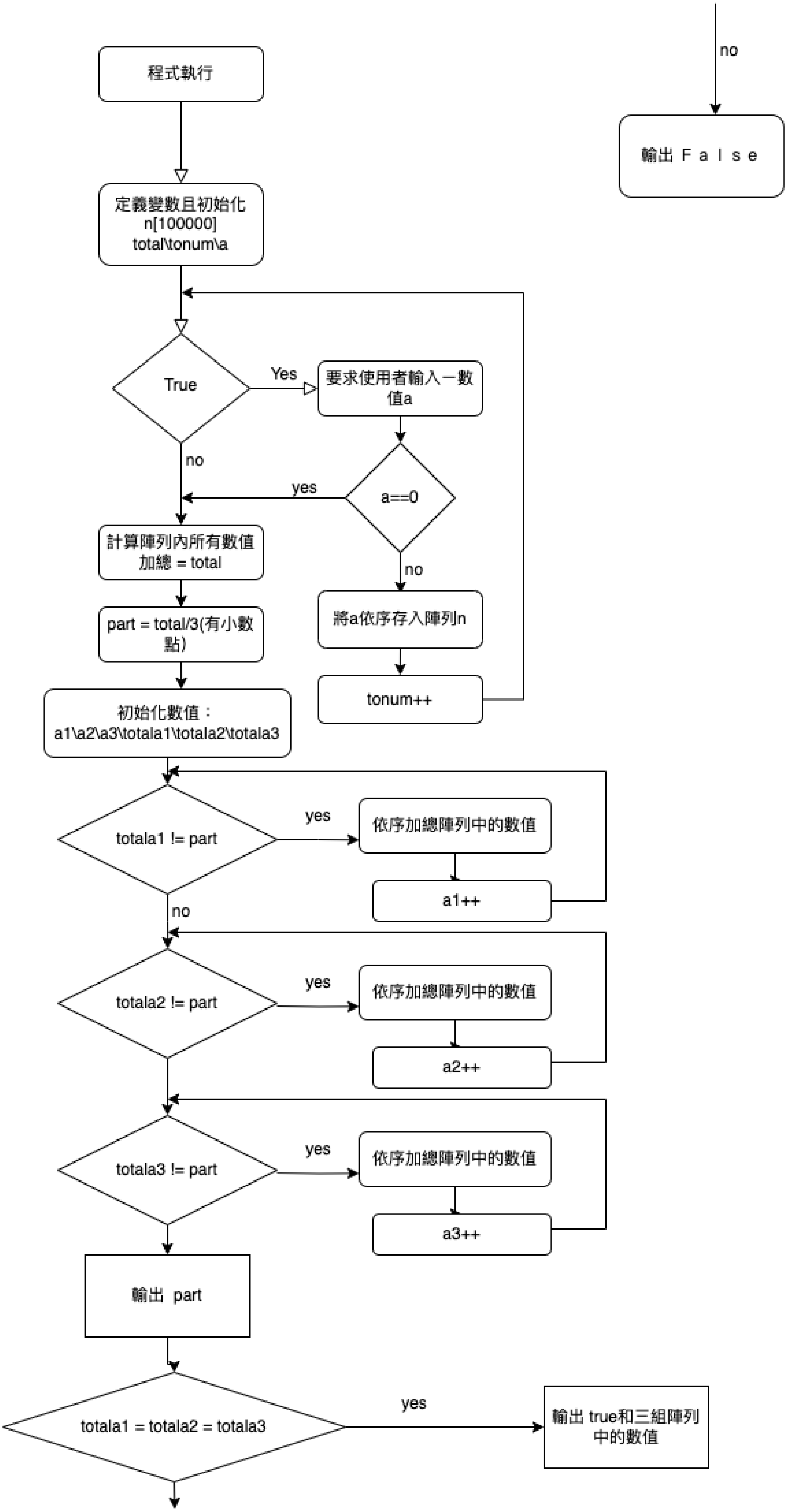


```python
def guess(answer):
    a = b = n = 0
    num = 0
    num += 1
    a = b = n = 0
    user = list(input('輸入四個數字：'))
    for i in user:
        if int(user[n]) == answer[n]:
            a += 1
        else:
            if int(i) in answer:
                b += 1
        n += 1
    return a,b




apass = password()
a=0
while a!=4:
    a,b=guess(apass)
    print("%dA %dB" %(a,b))
```



