

# 送分題 – Hello World

---

(30 分)

## 前言

比賽開始了！

趕快驗證一下，

網路是否設定正確？

上傳競賽程式是否順利？

程式解答是否用 `STDOUT` 輸出？

都沒問題，30 分就到手了！繼續 ... 衝！衝！衝！

## 問題描述

請寫一個程式輸出 `Hello World!`

## 輸入格式

本題無需輸入值

## 輸出格式

`[A~Z][a~z]`, 空格, 以及及常用英文符號。

## 資料範圍

英文 `[A~Z][a~z]`, 空格, 以及驚嘆號 `“!”`

## 資料範例

輸入範例 1

(無輸入值)

輸出範例 1

`Hello World!`

## 範例解釋

輸入範例 1, 簡單而快樂的輸出 `Hello World!`

# 問題 1 - 費波那契數列 (Fibonacci)

---

(5 分)

## 問題敘述

費波那契數列 (Fibonacci numbers) 透過遞迴的方式來定義的，數列中的第 0 項為 0，第 1 項為 1，接著第 2 項開始的值為前面兩項的值相加，以此類推。

$F = 0, 1, 1, 2, 3, 5, 8, 13, \dots$

寫一支程式算出費波那契數列的第  $N$  項為多少

## 輸入格式

$N$

$N$  為一正整數

## 輸出格式

輸出另一正整數

## 資料範圍

$1 < N \leq 50$

## 資料範例

輸入範例 1

8

輸出範例 1

21

輸入範例 2

10

輸出範例 2

55

輸入範例 3

14

輸出範例 3

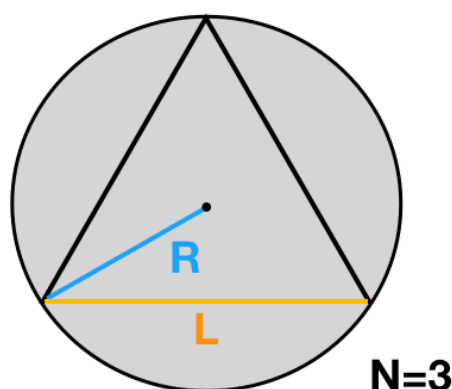
377

## 問題 2 - 圓形和多邊形 (Circle & Polygon)

(10 分)

### 問題敘述

給定兩整數  $N$  和  $L$ ，當  $N \geq 3$  時必定能在平面上畫出一個單邊長為  $L$  的正  $N$  邊形，請找出能完整覆蓋此正  $N$  邊形的最小圓，並輸出此圓的半徑  $R$ 。舉例正三邊形的例子如下圖：



### 輸入格式

N1 L2  
N2 L2

每行輸入皆包含兩個整數，依序代表正多邊形的邊長數  $N$  和邊長  $L$ ，兩整數間以一個空白字元分隔，每一行的結尾為 `\n` 換行符號，讀到 EOF 表示結束。

### 輸出格式

R1  
R2

每行輸出圓的半徑  $R$ ，四捨五入取至小數點後三位。

### 資料範圍

$3 \leq N \leq 1000$   
 $0 \leq L \leq 1000$

Hint: 在數學中  $\pi$  值約等於 3.14159265359

## 資料範例

### 輸入範例 1

3 3  
5 3  
9 100

### 輸出範例 1

1.732  
2.552  
146.190

### 輸入範例 2

3 11  
4 10  
5 9  
6 8  
7 7

### 輸出範例 2

6.351  
7.071  
7.656  
8.000  
8.067

## 問題 3 – 服務機器人最適路徑 (Pass Through)

(15 分)

### 問題敘述

越來越多的服務機器人出現在我們的身邊，服務機器人可以用更精確而有效率的方式完成服務的工作。其中最初步的要求，就是用最短的時間照顧到每個需要服務的對象。

所以機器人需要先確認所有需要服務對象的位置，尋找最適合的路徑到達身邊。但是因為硬體設備的限制，機器人只能往前、後、左、右四個方向前進，不能斜向移動，每移動一次算一步。

