

計算機程式 I 作業

~~~~~

### 一、程式作業繳交注意事項

1. 作業遲交不計分。
2. 程式碼有抄襲者，任何一次抄襲，學期成績不及格。

### 二、程式作業撰寫規則，違反規則該次作業不予計分

1. 程式碼不得使用全域變數，亦即，宣告在 main 外面的變數！
2. 程式碼不得使用 goto 指令。
3. 程式碼不得使用 system("pause") 功能。
4. 程式碼不得使用 非 C 語言的功能，如 C++ for (int i=0; i<x, i++) {}。
5. 作業上傳請繳交.c 檔，不得繳交 .cdp .zip .rar .7z .exe 等不合法檔案！

### 三、課程注意事項

1. 跟助教、卓越小老師預約課後輔導時間，請準時，學校會派人來簽到。
2. 作業要跟助教討論除錯問題，請跟助教約時間，當面討論比較有效果。
3. 作業等相關問題可寄至助教信箱詢問。
4. 下列網站提供很好的查詢資源請同學使用。

<http://www.cplusplus.com/reference/clibrary/>

=====

## Week 1

### 001. 計算總成績、平均

某一學生修國文、計算機概論、計算機程式設計三科，使用者輸入名字（一個 char）、學號（int）、三科成績(int)。

- (1) 計算學生總成績、平均。
- (2) 印出名字、學號、總成績、平均。

Input

K

905067

100

100

100

Output

Name:K

ID:905067

Average:100

Total:300

-----

Week 1

002. 一元二次方程式

一元二次方程式， $aX^2 + bx + c = 0$ ，輸入 a, b, c，求 方程式的兩個實根。

-----

輸入說明

第一個數(int) a

第二個數(int) b

第三個數(int) c

-----

輸出說明

第一個實根(double)， $x1 = ((-b) + \sqrt{b*b - 4*a*c}) / (2*a)$

第二個實根(double)， $x2 = ((-b) - \sqrt{b*b - 4*a*c}) / (2*a)$

須 #include < math.h >

x1, x2 輸出到小數點第一位

printf("%.1f", x1);

-----

Input

1

-2

1

Output

1.0

1.0

-----

Week 1

### 003. 數值運算

分別輸入(int) num1 num2 求出兩數的 Sum,Difference,Product,Quotient。

Difference 結果取絕對值，使用 double fabs(double);

須 #include < math.h >

結果須輸出到小數點第二位(double)

```
printf("%.2f",x1);
```

輸入：

25

2

輸出：

Difference:23.00

Sum:27.00

Quotient:12.50

Product:50.00

-----

Week 1

### 004. 判斷基數偶數

輸入說明：

輸入一個整數

輸出說明：

輸出奇數或偶數

輸入範例：

3

輸出範例：

odd

輸入範例：

6

輸出範例：

even

-----

Week 1

005. BMI 計算

輸入身高(公尺) 體重(公斤)(皆使用 double)

$BMI = \text{體重(公斤)} / (\text{身高} * \text{身高})(\text{公尺})$

結果須輸出到小數點第一位(double)

```
printf("%.1f",x1);
```

輸入：

1.72

60

輸出：

20.3

-----

Week 2

006. 判斷何種三角形

當三個邊長能構成三角形時，再判斷該三角形是否為正三角形，若否，則判斷是否為等腰三

角形：

1. 不能成為三角形：任兩邊和不大於第三邊，或任一邊長不大於 0。
2. 正三角形：三個邊相等。
3. 等腰三角形：任兩邊相等，平方和大於第三邊的平方。
4. 一般三角形：非正三角形與等腰三角形。

此題必須寫一個運算的 function