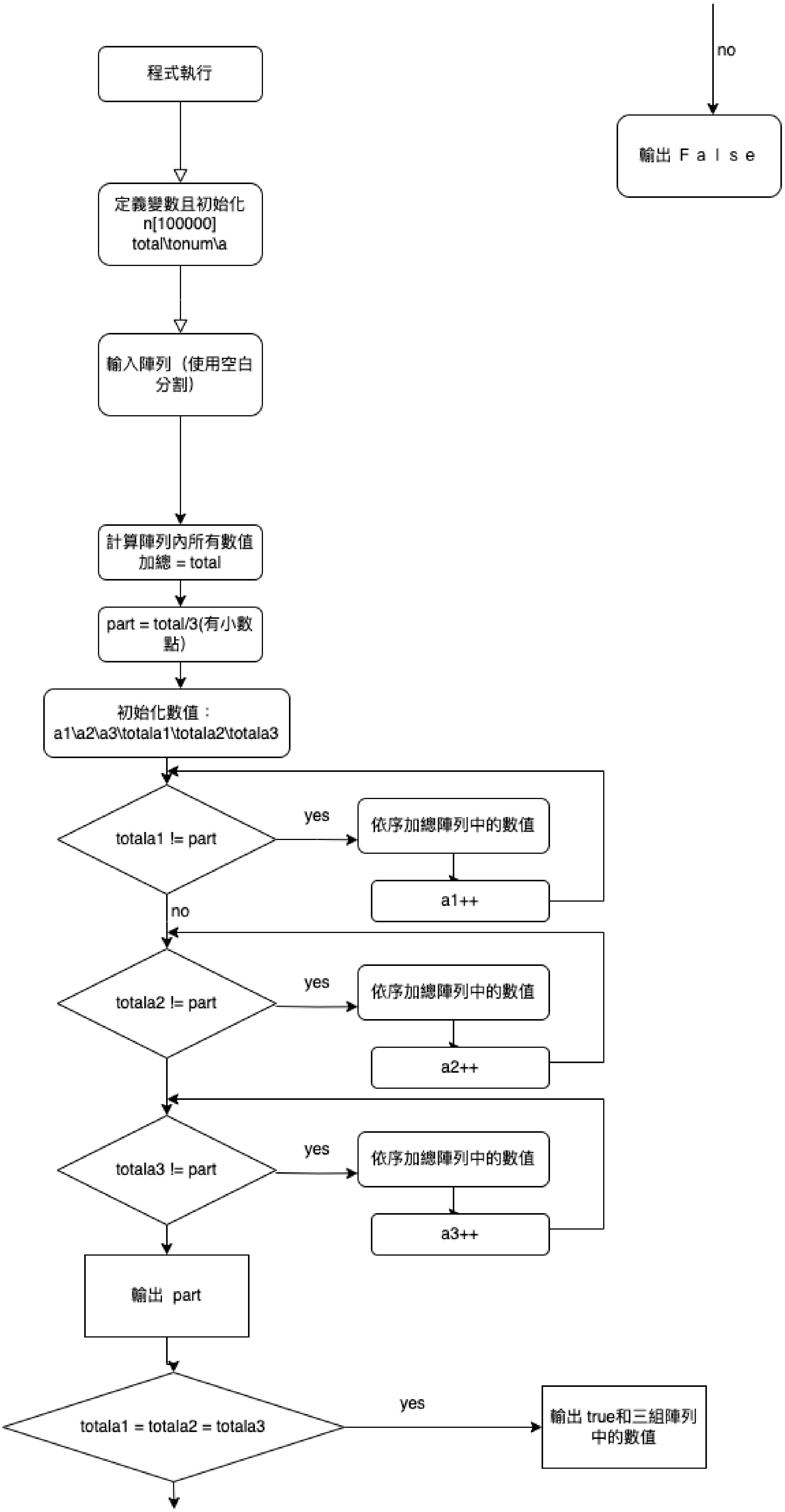
```
[4, 5, 9, 6]
輸入四個數字：5454
0A 4B
輸入四個數字：2323
0A 0B
輸入四個數字：4596
4A 0B
```