此外，受服務對象還有不同的優先權，優先權共分三級 (1~3)，數值越高的對象必須優先接受到服務。

為了讓機器人服務最有效率，在地圖上有兩個充電站，分別設置在左上角及右下角，為了簡化問題，我們設計讓機器人從地圖左上角的充電站出發，服務完所有對象後回到右下角充電站準備下次任務。

另外，即使一位低優先權的對象位於服務另一位高優先權的對象的必經(或可能)路徑上，也不用考慮先提供服務，仍須依照優先權順序決定順序。

最後機器人完成任務的評價是以行走的總步數計算，而以最小數字為最佳。請協助機器人在地圖上找到最佳的路徑吧！

### 輸入格式

M N

0 0 0 0 0 0 0 0

0 0 2 0 0 0 3 0

0 0 0 1 0 0 0 0

0 0 0 0 0 2 0 0

輸入第一行正整數 M 及 N, 代表接下來的地圖是一個 M x N 的矩陣。

接下來會有 M 列 (row) 及 N 行 (column) 以一格空白區隔的連續數字表示地圖上的座標，每個座標可以用 (m,n) 的表示法。其中會有若干個座標標示為 1 到 3 的數字，表示服務對象的優先權，其餘均為 0。機器人從最左上角 (0,0) 充電站出發，服務完所有對象後，機器人需要回到最右下角 (M-1,N-1) 充電站準備下次任務。

### 輸出格式

C

```
(0,0) (m1,n1) (m2,n2) (m3,n3) (M-1,N-1)
(0,0) (m1,n1) (m3,n3) (m2,n2) (M-1,N-1)
...
```

第一行 C 表示服務完成回到充電站的最小總步數。

第二行之後表示最適當路徑，從(0,0)出發，以(M-1,N-1)結束。座標間以一格空白區隔，座標內逗號前後沒有空白。

若有多條路徑，則繼續在第三行之後繼續列出，多條路徑不同 (可交換) 的部分以 m 座標先從小到大排序，再以 n 座標排序。以上表輸出格式為例：

- $m2 \leq m3$
- 若  $m2 = m3$  , 則  $n2 < n3$

## 資料範圍

$2 \leq M, N \leq 20$

地圖大小在  $2 \times 2 \sim 20 \times 20$  之間

## 資料範例

### 輸入範例 1

```
5 5
0 0 0 0 0
0 2 0 1 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
```

### 輸出範例 1

```
8
(0,0) (1,1) (1,3) (4,4)
```

### 輸入範例 2

```
9 10
0 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 0 0 0
0 0 0 2 0 0 0 0 0 0
0 0 0 0 0 2 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 3 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
```

### 輸出範例 2

```
31
```

(0,0) (7,2) (3,3) (4,5) (1,7) (8,9)

### 輸入範例 3

```
15 8
0 0 0 0 0 0 0 0
0 2 0 1 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 3 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 2 0 2
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 1 0 0 3 0 0 0
0 0 0 0 0 0 1 0
```

### 輸出範例 3

```
61
(0,0) (5,1) (13,4) (10,5) (10,7) (1,1) (1,3) (13,1) (14,6) (14,7)
(0,0) (5,1) (13,4) (10,7) (10,5) (1,1) (1,3) (13,1) (14,6) (14,7)
```

## 範例解釋

範例 1:

輸入為一個 5x5 的地圖，在 (1,1) 位置有一位優先權 2 的服務對象，在(1,3)位置有一個優先權 1 的服務對象。

因此，服務路徑從 (0,0) 出發，完成所有服務後，回到 (4,4) 最短路徑只有一條，總行進步數為 8。輸出第一行 8, 第二行表示此唯一路徑。

範例 2:

輸入為一個 9x10 的地圖，在 (7,2) 位置有一位優先權 3 的服務對象，其他對象位置依此類推。

因此，服務路徑從 (0,0) 出發，完成所有服務後，回到 (4,4) 最短路徑只有一條，總行進步數為 31。輸出第一行 31, 第二行表示此唯一路徑。

範例 3:

輸入為一個 15x8 的地圖，在 (5,1) 及 (13,4) 位置各有一位優先權 3 的服務對象，其他對象位置依此類推。

因此，服務路徑從 (0,0) 出發，完成所有服務後，回到 (4,4) 最短路徑將會有 2 條，總行進步數為 61。輸出第一行 61, 第二行開始，依照輸出格式說明的順序列出路徑。

例如以下兩條路徑，因為 (10,5) 及 (10,7) 兩個位置不論服務次序先後，均符合最短路徑，因此以 (10,5) 在前先列出。

(0,0) (5,1) (13,4) (10,5) (10,7) (1,1) (1,3) (13,1) (14,6) (14,7)  
(0,0) (5,1) (13,4) (10,7) (10,5) (1,1) (1,3) (13,1) (14,6) (14,7)



## 問題 4 – 數字驗證系統 (Digit Error Detector)

---

(10 分)

### 問題描述

隨著人工智慧走進人類的生活，透過電腦做分析、學習和判斷來解決人類日常生活面臨的各種需求，以協助改善人類生活的品質，舉凡自駕車、居家照護、醫療、智能家庭等許多方面，都是當今如火如荼正在努力的創新發展項目。

在這其中因為與人類或其他動植物接觸，安全性的問題日益受到重視，例如：駭客透過攻擊（提供錯誤資訊）高速行進中的汽車內建的感應器，造成汽車立即停止及內部人的傷亡。因此，如何驗證這些人工智慧系統外部接受到的訊息是很大的挑戰。

此系統使用以下方式及規則驗證其數字系統是否有問題，

- 輸入檔案內可能有多行算式，
- 每一行有 3 個數字，第一個數字接一個運算子 (加減乘除之一)，第二個數字和第三個數字間為一個等號。
- 數字是整數，且等號(=) 左右都要檢查，
- 運算子僅考慮：加 (+), 減(-), 乘(\*), 除(/), 等號(=)，
- 除法要捨去小數，
- 當檔案內的所有算式(加減乘除) 都正確時，輸出 0，
- 當一個數字必須和另一個數字互換時，輸出依小到大排列如範例 1 之 78,
- 如果太多可能，或超過兩個數字互換，或無解時則輸出 100。

舉例如下：

### 資料範圍

數字都是正整數(無括號)並都在 10000 以下，數學式子在 10 行以下。

### 資料範例

#### 輸入範例 1

8+7=15  
53/6=7  
3+4=8  
4+5=9

#### 輸出範例 1

78

### 輸入範例 2

$$3+4=7$$

$$4+5=9$$

$$7+8=15$$

### 輸出範例 2

0

### 輸入範例 3

$$9-2=4$$

$$4+3=7$$

### 輸出範例 3

100

### 範例解釋

範例 1，原檔案中所有 7 換成 8，而所有 8 換 7 的話，可得正確運算式。

範例 2，因為等式皆正確，故為 0。

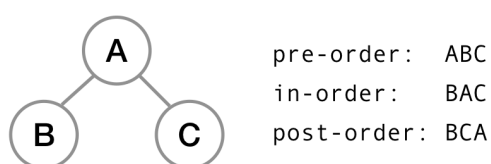
範例 3，因為題目有 25,69 兩組解，故為 100。(9-5=4,4+3=7 或是 6-2=4,4+3=7)

## 問題 5 - Binary Tree Traversals

(15 分)

### 問題敘述

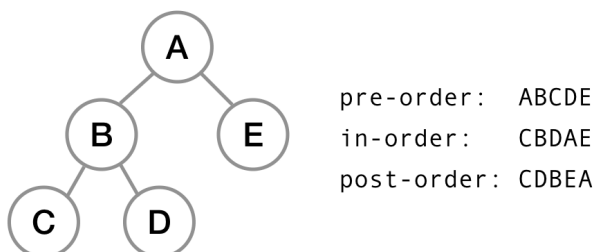
二元樹 (binary-tree) 是每個節點 (node) 最多只有兩個分支的樹狀結構，每個分支可能為單一節點 (node) 或是另一個子樹 (sub-tree)。二元樹有許多尋訪方式，其中包含了 pre-order、in-order、post-order 尋訪，範例如下：