```
int getTriangle(int a, int b, int c);
```

輸入說明：

-----

輸入三個整數邊長

輸出說明：

-----

1. 不能成為三角形：輸出 not triangle。
2. 正三角形：輸出 equilateral triangle。
3. 等腰三角形：輸出 isosceles triangle。
4. 一般三角形：輸出 triangle。

測試案例(Test Case)資料：

Input：

4 1 1

Output：

not triangle

-----

Input：

3 3 3

Output：

equilateral triangle

-----

Input：

3 2 3

Output :  
isosceles triangle

-----

Input :  
7 8 9

Output :  
triangle

-----

Week 2

007

請輸入五個數字，分別計算出平均數、變異數、標準差，  
並精確到小數點後第二位(註，之後的小數捨去)。

變異數參考公式：  
 $\Sigma (X_i - \mu)^2 / N$

標準差參考公式：  
 $(\Sigma (X_i - \mu)^2 / N)^{0.5}$

平均數參考公式：  
 $\mu = \Sigma (X_i) / N$

例如：1 2 8 9 10

變異數：14.00

>  $\Sigma (X_i - \mu)^2 = (1-6)^2 + (2-6)^2 + (8-6)^2 + (9-6)^2 + (10-6)^2$   
= 25 + 16 + 4 + 9 + 16 = 70  
700. / 5.0 = 14.00

標準差：3.74

>  $14^{0.5} = 3.74165$   
取兩位小數 = 3.74

平均值：6.00

> (1+2+8+9+10)/5.0 = 6.00

輸入說明：

-----

輸入五個整數

輸出說明：

-----

變異數

標準差

平均值

輸出到小數點第二位 `printf("%.2f",x);`

測試案例(Test Case)資料：

Input：

1 2 8 9 10

Output：

14.00

3.74

6.00

-----

Week 2

008. 一元二次方程式

一元二次方程式， $aX^2 + bx + c = 0$ ，輸入 a, b, c，求 方程式的兩個實/虛根。

$T = \sqrt{b*b-4*a*c}$

第一個根，  $x1 = (-b+T)/(2*a)$

第二個根，  $x2 = (-b-T)/(2*a)$

本題須 `#include < math.h >`

-----

輸入說明：

第一個數 a

第二個數 b

第三個數 c

-----  
輸出說明：

當  $T \geq 0$

輸出 x1, x2 為實根，輸出到小數點第一位 `printf("%.1f",x1);`

當  $T < 0$

輸出 x1 , x2 為虛根，輸出到小數點第一位

`printf("%.1f+%.1fi",x11, x12);`

`printf("%.1f-%.1fi",x21, x22);`

若 x11 或 x21 為 0.0 時，不須輸出正負號

測試案例(Test Case)資料：

Input：

1

-2

1

Output：

1.0

1.0

-----  
Input：

1

2

2

Output：

-1.0+1.0i

-1.0-1.0i

-----  
Week 2



A、B、C 三本書價格及折扣表如下：

|   | 定價   | 1~10 本 | 11~20 本 | 21~30 本 | 31 本以上 |
|---|------|--------|---------|---------|--------|
| A | 380  | 原價     | 打 9 折   | 打 8.5 折 | 打 8 折  |
| B | 1200 | 原價     | 打 9.5 折 | 打 8.5 折 | 打 8 折  |
| C | 180  | 原價     | 打 8.5 折 | 打 8 折   | 打 7 折  |

有一顧客欲購買 A：x 本、B：y 本、C：z 本（x、y、z 為使用者輸入），請計算需花費多少錢？

例如：

A 購買 6 本 B 購買 12 本 C 購買 30 本

$6*380+12*1200*0.95+30*180*0.8$

$=20280$

輸入說明：

-----  
 A 書本數量，整數  
 B 書本數量，整數  
 C 書本數量，整數

輸出說明：

-----  
 費用，整數  
`printf("%d", x);`

測試案例(Test Case)資料：

Input：

6

12

30

Output：

20280

-----  
 Week 2

010. 最佳資費選擇

輸入每月網內、網外、市話、通話時間(秒)及網內、網外簡訊則數，求最佳資費。

費率如下表：

| 資費類型 | 183 型               | 383 型 | 983 型 |
|------|---------------------|-------|-------|
| 月租費  | 183 元               | 383 元 | 983 元 |
| 優惠內容 | 月 租 費 可 抵 等 額 通 信 費 |       |       |

|          |       |       |       |
|----------|-------|-------|-------|
| 語音費率 網內  | 0.08  | 0.07  | 0.06  |
| (元/秒) 網外 | 0.139 | 0.130 | 0.108 |
| 市話       | 0.135 | 0.121 | 0.101 |
| (元/秒)    |       |       |       |

|          |       |       |       |
|----------|-------|-------|-------|
| 簡訊費率 網內  | 1.128 | 1.128 | 1.128 |
| (元/則) 網外 | 1.483 | 1.483 | 1.483 |

-----  
輸入說明：

-----  
網內語音(秒)，整數  
網外語音(秒)，整數  
市話 (秒)，整數  
網內簡訊數，整數  
網外簡訊數，整數

輸入說明：

-----  
費用，整數  
最佳資費類型，(183, 383, 983)

測試案例(Test Case)資料：

Input：

120

150

20

10

5

Output :

183

183

-----

Input :

3000

4000

200

5

5

Output :

767

383

-----

Week 3

011.

請檢查輸入的三門課程是否衝堂。

依序輸入

課程編號 (4 位數字)、

上課小時數 (1-2 小時)、

上課時間 (依小時數輸入上課時間, 星期 1-5, 第 1-8 節)。

輸入任何錯誤, 輸出-1。

輸入說明:

-----

1002 (第一門課課程編號)

2 (2 小時)

33 (星期 3 第 3 節課)

59 (星期 5 第 9 節課)

1003 (第二門課課程編號)

2 (2 小時)

11 (星期 1 第 1 節課)

33 (星期 3 第 3 節課)

3003 (第三門課課程編號)

2 (2 小時)

11 (星期 1 第 1 節課)

33 (星期 3 第 3 節課)

(上課時間輸入順序為星期 1~5，第 1 節 ~ 第 8 節)

(課程編號輸入順序為編號大小)

輸出說明：

-----

1002,1003,33

1002,3003,33

1003,3003,11

1003,3003,33

輸出 課程編號,課程編號,衝突時間

若沒有衝突則輸出 correct

若有錯誤輸入則輸出 -1

(每次列出兩個衝突課程編號，一個衝突時間，所有倆倆課程衝突均要列出)

(衝突輸出順序，課程 1-2, 1-3, 2-3, 第 1 節 ~ 第 2 節)

測試案例(Test Case)資料：

Input：

1001

2

12

13

1002

2

13

21

1003

2  
21  
25

Output :

1001,1002,13  
1002,1003,21

-----

Input :

1001  
1  
21  
1002  
2  
32  
33  
1003  
2  
45  
46

Output :

correct

-----

Input :

101  
3  
99  
1002  
2  
32  
33  
1003  
2  
45  
46

Output :

-1

-----

Week 3

012.

請算出 a,b,c 三條線在 X 軸上所涵蓋的長度(不含重複線段)

例如:a(x1,x2)表示 a 線段為 X 軸上點 x1 到點 x2 的線