5.1 C code流程圖

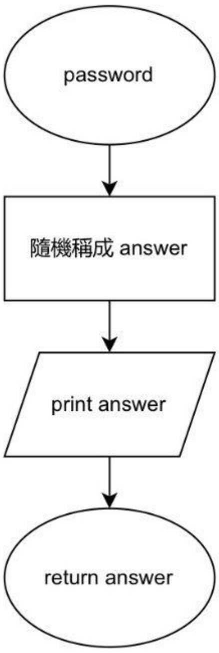


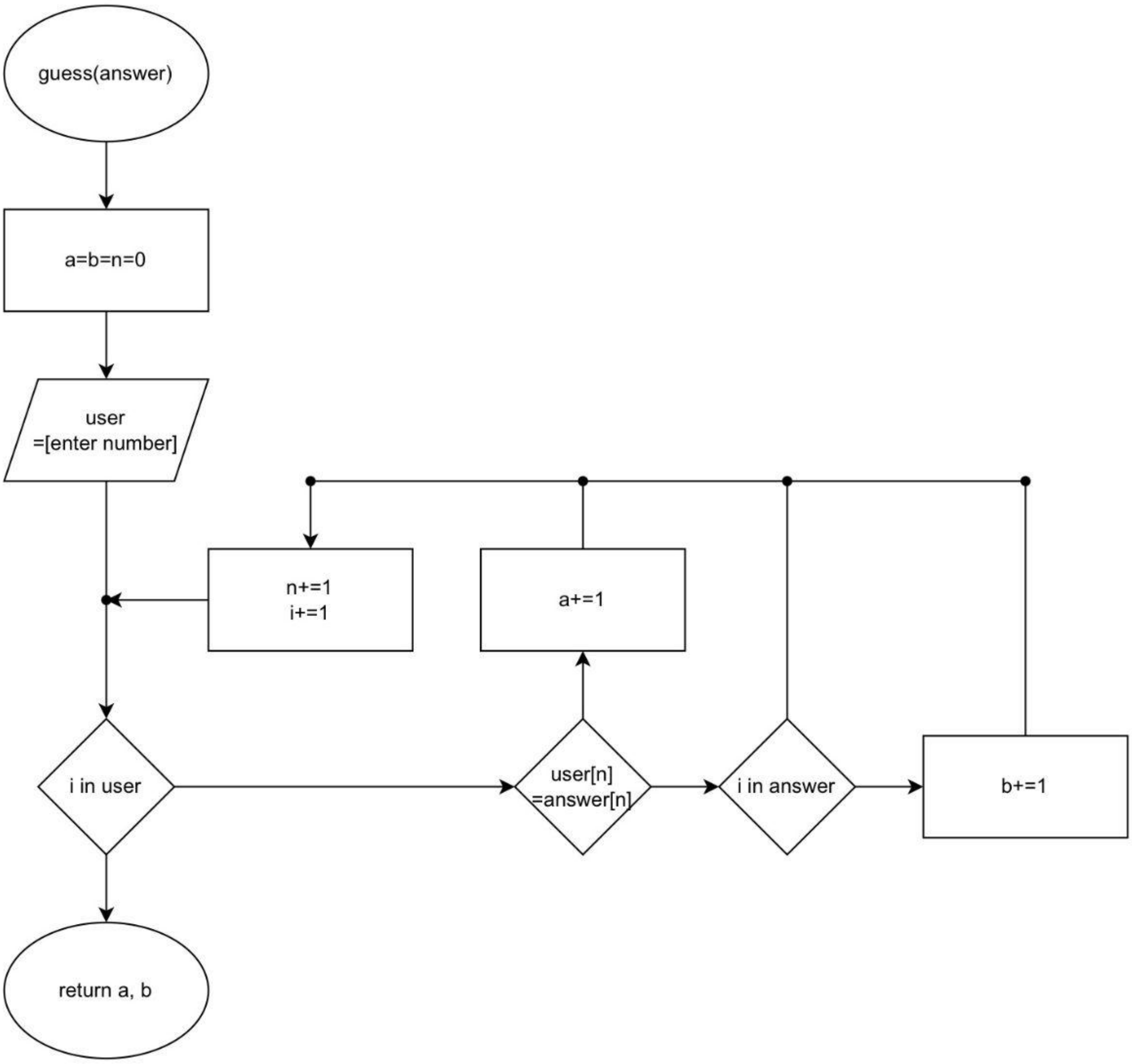
no
輸出 F a l s e
程式執行
定義變數且初始化
n[100000]
total\tonum\a
True
Yes
要求使用者輸入一數值a
no
yes
a==0
計算陣列內所有數值加總 = total
no
將a依序存入陣列n
part = total/3(有小數點)
tonum++
初始化數值：
a1\a2\a3\totala1\totala2\totala3
totala1 != part
yes
依序加總陣列中的數值
a1++
no
totala2 != part
yes
依序加總陣列中的數值
a2++
totala3 != part
yes
依序加總陣列中的數值
a3++
輸出 part
totala1 = totala2 = totala3
yes
輸出 true和三組陣列中的數值