以上圖為例，尋訪將從根節點 (root node) 開始出發，每一個節點的尋訪規則如下：

- \* pre-order 尋訪: 先訪問自身節點 A，接著訪問左分支 B，最後是右分支 C。
- \* in-order 尋訪: 先訪問左分支 B，接著訪問自身節點 A，最後是右分支 C。
- \* post-order 尋訪: 先訪問左分支 B，接著訪問右分支 C，最後是自身節點 A。

另一個範例如下：



題目將給定一個 binary-tree 的 in-order 和 post-order 尋訪結果，請計算出對應的 pre-order 尋訪。

### 輸入格式

第一行輸入為一個數字，代表接下來有多少筆測試資料。每筆測試資料以三個欄位表示一個 binary-tree，格式如下：

- \* 欄位 1：數字，表示此 binary-tree 有多少 node。
- \* 欄位 2：字串，為 in-order 尋訪結果。
- \* 欄位 3：字串，為 post-order 尋訪結果。

備註：每個 node 以不重複的 [A-Za-z] 單一字元 (char) 表示。

## 輸出格式

每筆測試資料以一行輸出 binary-tree 的 pre-order 尋訪結果。

### 輸入範例 1

1  
3 BAC BCA

### 輸出範例 1

ABC

### 輸入範例 2

1  
5 CBDAE CDBEA

### 輸出範例 2

ABCDE

### 輸入範例 3

3  
3 BAC BCA  
5 CBDAE CDBEA  
6 ABDECF DBEAFC

### 輸出範例 3

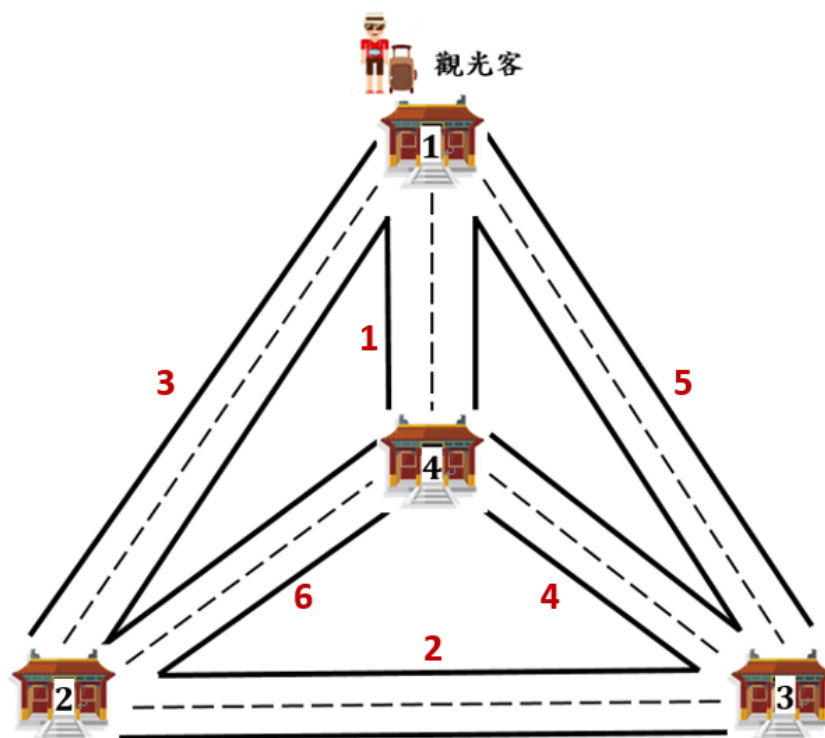
ABC  
ABCDE  
CAEBDF

## 問題 6 – 觀光路線 (Tourist Routes)

(20 分)