輸入說明:

-----

-1 (a 的 x1 座標為 -1)

1 (a 的 x2 座標為 1)

0 (b 的 x1 座標為 0)

2 (b 的 x2 座標為 2)

1 (c 的 x1 座標為 1)

3 (c 的 x2 座標為 3)



輸出說明:

-----

4

-----

Week 3

013.

小明到保齡球館打保齡球，一輪有十局，假設小明一到八局都拿零分，剩下最後兩局

每局有十瓶保齡球瓶，倒一瓶保齡瓶得一分，每一局最多為十分，

每一局可以打兩次，若在第10局打出 strike，可以再多打一局，但當局只能打一次，試算  
出總得分

測試案例(Test Case)資料：

Input：

2

5  
7  
1

Output :

15

-----

Input :

5  
5  
10  
0  
8

Output :

28

=====

Week 4

#### 014. 停車費計算

假設某個停車場的費率是停車 2 小時以內，每半小時 30 元，未滿半小時部分不計費。  
超過 2 小時，但未滿 4 小時，每半小時 40 元，未滿半小時以半小時計費。  
超過 4 小時以上的部份，每半小時 60 元，未滿半小時以半小時計費。

請撰寫程式計算輸入數筆資料，共需繳交的停車費。  
本程式不考慮隔夜情況。

輸入說明：

-----

輸入 3 組時間，每組分別為開始與離開時間，24 小時制。  
若輸入格式錯誤，則輸出 error

輸出說明：

-----

輸出總停車費。

測試案例(Test Case)資料：

Input：

8:00

9:15

13:45

16:50

6:20

10:50

Output：

60

240

340

Input：

00:01

25:00

00:11

23:66

-01:00

12:34

Output：

error

error

error

-----

Week 4

## 015. 羅馬數字轉換

本題禁止使用迴圈。

羅馬數字共有 7 個，即 I（1）、V（5）、X（10）、L（50）、C（100）、D（500）和M（1000）。



按照下述的規則可以表示任意正整數。

需要注意的是羅馬數字中沒有「0」，與進位制無關。一般認為羅馬數字只用來記數，而不作演算。

重複數次：

一個羅馬數字重複幾次，就表示這個數的幾倍。

右加左減：

在較大的羅馬數字的右邊記上較小的羅馬數字，表示大數字加小數字。

在較大的羅馬數字的左邊記上較小的羅馬數字，表示大數字減小數字。

左減的數字有限制，僅限於 I、X、C。比如 45 不可以寫成 VL，只能是 XLV

但是，左減時不可跨越一個位值。比如，99 不可以用 IC (100-1) 表示，而是用 XCIX ([100-10]+[10-1]) 表示。(等同於阿拉伯數字每位數字分別表示。)

左減數字必須為一位，比如 8 寫成 VIII，而非 IIX。

右加數字不可連續超過三位，比如 14 寫成 XIV，而非 XIIII。(見下方「數碼限制」一項。)

數碼限制：

同一數碼最多只能連續出現三次，如 40 不可表示為 XXXX，而要表示為 XL。

Hint：請由下表中找出規律，並用判斷式實作此題，若使用迴圈則不計分  
禁止使用暴力破解

輸入說明：

-----

輸入為一個正整數，並且小於 100

輸出說明：

-----

根據輸入的整數，輸出對應的羅馬數字

測試案例(Test Case)資料：

Input：

3

Output：

III

Input：

40

Output：

XL

-----  
Week 5

## 016.計算電費

輸入月份、輸入前年所使用的度數及今年使用的度數，  
依據月份是否為夏季電費做計算依據，並計算出電費，  
若今年度有較前年成功節電，則每節一度電可省 0.6 元

| 每度(元)     | 夏季電價(7~9 月) | 非夏季電價 |
|-----------|-------------|-------|
| 120 度以下   | 2.10        | 2.10  |
| 121-330 度 | 3.02        | 2.68  |
| 331-500 度 | 4.39        | 3.61  |
| 501-700 度 | 4.97        | 4.01  |
| 701 度以上   | 5.63        | 4.50  |

輸入說明：

-----  
輸入月份  
輸入前年使用度數  
輸入今年使用度數

輸出說明：

-----  
測試案例(Test Case)資料：

Input：

7

505

525

Output：

2609.25

Input：

7

550

525

Output

2594.25

(以此題為例，較上年度節電 25 度，故計算方式為  $525 \times 4.97 - (550 - 525) \times 0.6$ )

-----  
week6 (不可以使用陣列)

017.

求階層(N!)

輸入說明:

輸入一個整數 N

-----  
輸出說明:

求出小於 N 的所有質因數的階層，顯示每個質因數階層值和階層值總和

排除輸入錯誤(小數點、負數、字元)

不合法的輸入則輸出 E

input :

10

output:

2 120

122

-----  
input

10.2

output:

E

-----  
(不可以使用陣列)

018.

小明的家庭作業裏有很多數列填空練習。

填空練習的要求是：已知數列的前四項，填出第五項。

因為已經知道這些數列只可能是等差或等比數列，他決定寫一個程式來完成這些練

輸入說明:

第一行是數列的數目  $t$  ( $1 \leq t \leq 20$ )。

以下每行均包含四個整數，表示數列的前四項。

輸出說明：

對輸入的每個數列，輸出它的前五項。

不合法的輸入則輸出 E。

input :

2

1 2 3 4

32 16 8 4

output:

1 2 3 4 5

32 16 8 4 2

-----

(不可以使用陣列)

019.

請使用一個 while loop 和 function call

給予一個十進位整數，請撰寫一程式可以將此十進位整數轉換為指定的進制的整數。

輸入說明：

輸入分為兩部份，包括指定的進制數

(2 ~ 16)

與十進位整數(0 ~ 1000000000)

輸出說明：

經轉換後的新進位制的整數( y )

不合法的輸入則輸出 E

input:

16 1234

output:

4D2

-----

Input:

8 56456456

Output :

327272410

-----

Input :

11 17489465

Output :

9966104

-----

Input :

4 17489

Output :

10101101

-----

(不可以使用陣列)

020.

請使用一個 while loop 和 function call

給定一些線段，求這些線段所覆蓋的長度，注意，重疊的部分只能算一次。

輸入說明：

第一列有一個正整數 n:

代表共有 n 組測試案例。

接下來每一組測試案例的第一列是一個整數 m

表示此測試案例有 m 個線段，

接著的 m 列每一列是一個線段的兩端點，

每一個端點是一個整數介於 0~60000 之間，

端點之間以一個空格區隔，線段個數不超過 5000。

起始端點小的先輸入。

此題不使用陣列。

-----

輸出說明：

針對每一組測試案例，輸出其覆蓋的長度，每組測試案例輸出一列。  
不合法的輸入則輸出 E

Input :

2 (共有 2 組次是案例)  
2 (此組測試案例有 2 個線段)  
0 1  
2 3  
3 (此組測試案例有 3 個線段)  
0 20  
10 30  
40 50

Output :

2  
40  
-----

Input :

3  
3  
10 111  
150 3450  
160 180  
2  
100 180  
150 3333  
1  
150 150

Output :

3401  
3233  
0  
-----

Input :

1  
2  
150 320  
160 190

Output:

170

-----

(不可以使用陣列)

021.

請使用 while loop 或 for loop

第一個輸入意義為選擇三種圖形:

- 1 三角形方尖方面向右邊
- 2 三角形方尖方面向左邊
- 3 菱形

第二個輸入意義為畫幾行

(奇數，範圍為 3,5,7,9,...,21)

input

1 (第一種圖形，三角形尖方面向右邊)

9 (共 9 行)

-----

output

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

-----

input

2 (第二種圖形，三角形尖方面向左邊)

5 (共 5 行)

-----

output

..\*

. \*\*

\*\*\*