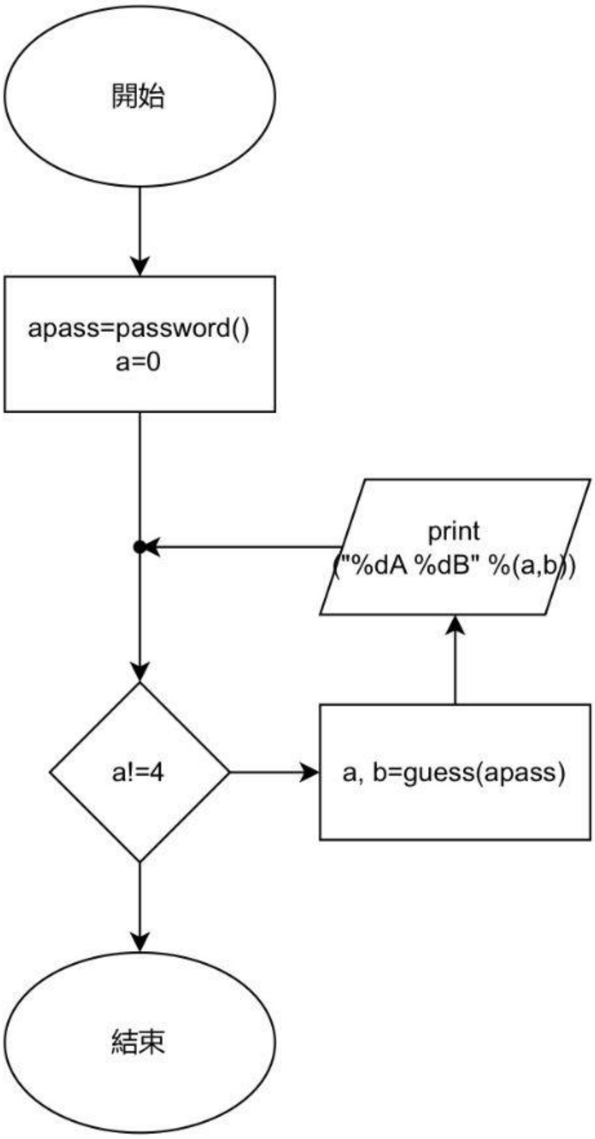
5.2.1Python code流程圖


no
輸出 F a l s e
程式執行
定義變數且初始化
n[100000]
total\tonum\a
輸入陣列（使用空白分割）
計算陣列內所有數值加總 = total
part = total/3(有小數點)
初始化數值：
a1\a2\a3\totala1\totala2\totala3
totala1 != part
yes
依序加總陣列中的數值
a1++
no
totala2 != part
yes
依序加總陣列中的數值
a2++
totala3 != part
yes
依序加總陣列中的數值
a3++
輸出 part
totala1 = totala2 = totala3
yes
輸出 true和三組陣列中的數值

