

# 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, 以及常用英文符號。

## 資料範圍

`[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, 無輸入值，簡單而快樂的輸出 Hello World!

# 問題 1 – 補習班 (Cram School)

---

(10 分)

## 問題敘述

APCS Camp 是全台灣最大的 APCS 補習班，每年吸引數以萬計（實際上大概是百萬）的學生報名參加。作為一個補習班，把學生的榜單貼在教室門口是必要的。但是近年大家越來越重視隱私，學生們不想自己的名字、考試年份等重要資訊被洩漏。APCS Camp 只好把年份與名字隱藏，只公佈學生當次考試在台灣的名次。今天高中生大倍倍正在考慮要不要報名 APCS Camp，於是他想了解這個補習班最少有多久的歷史。無從下手的他，決定利用表單上的名次，推估 APCS Camp 最少有幾年的歷史。身為大倍倍好朋友的你，可以幫他解決這個問題嗎？

## 輸入格式

第一行有一個整數  $N$  代表榜單長度。

第二行有  $N$  個整數  $A_i$ ，代表每個學生在其中一次比賽的台灣排名。

## 輸出格式

輸出一個整數  $Y$  代表 APCS Camp 至少有幾年的歷史。

## 資料範圍

$$1 \leq N \leq 10^6$$

$$1 \leq A_i \leq 10^9$$

## 輸入範例 1

4

1 1 1 1

## 輸出範例 1

4

## 輸入範例 2

4  
1 2 3 4

## 輸出範例 2

1

## 輸入範例 3

3  
1 1 2

## 輸出範例 3

2

## 範例說明

範例 1 有四個第一名，一定是在不同年份，所以至少有四年歷史。

範例 2 的四個名次有可能都是同一年，所以最少可能只有一年歷史。

範例 3 的兩個第一名一定在不同年，所以最少有兩年歷史。

## 問題 2 – 大波的函數 (Haskell)

---

(15 分)

### 問題敘述

成功通過程式挑戰營的大波，為了專題研究正在努力學習一個新的程式語言 Haskell。Haskell 是一個函數式程式語言，也就是程式裡所有的東西都是一個函式。但第一次接觸到這些的大波，卻被這一堆的函式給弄得眼花撩亂，讓他連自己寫出的程式能不能編譯都不知道。

Haskell 是一個很神奇的程式語言，編譯器會在編譯的時候自動幫你加上括號與逗號。因此如果寫出  $f\ x\ y\ z$ ，而且  $f$  需要三個引數，則編譯器會自動把它變成  $f(x, y, z)$ 。又例如若  $f$  需要兩個引數， $g$  需要一個引數，編譯器就會把  $f\ g\ x\ y$  變成  $f(g(x), y)$ 。

現在，大波定義了  $N$  個函式，第  $i$  個需要  $a_i$  個引數（如果  $a_i = 0$  那他是一個常數）。這時大波寫下一個用這些函式表示的程式碼，請你幫幫他判斷這個程式碼能不能通過編譯吧！