```
.**  
..*
```

-----  
input

3 (第三種圖形：菱形 )

3 (共 3 行數)

輸出

```
.*  
***  
.*
```

-----  
(不可以使用陣列)

022.

輸入第一個整數，決定輸出圖形種類，

輸入第二個整數，決定輸出的行數。

```
..1  
.121  
12321  
.121  
..1
```

```
7531357
```

```
.53135
```

```
..313
```

```
...1
```

輸入說明：

每一組兩個數字，輸入-1 結束所有輸入。

第一個正整數，1 為菱形數字，2 為三角形數字，其餘輸入均為不合法。

第二個正整數為行數 N， 第一種圖形合法輸入 N 為奇數， $1 \leq N \leq 18$ ，

第二種圖形合法輸入， $1 \leq N \leq 5$ ，其餘輸入均為不合法。

不合法輸入，則輸出 E。

輸出說明：

每個圖形之間需留空行

請參考範例輸出。



Sample Input1:

1 9

1 13

-1

Sample Output:

....1

...121

..12321

.1234321

123454321

.1234321

..12321

...121

....1

.....1

.....121

....12321

...1234321

..123454321

.12345654321

1234567654321

.12345654321

..123454321

...1234321

....12321

.....121

.....1

Sample Input2:

1 22

1 5

-1

Sample Output:

E

..1

.121

12321  
.121  
..1

Sample Input3:

2 5  
2 4  
2 2  
2 12  
-1

Sample Output:

975313579  
.7531357  
..53135  
...313  
....1

7531357  
.53135  
..313  
...1

313  
.1

E

-----  
(不可以使用陣列)

023.

我們常用數字密碼鎖來保護重要的東西，但要是不小心忘了密碼麻煩就大了！  
以四位數字的密碼鎖為例，我們最多要嘗試  $10^4=10000$  次才能解鎖。這時候要是  
有辦法知道目前嘗試的密碼錯了幾個字，那解鎖的速度就快多了。請寫一個程式，  
可以判斷每組數字跟正確答案差了幾個字。

輸入說明：  
多筆輸入。

第一行有四個介於 0-9 之間的數字，代表正確的密碼

第二行以後，每行有四個介於 0-9 之間的數字，每行各代表一組嘗試的密碼。

輸入 -1 則結束輸入

輸出說明

對於每組嘗試的密碼，若有 p 個數字的值正確，且在正確的位子上，

另外有 q 個數字的值正確，但不在正確的位子上，

輸出 pAqB。

範例輸入

1 2 3 4

1 1 4 5

1 2 4 3

1 1 4 4

4 3 2 1

-1

範例輸出

1A1B

2A2B

2A0B

0A4B

範例輸入

1 1 1 5

1 1 1 1

0 9 2 8

1 5 2 3

1 1 5 1

-1

範例輸出

3A0B

0A0B

1A1B

2A2B

提醒：

如果有出現重複的數字，則重複的數字每個也只能算一次，且以最優的結果為準，

如正確答案為 5543，猜的人猜 5255，則在這裡不能認為猜測的第一個 5 對正確答案第二

個，根據最優結果為準的原理和每個數字只能有一次的規則，兩個比較後應該為 1A1B，第一個 5 位子正確，記為 1A；猜測數字中的第三個 5 或第四個 5 和答案的第二個 5 匹配，只能記為 1B。當然，如果有猜 5267 中的第一個 5 不能與答案中的第二個 5 匹配，因此只能記作 1A0B。

-----  
(不可以使用陣列)

024.

回文結構數字

輸入兩整數，找出兩整數區間內所有具有回文結構的數字。

例如，輸入 80 100，則輸出為 88 99。

例如，100 200，則輸出為 101 111 121 131 141 151 161 171 181 191。若沒有則輸出 0。

輸入說明：

輸入兩整數。

輸出說明：

輸出區間內所有具有回文結構的數字。

input:

1 10

output:

0

-----  
input:

10 70

output:

11 22 33 44 55 66

-----  
input:

1300 2000

output:

1331 1441 1551 1661 1771 1881 1991

-----  
input:

17000 18000

output:

17071 17171 17271 17371 17471 17571 17671 17771 17871 17971

-----

025.

一個有限狀態機(Finite state machine, FSM) 從起始狀態(start)開始運作，包含一組橢圓形表示的狀態(State)和一組箭頭表示的轉換(Transition)連接兩個狀態，且接受輸入字元資料。

請參考圖片作答，來源網址：<https://imgur.com/a/HOUya>

寫一個 function 模擬 FSM 運作流程，識別 C 語言正確的識別字(Identifier)與非負整數(Number)。每項資料項目以一或多的空白結束，以句點(.)結束。

需偵測不合法(Invalid)的輸入。

FSM Transition Symbol

Digit(1) 0, 1, ..., 9  
Period (2) .  
Blank(3) 空白  
Letter(4) A, ..., Z, a, ..., z  
Letter(5) A, ..., Z, a, ..., z  
Digit(6) 0, 1, ..., 9  
Underscore(7) \_  
Blank(8) 空白  
Digit(9) 0, 1, ..., 9  
Blank(10) 空白  
Others1(11) 非空白、非 Digit、非 Letter、非底線  
Blank(12) 空白  
Others2(13) 非空白、非 Digit  
Others3(14) 非空白

Input:

rate R2D2 48 2 time 555666 .

Output:

rate - Identifier

R2D2 - Identifier

48 - Number

2 - Number

time - Identifier

555666 - Number

-----  
Input: 4132 6rte r\_yg t#ee 6677 .

Output:

4132 - Number

6rte - Invalid

r\_yg - Identifier

t#ee - Invalid

6677 - Number

-----

026.

所有的正整數都可以以連續的正整數相加的和來表達。

例如：9 有 3 種表達方式

2+3+4

4+5

9

給一個正整數  $N(N>0)$ ，請計算出  $N$  有多少種以連續的正整數相加的和來表達的方式。

輸入說明：

輸入一個正整數  $N$  ( $N>0$ )

輸出說明：

輸出有哪幾種相加方式。

輸出順序由小到大。

不合法的輸入則輸出 E。

-----  
input:

9

output:

2+3+4

4+5

9

-----  
input:

40

output:

6+7+8+9+10

40

-----  
input:

-40

output:

E

-----  
-----  
027.

\*\*\*請注意，double 強制轉成 int 有時候會有問題

\*\*\*例如 pow(10,2)得出 100.00

\*\*\*但強制轉成 int 會變成 99

使用切割面積加總法公式，

$T = (h/2)(f(p) + f(q) + 2 f(x_i))$ ，

$h = (q-p)/n$ ， $h$  是 double，

計算  $f(x)$  的面積，到小數點第五位四捨五入。

$x$  值從區間起始  $p$  到區間終點  $q$  的面積，

$n$  為切割數，初始設定  $n_0=2$ ，每次  $n_{j+1} = n_j + 1$ 。

$err$  為精確小數位數，例如  $err=6$ ，即  $n_{j+1}$  和  $n_j$  兩次切割，

計算出的面積值的差小於 10 的負 6 次方，亦即小於 0.000001。

$f1(x) = ((a-x^2))^{(1/2)}$ ,  $a$  為常數。  $f2(x) = (ax^3 + 7x)/(a+x)^{(1/2)}$ ,  $a$  為常數。

輸入說明：

option a p q err 0

option 輸入 1 代表計算  $f1(x)$  面積，輸入 2 代表計算  $f2(x)$  面積，輸入 0 為停止  
接著依序，輸入 a, p, q, err

此題測試資料為連續輸入一至多筆資料。

若輸入多筆資料 輸入格式如下

option a p q err option a p q err option a p q err 0

-----  
Input

1 4 -2 1 9 0

Output

5.05481

-----  
Input

2 5 1 2 7 0

Output

11.37718

-----  
Input:

2 5 -2 1 7 0

Output:

20.22476

-----  
HINT

方程式可能會跑到  $y < 0$  的地方，此時要算面積，必須將  $y$  取絕對值。

若  $err=3$ ,  $Area_{(j+1)}$  跟  $Area_{(j)}$  的差小於 10 負 3 次方

$fabs(Area_{(j+1)} - Area_{(j)}) * 10 * 10 * 10 < 1$

若以整數比較會好一點