5.2.2 Python code流程圖







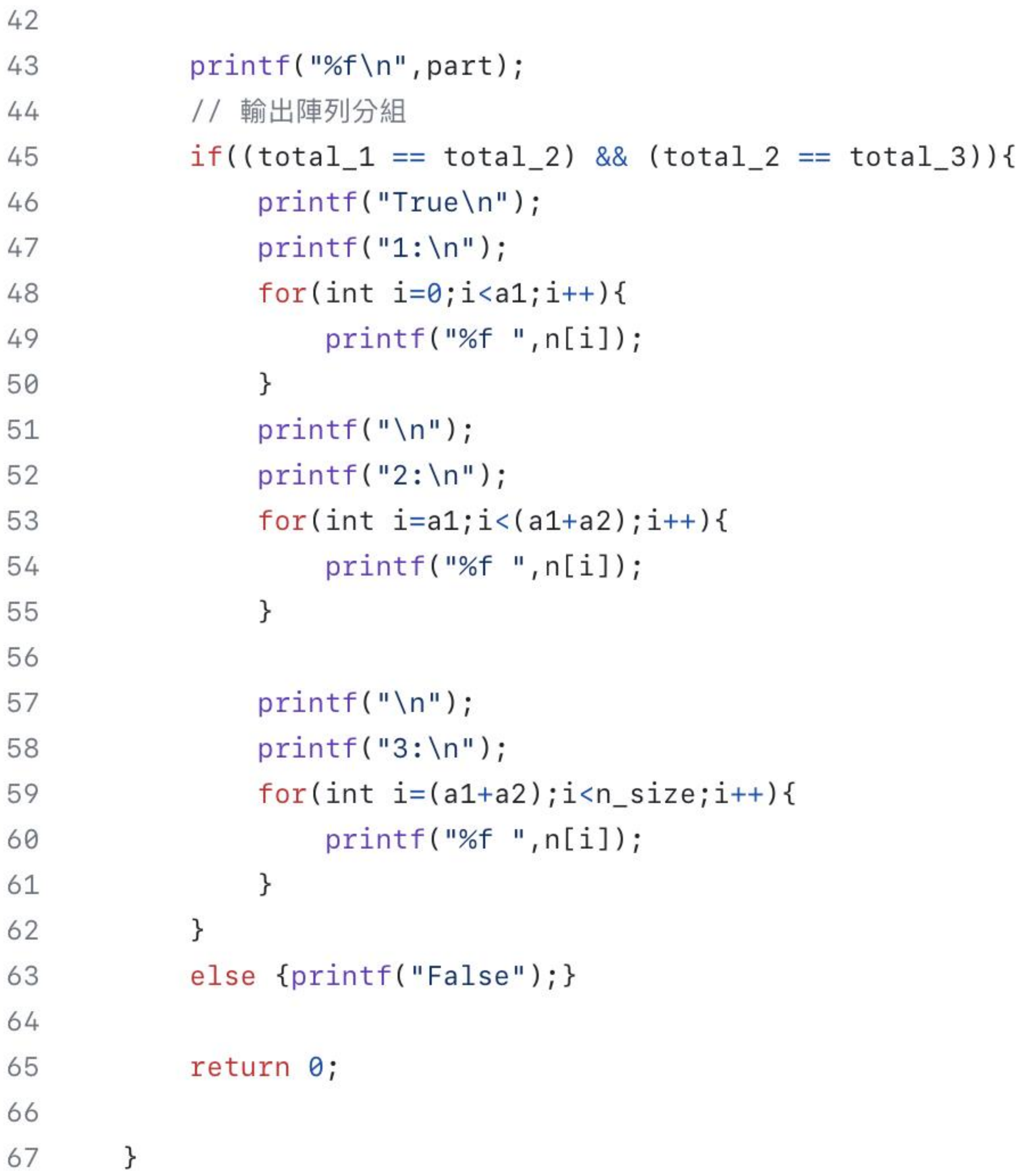
討論與分析:

5.1 C

1. 邊界情況檢查:
   萬一程式碼在畫分組別時, 沒有檢查是否所有元素都有被使用。我們應該有一個檢查來處理無法準確地將元素分成三組的情況。
2. 使用函數:
   程式碼中有許多相像的地方, 像是在尋找分組時, 就可以使用函數。提高程式碼的可維護性。
3. 邏輯判斷:
   在課堂上, 我們原本在邏輯判斷式寫 (totala1 == totala2 == totala3) 一直發生錯誤, 後來才發現原來應分開判斷, 並用且連接, 應改為 (totala1 == totala2) && (totala2 == totala3) 。

5.1 C 註解後程式:

```c
#include <stdio.h>
#include <stdlib.h>
int main(){
    float n[10000];//若使用動態陣列容易出事，所以先給陣列大小
    float total=0; int n_size = 0; //total = 陣列加總、n_size = 陣列大小
    float a;//輸入暫存到a

    //輸入陣列數值
    while (1){
        printf("enter a element:");
        scanf("%f",&a);
        if (a == 0) break; // a=0 跳出迴圈
        n[n_size] = a;
        n_size++;
    }
    //加總陣列數值
    for(int i=0 ; i<n_size ; i++){
        total += n[i];
    }

    float part;
    part = total / 3;
    int a1=0 , a2=0;//切割陣列做記號
    float total_1 = 0 , total_2 = 0 , total_3 = 0;//加總陣列分組數值

    for(int i=0 ; i<n_size ; i++){
        if(part == total_1) break;
        a1++;
        total_1 += n[i];
    }
    for(int i=a1; i<n_size ; i++){
        if(part == total_2) break;
        a2++;
        total_2 += n[i];
    }

    for(int i=0 ; i<n_size ; i++){
        if(part == total_3) break;

        total_3 += n[i];
    }
```



```
    printf("%f\n",part);
    // 輸出陣列分組
    if((total_1 == total_2) && (total_2 == total_3)){
        printf("True\n");
        printf("1:\n");
        for(int i=0;i<a1;i++){
            printf("%f ",n[i]);
        }
        printf("\n");
        printf("2:\n");
        for(int i=a1;i<(a1+a2);i++){
            printf("%f ",n[i]);
        }

        printf("\n");
        printf("3:\n");
        for(int i=(a1+a2);i<n_size;i++){
            printf("%f ",n[i]);
        }
    }
    else {printf("False");}

    return 0;

}
```