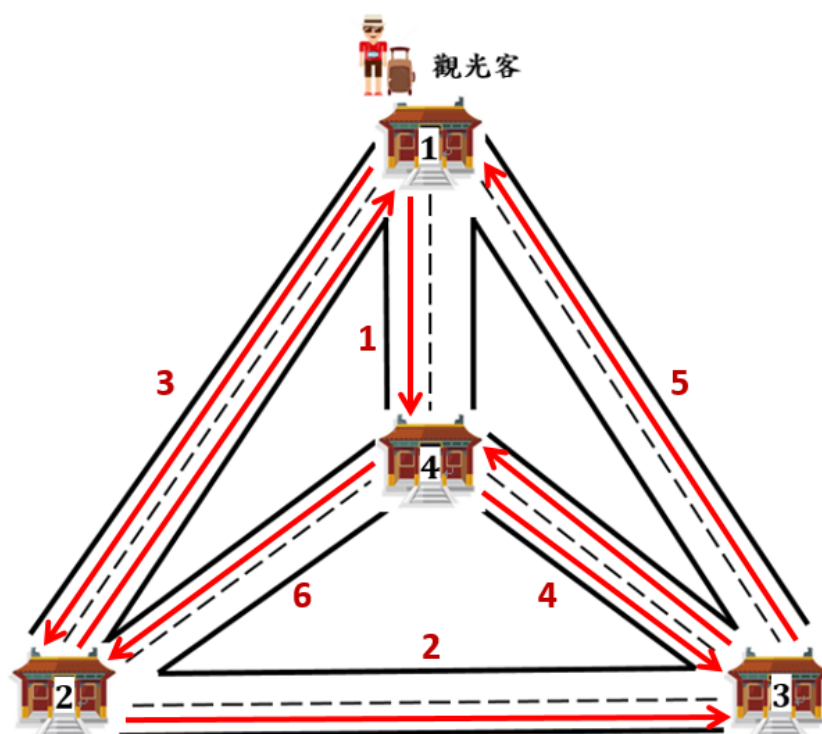
### 問題敘述

台南市是台灣著名的古都之一，市區有  $N$  座古蹟， $2 \leq N \leq 1500$ 。為了方便說明，我們將  $N$  座古蹟名稱以正整數  $\{1, 2, \dots, N\}$  來表示。 $N$  座古蹟有街道來連接，使得觀光客可從任一個古蹟出發，經由幾條街道抵達另一個古蹟。我們可以用圖形來表示這些古蹟跟街道的關係：節點表示古蹟；而連接結點的連結線則代表連接兩個古蹟之間的街道(如圖一，其中古蹟名稱以黑色數字表示，而紅色數字則代表行走街道所需花費的時間)。我們以符號  $(I, J)$  來表示連接古蹟  $I$  和古蹟  $J$  的連結線。每一條連結線  $(I, J)$  都結合一個權重  $w(I, J)$  來代表走完這條街道所要花費的時間， $w(I, J)$  為正整數滿足  $1 \leq w(I, J) \leq 888$ 。給定一個起始古蹟，請寫一個程式計算觀光客從起始古蹟出發，把每條街道都走過至少一次再回到起始古蹟所需花費的最短時間為何。