```
int k=fabs(Area_(j+1)-Area_(j))*10*10*10*10;  
if (k>10) {...}
```

面積差，為兩次計算出面積的差值(相減)。

數值分析的誤差是兩次面積計算的差值除以，前一次的面積值。

028.

### 神奇寶貝進化計算器

在 pokemon go 遊戲中，傳送一隻神奇寶貝給博士可以得到 1.5 公升進化藥水，每完成一隻神奇寶貝的進化也可以得到 1.5 公升進化藥水。

現在玩家想要在最短時間內完成最多次的進化，請幫助玩家最多可以進化幾隻神奇寶貝

已進化和未進化的神奇寶貝都可以被傳送走，傳送走該神奇寶貝即消失

每隻神奇寶貝只能進化一次，已進化的神奇寶貝不能再進化，僅能留下來或被傳送，傳送一樣可以拿 1.5 公升進化藥水

輸入說明：

一次測試會輸入三個數字

第一個數為神奇寶貝進化需要消耗的進化藥水  $N$  ( $-100 < N < 0$ )，必須為負小數，顯示到小數點後一位，例如:-12.3 或-99.0

第二個數為玩家所擁有的進化藥水  $C$  ( $0 \leq C \leq 30000$ )，必須為正小數，顯示到小數點後一位，例如:3255.6 或 553.0

第三個數為玩家所擁有的未進化神奇寶貝數量  $W$  ( $0 \leq W \leq 10000$ )，必須為正整數，例如:5。

不合法的輸入如下，遇到不合法輸入則輸出 E

- (1)當需輸入正數，但使用者輸入負數時
- (2)當需輸入負數，但使用者輸入正數時
- (3)當需輸入整數，但使用者輸入小數時

輸出說明：

輸出進化次數。

範例輸入：

-12.3 22.5 5

範例輸出：

2

-----

範例輸入說明：

1. 先進化 1 隻神奇寶貝，用掉 12.3 公升進化藥水，進化後再得 1.5 公升進化藥水，剩 11.7 公升進化藥水和 4 隻未進化神奇寶貝和 1 隻已進化神奇寶貝。
2. 把 1 隻已進化神奇寶貝傳送給博士換取 1.5 公升進化藥水，剩 13.2 公升進化藥水和 4 隻未進化神奇寶貝。
3. 再進化 1 隻神奇寶貝，得 1.5 公升進化藥水，最後剩 2.4 公升進化藥水及 3 隻未進化神奇寶貝和 1 隻已進化神奇寶貝。
4. 因為剩下的 4 隻神奇寶貝即使傳送給博士，得到的進化藥水也不夠進化的量，所以總共只能進化 2 次。

-----測資-----

input：

56 22.5 5

output：

E

-----

input：

-12.3 31.1 3

output：

3

-----

input：

-12.3 18.6 5

output：

2

-----

-----

029.

最大質數

輸入一個數  $x$ ，根據以下規則得到  $k$ ，並找出 2 到  $k$  之間最大的質數。  
處理輸入請使用有限狀態機，不要使用陣列

規則：

1. 若  $x$  為正整數，且  $2 \leq x \leq 1000$ ，則  $k = x$
2. 若  $x = 0$  或  $x = 1$ ，則  $k = 1000$
3. 若  $x$  為負整數，則  $k = x * -11$
4. 若  $x$  為正小數，則  $k =$  小數點前面的數
5. 若  $x$  為負小數，則  $k =$  小數點後面的數

規則範例：

1.  $x = 50$ ， $k = 50$
2.  $x = 0$ ， $k = 1000$
3.  $x = -90$ ， $k = 990$
4.  $x = 66.22$ ， $k = 66$
5.  $x = -33.44$ ， $k = 44$

-----

輸入說明：

輸入一個數字。

輸出說明：

根據規則找出  $k$ ，輸出 2 到  $k$  之間最大的質數。  
不合法的輸入則輸出 Error!

-----

輸入範例：

50

輸出範例：

47

-----

輸入範例：

0

輸出範例：

997

輸入範例：

-90

輸出範例：

983

輸入範例：

66.22

輸出範例：

61

輸入範例：

-33.44

輸出範例：

43

輸入範例：

1001

輸出範例：

Error!

030.

方程式計算

(此題不使用陣列)

寫一個 function 輸入 XY 座標系統的兩個座標  $(x1, y1)$ ,  $(x2, y2)$  ;

輸出兩個座標經過的 XY 方程式  $y=mx+b$ ;