也就是，編譯器是否有辦法把程式碼中的空白換成適當的括號與逗號，使得得到的表達式裡都是已定義的函式，而且他的引數個數符合他的定義。

### 輸入格式

輸入的第一行包含一個正整數  $N$ ，表示大波擁有的函式個數。

接著有  $N$  行，第  $i$  行包含一個字串  $s_i$  與一個數字  $a_i$ ，兩者間以空白分隔。

這表示大波的第  $i$  個函式需要  $a_i$  個引數，而且名稱為  $s_i$ 。

最後一行是大波想要編譯的程式碼  $S$ 。

## 輸出格式

若該程式碼可以編譯，請輸出 Accepted。  
否則請輸出 Compilation Error。

## 資料範圍

$$1 \leq N \leq 10^5$$

$$|s_i| \leq 10$$

$$0 \leq a_i \leq N$$

$$|S| \leq 2 \times 10^6$$

$s_i$  僅包含大小寫英文字母、阿拉伯數字。

$s_i$  兩兩相異。

$S$  只包含大小寫英文字母、阿拉伯數字與空白。

$S$  不包含連續兩個空白， $S$  的最後一個字元也不會是空白。

## 輸入範例 1

```
2
x 0
y 0
x y
```

## 輸出範例 1

```
Compilation Error
```

## 輸入範例 2

```
2
function 1
argument 0
function argument
```

## 輸出範例 2

```
Accepted
```

### 輸入範例 3

```
2
getVal 1
object1 0
getVal object2
```

### 輸出範例 3

```
Compilation Error
```

### 輸入範例 4

```
4
f 3
x 0
y 0
z 0
f x y z
```

### 輸出範例 4

```
Accepted
```

### 輸入範例 5

```
5
f 2
g 1
x 0
y 0
z 0
f g g x g y
```

### 輸出範例 5

```
Accepted
```

### 輸入範例 6

```
5
Matsuri 2
Fubuki 1
Rushia 2
```

Marine 0

Nakiri 0

Matsuri Fubuki Nakiri Rushia Marine

## 輸出範例 6

Compilation Error

## 範例說明

範例輸入 1, 編譯器只能把它變成  $x(y)$ , 但是  $x$  卻不是函數。

範例輸入 2, 編譯器可以把它變成  $function(argument)$ 。

範例輸入 3, `object2` 未定義。

範例輸入 4, 編譯器可以把它變成  $f(x, y, z)$ 。

範例輸入 5, 編譯器可以把它變成  $f(g(g(x)), g(y))$ 。

## 問題 3 – 粉雪 (Powder snow)

---

(15 分)

### 問題敘述

在這細雪紛飛的季節，你與她道別。

你收到了一個字串作為餞別禮。

這個字串的強度，等於 $S_i S_{i+1} = S_j S_{j+1}$  ( $1 \leq i < j \leq |S| - 1$ )的數量。

聊表思念，你要求出這個字串的強度。



### 輸入格式

S

一個僅由小寫英文字母組成的字串。

### 輸出格式

X

一個非負整數，代表字串 S 的強度。

### 資料範圍

$|S| \leq 10^6$

### 輸入範例 1

aabaab

### 輸出範例 1

2

### 輸入範例 2

youngturingprogram

### 輸出範例 2

1

### 輸入範例 3

konayuki

### 輸出範例 3

0

## 範例說明

範例 1 中， $S_1S_2 = S_4S_5 = aa$ 、 $S_2S_3 = S_5S_6 = ab$ ，因此強度為 2。

範例 2 中， $S_4S_5 = S_{10}S_{11} = ng$ ，因此強度為 1。

範例 3 中，沒有任何合法的配對，因此強度為 0。

# 問題 4 – 自走車(Car)

---

(15 分)

## 問題敘述

Lawrance 是臺灣大學電機系的學生，今年修了一門叫電資工程入門設計與實作的課，期末專案是要做一台可以走迷宮的自走車。在各組焦頭爛額的思考如何設計讓車能自己找到出口的程式時，神通廣大的 Lawrance 從助教學姐取得了機密的期末專案地圖。

地圖是一張  $N \times N$  的方格，每個格子都有可能會有障礙物，起點是  $(1,1)$  終點是  $(N,N)$ 。想要拿 A+ 的 Lawrance 想要先找到破解這個迷宮的最短路徑，並且使用 U、D、L、R 四種指令儲存最短路徑，輕鬆在期末專案取得 A+。

但是只顧著社交的 Lawrance 就算看著地圖也沒有辦法找出最短路徑，擅長迷宮的你，願意幫助他取得 A+ 嗎？

起點是  $(1,1)$  終點是  $(N,N)$  都不會有障礙物，而且保證至少一條路可以從起點走到終點。否則助教學姐就沒辦法幫大家打分數了！

## 輸入格式

第一行有一個正整數  $N$  表示地圖寬度

接下來  $N$  行每行有長度為  $N$  的字串，X 代表障礙物 O 代表可以通行。

## 輸出格式

輸出由 U、D、L、R 組成的最短路徑指令，U 對應  $(-1,0)$ ，D 對應  $(+1,0)$ ，L 對應  $(0,-1)$ ，R 對應  $(0,+1)$ 。如果有多條最短路徑，請輸出字典序最小的解答。

## 資料範圍

$$1 \leq N \leq 10^2$$

## 輸入範例 1

5  
OXXXX  
OXOOO  
OOOXO  
XXXXO  
OOXOO

## 輸出範例 1

DDRRURRDDD

## 輸入範例 2

5  
OOOOO  
OXXXO  
OXXXO  
OXXXO  
OOOOO

## 輸出範例 2

DDDDRRRR

## 輸入範例 3

5  
OOOOO  
XXXXO  
OOOOO  
OXXXX  
OOOOO

## 輸出範例 3

RRRRDDLLLLDDRRR

## 範例說明

第 1 筆和第 3 筆的範例測資都只有一條最短路徑，並且如同輸出所描述。  
第 2 筆則有兩條最短路徑： DDDDRRRR 與 RRRRDDD，而前者的字典序較小  
所以要輸出 DDDDRRRR。