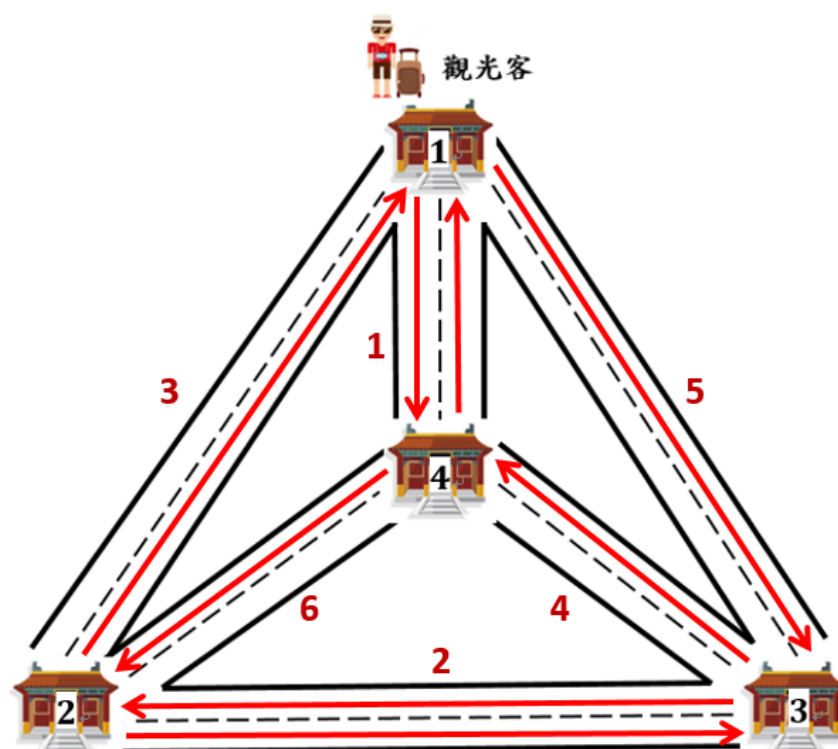


圖一

圖二說明其中一種走法為： $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 2 \rightarrow 1 \rightarrow 4 \rightarrow 3 \rightarrow 1$ ，可在 28 單位時間將所有街道都走過至少一次，然而此種走法所需的時間並非最短。事實上，此例中花費時間為最短的走法如圖三所示為  $1 \rightarrow 4 \rightarrow 2 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 1$ ，所花費時間為 24 單位。



圖二



圖三

## 輸入格式

第一行有三個數字，連續兩個數字之間以空白符號做區格。第一個數字  $2 \leq N \leq 1500$  代表圖形的節點(古蹟)個數；第二個數字  $1 \leq M \leq 1500$  代表圖形的連結線(街道)個數；第三個數字則代表起始古蹟之名稱。從第二行起連續有  $M$  行，表示  $M$  條連結線(街道)，每行有三個數字，連續兩個數字之間以空白符號做區格：前二個數字代表連結線的兩個端點，第三個數字代表連結線的權重。

## 輸出格式

依序輸出每組測資的解，每行輸出一個數字。

## 資料範圍

$2 \leq N \leq 1500$

古蹟個數  $N$ : 介於 2 與 1500 間的正整數

$1 \leq M \leq 1500$

連結線(街道)個數  $M$ : 介於 1 與 1500 間的正整數

$1 \leq w(I,J) \leq 888$

權重  $w(I,J)$ : 介於 1 與 888 間的正整數

## 資料範例

輸入範例 1:

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

輸出範例 1:

24

輸入範例 2:

```
12 20 1
1 2 8
1 8 5
2 3 6
1 9 1
2 10 2
8 9 1
9 10 1
10 3 1
8 7 2
9 12 3
10 11 1
3 4 1
7 12 1
12 11 2
11 4 1
7 6 6
```

12 6 2  
11 5 1  
4 5 2  
6 5 7

輸出範例 2：

64

## 範例解釋

範例輸入 1,

第 1 行: 4 6 1, 代表 4 個古蹟, 6 條街道, 將名稱 1 作為起始點

第 2 行: 1 2 3 代表 古蹟 1 到古蹟 2 的權重為 3

第 3 行: 1 3 5 代表 古蹟 1 到古蹟 3 的權重為 5

第 4 行: 1 4 1 代表 古蹟 1 到古蹟 4 的權重為 1

第 5 行: 2 3 2 代表 古蹟 2 到古蹟 3 的權重為 2

第 6 行: 2 4 6 代表 古蹟 2 到古蹟 4 的權重為 6

第 7 行: 3 4 4 代表 古蹟 3 到古蹟 4 的權重為 4

範例輸出 1,

24 代表 把每條街道都走過至少一次再回到起始古蹟所需花費的最短時間