```
m=(y1-y2)/(x1-x2)
b=(x2y1-x1y2)/(x2-x1)
void f1_m(int x1, int y1, int x2, int y2, double m, double b);
```

寫一個 function 輸入 XY 座標系統的兩個座標 (x1, y1), (x2, y2);  
輸出兩個座標經過的 XY 方程式  $y=m1/m2x+b1/b2$ ;

```
m1=(y1-y2), m2=(x1-x2),
b1=(x2y1-x1y2), b2=(x2-x1),
void f2(int x1, int y1, int x2, int y2, int m1, int m2, int b1, int b2);
```

-----

輸入說明:

XY 的兩個座標 x1, y1, x2, y2, 均為整數。

輸出說明:

(1)  $y=mx+b$ ,

m, b 計算至小數第二位。

(2)  $y=mx+b$

m, b 以分數表達，分數不需要化簡。

=>方程式有可能沒有 x 項，或沒有 y 項。

=>沒有 x 項則  $y=b$ ，沒有 y 項則  $x = -b/m$ 。

=>若 m, b 為整數，則使用整數表達。

不合法的輸入則輸出 Error!

-----

輸入範例:

3 4 -3 0

輸出範例:

$y=0.67x+2$

$y=4/6x+2$

-----

輸入範例:

3.3 4 -3 0

輸出範例:

Error!

-----  
輸入範例：

1 0 0 1

輸出範例：

$y = -x + 1$

$y = -x + 1$

-----

-----

031.

(此題不使用陣列，必須使用指標)

寫一個 function 輸入 N 組矩形 XY 座標中的兩個點，輸出 N 組矩形中的最大的周長和最大的面積。

程式中必須使用以下 function 定義，計算一個矩形的周長與面積。

-----

void compute(int x1, int y1, int x2, int y2, int \*perimeter, int \*area);

第一個點 (x1, y1)

第二個點 (x2, y2)

計算周長結果 \*perimeter

計算面積結果 \*area

-----

輸入幾組矩形的兩個點，輸入-1 停止輸入。

輸出最大的面積和最大的周長。

-----

輸入說明：

分別輸入幾組矩形的兩個點，

第一筆輸入第一個矩形兩個點，x1, y1, x2, y2，2 組座標為對角關係

換行再輸入第二個矩形兩個點，以此類推，直到輸入 -1 停止。

計算矩形邊長時請使用絕對值，若同一行的 2 組座標無法形成矩形則輸出"Error"後結束

-----

輸出說明：

輸出所有矩形中最大的面積和最大的周長。

input:

-10 -10 20 60

2 2 52 52

85 8 5 38

-1

output:

2500

220

-----

032.

分數計算

(此題不使用陣列，必須使用指標)

程式碼必須使用以下 function 定義，計算兩個分數的相加。

```
void add(int n1, int d1, int n2, int d2, int *numerator, int *denominator);
```

n1: 第一個數的分子

d1: 第一個數的分母

n2: 第二個數的分子

d2: 第二個數的分母

\*numerator: 相加結果的分子

\*denominator: 相加結果的分母

程式碼必須使用以下 function 定義，計算兩個分數的相乘。

```
void multiply(int n1, int d1, int n2, int d2, int *numerator, int *denominator);
```

n1: 第一個數的分子  
d1: 第一個數的分母  
n2: 第二個數的分子  
d2: 第二個數的分母  
\*numerator: 相乘結果的分子  
\*denominator: 相乘結果的分母

-----

輸入說明:

輸入二行，每一行代表一個分數

-----

輸出說明:

輸出分數相加結果

輸出分數相乘結果

若有輸入分數的分母或分子為 0，則輸出 ERROR。

若分數大於 1，則分數部分要加括號，如一又六分之一輸出為 1(1/6)

若為負數，負號在最前面。

若輸出為 0，則顯示 0

輸出的結果必須化簡為最簡分數

-----

範例輸入:

1/2

2/3

範例輸出:

1(1/6)

1/3

-----

範例輸入:

0/2

2/3

範例輸出:



ERROR

ERROR

-----

033.

小明有個  $N \times M$  ( $1 \leq N \leq 10, 1 \leq M \leq 10$ ) 的抽抽樂矩陣，裡面放著 1~9 號等獎項，小明在走樓梯的時候不小心手滑讓抽抽樂滾走，抽抽樂有三種滾動方式：向右轉 90 度，向左轉 90 度，向右轉 180 度，請幫助小明找出目前的獎項排列。

輸入說明：

3 3 (矩陣大小為  $3 \times 3$ )(數字間有空格)

123 (輸入矩陣內的獎項)(數字之間沒有空格)(數字可重複)

321

456

12321322 (滾動方式 1=: 向右轉 90 度 , 2=: 向左轉 90 度, 3=: 向右轉 180 度)

輸出說明：

輸出旋轉後的矩陣。(數字間沒有空格)

不合法的輸入則輸出 E。

-----測資-----

input:

3 4

1234

3216

1397

12321233

output:

131

322

913

764

-----

input

1 10

123456789

3321

output:

E

-----  
input

1 10

1234567893

123321313

output:

1

2

3

4

5

6

7

8

9

3

-----  
-----  
034.

### 整數的交集聯集

請設計程式，利用兩個一維整數陣列儲存兩個正整數集合 A, B，

兩個集合 A, B 的陣列大小為 20，正整數 1~100；

程式功能為：

- (1) 新增集合元素：輸入一集合與一正整數，若新增的元素與該集合內元素有重複，則無法新增。
  - (2) 刪除集合元素：輸入一集合與一正整數，若集合中無該元素，無法刪除。
  - (3) 計算集合是否空集合。
  - (4) 計算兩個集合的交集。
  - (5) 計算兩個集合的聯集。
  - (6) 計算集合是否為另一個集合的子集合。
-

輸入說明：

每一行第一數字輸入整數指令選項，意義如下：

0: 結束 exit

1: 清空 A，輸出 A 和 B 的內容。

2: 清空 B，輸出 A 和 B 的內容。

3: 新增元素到 A，後面接一個正整數為要插入的元素，  
輸出 A 和 B 的內容。

4: 新增元素到 B，後面接一個正整數為要新增的元素，  
輸出 A 和 B 的內容。

5: 從 A 刪除一個元素，後面接一個正整數為要刪除的元素，  
輸出 A 和 B 的內容。

6: 從 B 刪除一個元素，後面接一個正整數為要刪除的元素。  
輸出 A 和 B 的內容。

7: 輸出 A, B 聯集。comm

8: 輸出 A, B 交集。

9: 輸出 A 是否為 B 的子集合，若是則輸出 1，否則輸出 0。

輸出說明：

輸出集合格式

集合名稱:[集合元素,集合元素,...]，中間沒有空白，  
元素順序依照輸入先後順序。

交集與聯集的元素順序，先以 A 的元素順序再以 B 的元素順序。

每一指令輸出一行。

空集合為所有集合的子集合

-----

Input :

3 4

3 7

4 8

4 9

5 4

6 8

3 8

7

8

9  
0

Output:

A:[4,]B:[]  
A:[4,7,]B:[]  
A:[4,7,]B:[8,]  
A:[4,7,]B:[8,9,]  
A:[7,]B:[8,9,]  
A:[7,]B:[9,]  
A:[7,8,]B:[9,]  
[7,8,9,]  
[]  
0

---

035.

### 中序轉後序

給予一個由變數加上+\*/四則運算與冪次^運算的數學式，變數用一個小寫英文字母表示。  
運算優先次序是：冪次大於乘除大於加減。中序表示法是運算符號在兩運算子中間，可藉括號表示優先運算次序。

請將用中序表示的運算式改成後序表示。後序表示法不需藉由括號區分運算的先後次序。

### 輸入說明

輸入一行長度在 512 個字母內的中序式，使用字元陣列(提示:"%s")存取

### 輸出說明

一行長度在 512 個字母內的字串，用中序表示法表達的算式。

輸出說明：後序表示的算式。

---

input:

a+b\*(c-d)^e

output:

abcd-e^\*+

input:

$(a-b)^c+(d-e)^f$

output:

$ab-c^de-f^+$

-----

036.

數獨

在 9 格寬×9 格高的大九宮格中有 9 個 3 格寬×3 格高的小九宮格，每一列與每一行的數字均須包含 1~9，不能缺少，也不能重複。每一小九宮格(3\*3 的九宮格)的數字均須包含 1~9，不能缺少，也不能重複。

輸入一組測試資料為 9x9 的矩陣,判斷九宮格數字是不是一個數獨的正解。

輸入說明：

輸入九列數據，每一列輸入為 9 個整數分別由空格分開。

輸出說明：

輸出 Yes or No 代表九宮格數字是不是一個數獨的正解。

-----

輸入範例