5.2.1 Python

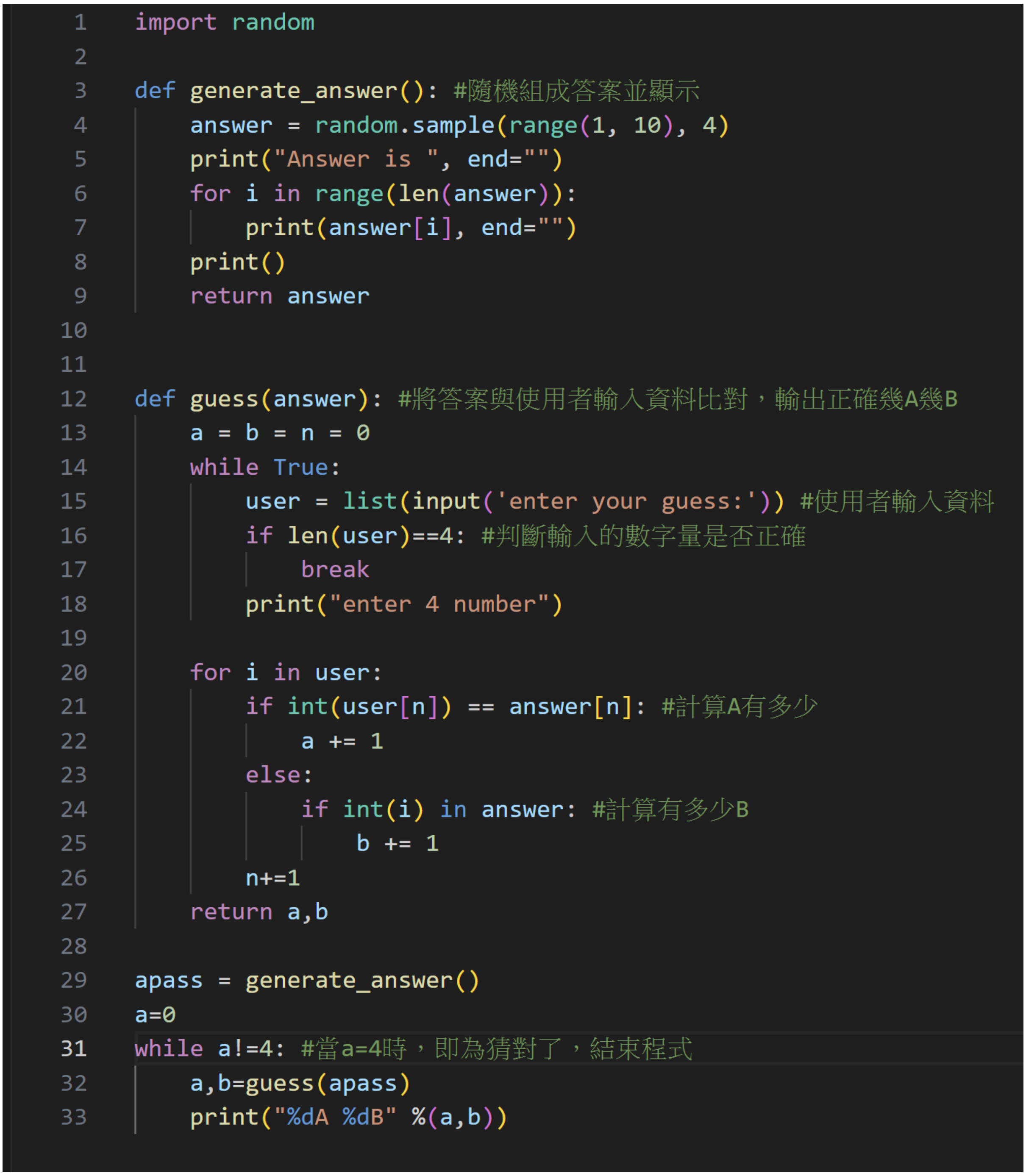
1. 程式碼簡潔：
   可以使用內建的sum函數來計算總合。
2. 語法錯誤：
   在Python中，應使用 and 並非 & 來進行運算。

5.2.2 Python

1. 程式碼簡潔：
   在函數guess中，有重複的變量初始化，可以刪除。
2. 函數命名：

在5.2.1 Python中也有相同的問題，變數名稱意義不明，在請教組員討論時，造成困難，程式可讀性極低。

5.2.2.Pyrhon註解後程式：



```python
import random

def generate_answer(): #隨機組成答案並顯示
    answer = random.sample(range(1, 10), 4)
    print("Answer is ", end="")
    for i in range(len(answer)):
        print(answer[i], end="")
    print()
    return answer


def guess(answer): #將答案與使用者輸入資料比對，輸出正確幾A幾B
    a = b = n = 0
    while True:
        user = list(input('enter your guess:')) #使用者輸入資料
        if len(user)==4: #判斷輸入的數字量是否正確
            break
        print("enter 4 number")

    for i in user:
        if int(user[n]) == answer[n]: #計算A有多少
            a += 1
        else:
            if int(i) in answer: #計算有多少B
                b += 1
        n+=1
    return a,b

apass = generate_answer()
a=0
while a!=4: #當a=4時，即為猜對了，結束程式
    a,b=guess(apass)
    print("%dA %dB" %(a,b))
```

總結而言，這次的lab我們的時間非常不夠，應是平時缺乏練習導致題目一變複雜我們就需要花很多時間來完成，最後寫出來的程式也非常雜亂，下課後我們潛心研究改進，期許自己下次更好。