```
1 2 3 4 5 6 7 8 9
2 3 4 5 6 7 8 9 1
3 4 5 6 7 8 9 1 2
4 5 6 7 8 9 1 2 3
5 6 7 8 9 1 2 3 4
6 7 8 9 1 2 3 4 5
7 8 9 1 2 3 4 5 6
8 9 1 2 3 4 5 6 7
9 1 2 3 4 5 6 7 8
```

輸出範例

No

輸入範例

1 9 3 2 6 5 4 7 8  
7 8 2 3 1 4 9 5 6  
4 5 6 9 7 8 1 3 2  
2 3 4 8 5 1 6 9 7  
9 6 5 4 3 7 2 8 1  
8 7 1 6 9 2 3 4 5  
3 1 9 5 8 6 7 2 4  
5 2 7 1 4 3 8 6 9  
6 4 8 7 2 9 5 1 3

輸出範例

Yes

037.

以陣列實做長整數的運算

將長整數儲存於一維陣列，長度為 60 位。  
計算兩數的相加、相減。

輸入說明：

兩個小於 60 位數的數字

輸出說明

兩數的相加、相減結果

輸入範例：

74678186416540450856046456406565453745354346846  
42541264516146186481412634534534535674461

輸出範例：

74678228957804967002232937819199988279890021307

74678143875275934709859974993930919210818672385

038.

五子棋

檢查 10\*10 五子棋可以構成 5 個連為一線的位置。

1 表示有放棋子，0 表示沒有放旗子。

輸入說明

輸入 10\*10 的資料

輸出說明

可構成 5 個連為一線的位置。例如

```
0 0 0 0 0 0 0 0 1 0
0 0 0 0 0 0 1 1 0 0
0 0 0 0 0 0 1 0 0 0
0 0 0 0 0 0 1 1 1 1
0 0 0 0 1 0 1 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 1 1 1 1 0 0 0 0
0 0 0 1 0 0 0 0 0 0
```

可以增加 5 個連為一線，以下圖表示。

```
0 0 0 0 0 0 x 0 1 0
0 0 0 0 0 0 1 1 0 0
0 0 0 0 0 0 1 0 0 0
0 0 0 0 0 x 1 1 1 1
```

```
0 0 0 0 1 0 1 0 0 0
0 0 0 x 0 0 x 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 x 1 1 1 1 x 0 0 0
0 0 0 1 0 0 0 0 0 0
```

其位置為

```
0 6
3 5
5 3
5 6
8 1
8 6
```

-----

SampleInput:

```
0 0 0 0 0 0 0 0 1 0
0 0 0 0 0 0 1 1 0 0
0 0 0 0 0 0 1 0 0 0
0 0 0 0 0 0 1 1 1 1
0 0 0 0 1 0 1 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0
0 0 1 1 1 1 0 0 0 0
0 0 0 1 0 0 0 0 0 0
```

Sample Output:

```
0 6
3 5
5 3
5 6
8 1
8 6
```



-----

039.

井字遊戲

-----

輸入說明：

第 1 行：

輸入誰先下，1 代表 Computer，0 代表 Player。

第 2 行~第 N 行：

第一種情況：兩個 0, 1, 2 數字

分別輸入 Computer 和 Player 的座標，直到一方贏才停止。

第二種情況：

w: 輸出所有 Computer 可能贏的座標，若沒有答案則輸出 no

d: 輸出 Computer 或 player 即將要贏，若沒有答案則輸出 no

座標位置如下：

00 | 01 | 02 |

-----

10 | 11 | 12 |

-----

20 | 21 | 22 |

-----

-----

輸出說明：

(1)

一旦有一方贏，要顯示是 Computer 或 Player 贏。

若為 Computer 贏，輸出 Computer win

若為 Player 贏，輸出 Player win

若為平手，輸出 fair

(2)

有座標重複輸入時，輸出 input error

(3) 輸入 w

輸出所有 Computer 可能贏的座標

(4) 輸入 d

輸出誰即將贏與贏的座標

Player will win with (x, y)

Computer will win with (x, y)

-----  
輸入範例說明

1 (Computer 先下)

0 0

1 1

0 2

d (顯示誰即將贏)

0 2 (已重複，要輸出 ERROR)

2 2

2 0

w (顯示所有 Computer 可能贏的座標)

1 0

0 1

-----  
輸出範例說明

Computer will win with (0, 1) (Computer 可能贏)

input error

Computer will win with (0, 1) (Computer 可能贏的第一個座標)

Computer will win with (1, 0) (Computer 可能贏的第二個座標)

Computer win

-----  
Sample Input

1

0 0

1 1

0 2

d

0 2

```
2 2
2 0
w
1 0
0 1
```

-----

Sample Output

```
Computer will win with (0, 1)
input error
Computer will win with (0, 1)
Computer will win with (1, 0)
Computer win
```

-----

040.

矩陣乘法

若  $A$  為  $m \times n$  矩陣， $B$  為  $n \times p$  矩陣，則他們的乘積  $AB$ (有時記做  $A \cdot B$ )會是一個  $m \times p$  矩陣。

範例：

$A =$

```
|a00|a01|a02|
|a10|a11|a12|
|a20|a21|a22|
```

$B =$

```
|b00|b01|
|b10|b11|
|b20|b21|
```

$A * B =$

```
|a00*b00+a01*b10+a02*b20|a00*b01+a01*b11+a02*b21|
|a10*b00+a11*b10+a12*b20|a10*b01+a11*b11+a12*b21|
|a20*b00+a21*b10+a22*b20|a20*b01+a21*b11+a22*b21|
```

## 輸入說明

每組測資第一行四個數字 a b c d

代表第一個矩陣有 a 列 b 行

第二個矩陣有 c 列 d 行

接下來 a 行，每行 b 個數字  
c 行，每行 d 個數字

a、b、c、d 為 1~99 的整數

測資第二行為輸入矩陣

每個數字以空白隔開

不懂請參考範例輸入

-----

## 輸出說明

輸出相乘之後的矩陣

每個數字以空白隔開

兩矩陣不能相乘請輸出 Error

## 範例輸入

```
3 2 2 3
1 2
3 4
5 6
1 2 3
4 5 6
```

## 範例輸出

```
9 12 15
19 26 33
29 40 51
```

範例輸入

1 2 3 4

範例輸出

Error

-----

041.

猜數字

程式的使用者設定一個答案  $X$ ，四位數，0~9 不重複。

程式的使用者輸入 4 位數字，以及這些 4 位數與答案  $X$  的相似程度。

程式必須根據相似程度資料，輸出使用者設定的答案  $X$ 。

相似程度的規則

(1)  $Y_i$  中有 1 位數字跟答案  $X$  一樣，且所在位置相同，例如千位對千位，或百位對百位，記為 1A。

若 2 位都有這情況，就是 2A。

例如  $X=1234$ ， $Y_1=1856$ ，兩者都有 1，位置都在千位，因此相似程度是 1A。

(2)  $Y_i$  中有 1 位數字跟答案  $X$  一樣，但所在位置不同，記為 1B。

若 2 位都有這情況，就是 2B。

例如  $X=1234$ ， $Y_1=8561$ ，兩者都有 1，但位置不同，所以是 1B。

(3) 以上兩條規則以 (1)優先，之後再考慮 (2)。

(4) 輸入至猜出答案為止

-----

輸入範例說明：

每一行輸入 4 位數字  $Y_i$ ，以及相似程度 ?A?B。

假設使用者設定的答案是 4237

輸入

1968,0A0B 數字都沒有對，所以相似程度為 0A0B。

7052,0A2B 有 2 個數字對 (7, 2) , 但位置不對, 相似程度為 0A2B。  
2347,1A3B 有 1 個數字對且位置對 (7), 3 個數字對 (2, 3, 4), 相似程度為 1A3B。  
3427,1A3B 有 1 個數字對且位置對 (7), 3 個數字對 (3, 4, 2), 相似程度為 1A3B。  
2473,0A4B 4 個數字對, 位置不對。

輸出範例說明:

4237 輸出使用者設定的答案

Sample Input

1968,0A0B  
7052,0A2B  
2347,1A3B  
3427,1A3B  
2473,0A4B

Sample Output

4237

-----

042.

數字迷宮

數字迷宮為一個二維的數字 (0-9) 陣列。

你可以用直角方向 (東、西、南、北)在迷宮中尋訪。

假設每一格的數字代表造訪該格的成本，求出從入口走到出口所需的最小成本。

-----

10131112191

-----

17131419191

-----

11171515131

-----

12131412151

-----

給你一  $N \times M$  ( $1 \leq N, M \leq 9$ ) 的數字迷宮，  
你必須求出從左上角走到右下角所需的最小成本。

上面範例的解答為 24。

---

#### 輸入說明

輸入的第一行為列數 N，第二行為行數 M，  
接下來 N 行每行代表迷宮的一行，含有以空白隔開的迷宮數字。

#### 輸出說明

對於每個迷宮，請輸出所需的最小值於一行。

---

#### 範例輸入

```
4
5
0 3 1 2 9
7 3 4 9 9
1 7 5 5 3
2 3 4 2 5
```

#### 範例輸出

```
24
```

---

#### 走出迷宮

給予一個二維整數陣列，以 1 表示牆壁，0 表示通道，構成一個迷宮。你可以用直角方向 (上 u, 下 d, 左 l, 右 r) 在迷宮中移動，想辦法從入口走到出口。入口座標固定為(0,1)，出口座標固定為(N-2, M-1)如下所示：

N 為 5

M 為 7

```
1 0 1 1 1 1 1
1 0 0 0 0 0 1
1 0 1 0 1 1 1
1 0 1 0 0 0 0
1 1 1 1 1 1 1
```

給你一個 N×M 的迷宮(3 ≤ N,M ≤ 9，N,M 為奇數)，你必須求出從入口走到出口的路徑，  
上例的解答輸出為 drrddrrr

---

#### 範例輸入

5

7

1 0 1 1 1 1 1

1 0 0 0 0 0 1

1 0 1 0 1 1 1

1 0 1 0 0 0 0

1 1 1 1 1 1 1

範例輸出

drrddrrr