# 問題 5 – kSum

---

(15 分)

## 問題敘述

最近在準備面試的 Gnodymmot 在知名刷題平台 LeetCode 上看到這麼一道題目：

給定一個陣列，和一個數字 Sum，選出其中的 K 個元素，使得這個 K 元組的和是 Sum。輸出所有滿足條件的 K 元組。

K 元組交換順序視為相同，所以像 (1, 2, 3) 和 (2, 1, 3) 是重複的答案，只要輸出一個即可。

Gnodymmot 在看到題目的時候，突然有另外一個想法，如果不考慮 K 元組的和是多少，直接統計總共有多少 K 元組，那該如何解呢？

## 輸入格式

第一行有兩個正整數  $N, K$  表示陣列長度和元組長度

第二行有  $N$  個整數  $A_1 \dots A_n$  表示測資的陣列。

## 輸出格式

如果不重複的 K 元組數量超過  $10^{12}$  輸出 OLE 代表輸出量太多了  
否則輸出一個整數 X 代表不重複 K 元組的數量。

## 資料範圍

$$1 \leq N \leq 10^5$$

$$1 \leq K \leq 10^3$$

$$1 \leq A_i \leq 10^9$$

## 輸入範例 1

```
5 3
1 1 2 3 3
```

## 輸出範例 1

```
5
```

## 輸入範例 2

```
3 2
1 2 3
```

## 輸出範例 2

```
3
```

## 範例說明

第 1 筆範例有  $(1, 1, 2)$ ,  $(1, 1, 3)$ ,  $(1, 2, 3)$ ,  $(1, 3, 3)$ ,  $(2, 3, 3)$  共五種不重複三元組

第 2 筆範例有  $(1, 2)$ ,  $(1, 3)$ ,  $(2, 3)$  三種不重複二元組

# 問題 6 – 納許均衡 (Nash Equilibrium)

---

(15 分)

## 問題敘述

在殘酷的經濟學戰場上，有兩個佼佼者正在戰鬥著：小  $E$  與小  $C$ 。今天，他們在玩一個遊戲，規則很簡單： $E$  將選擇 1 到  $N$  的一個數字，而  $C$  也將選擇 1 到  $M$  的一個數字。他們同時選擇，並且不知道對方會選什麼。若  $E$  和  $C$  分別選了  $i$  與  $j$  號，則  $E$  將拿到  $X_{ij}$  塊錢、 $C$  將拿到  $Y_{ij}$  塊錢。

假設已經知道對方將選  $x$  這個數字，則我選的一個數字  $y$  為一個**最佳反應**若且唯若我選  $y$  所得到的效益不會比任何另外一個選項差。舉例來說，對於  $C$  來說，如果對方選  $x$ ，則若  $y$  為一個**最佳反應**的話，則對於所有的  $y'$ ，皆會滿足

$$Y_{xy} \geq Y_{xy'}$$

請注意，**最佳反應**不一定唯一。

如果兩人所選的數字恰好分別為對方的**最佳反應**，則稱這個選項為一個**納許均衡**。你的工作就是，給定  $N, M$ ，和所有  $X_{ij}, Y_{ij}$  的值，計算出有多少個**納許均衡**。

請注意，一切的名詞、人名等皆為虛構，若有與真實生活中有所雷同者，純屬意外，請以題目的定義為主。

## 輸入格式

輸出的第一行將有兩個數字  $N$  與  $M$ 。接下來，有  $N$  行數字，每一行有  $2M$  個數字，第  $i$  行的第  $2j - 1$  個數字為  $X_{ij}$ ，而第  $2j$  個數字為  $Y_{ij}$  ( $1 \leq i \leq N, 1 \leq j \leq M$ )。

## 輸出格式

請輸出一個數字，代表答案。

## 資料範圍

- $1 \leq N, M \leq 10^3$
- $|X_{ij}|, |Y_{ij}| \leq 10^9$

### 輸入範例 1

```
3 3
1 1 1 1 1 1
1 1 1 1 1 1
1 1 1 1 1 1
```

### 輸出範例 1

```
9
```

### 輸入範例 2

```
2 2
-1 -1 -3 0
0 -3 -2 -2
```

### 輸出範例 2

```
1
```

### 輸入範例 3

```
2 2
1 2 0 3
0 3 1 2
```

## 輸出範例 3

0

## 範例說明

範例 1，任一選項都是等價的，怎麼選都是納許均衡，所以有 9 種 ( $i=1\sim 3$ ,  $j=1\sim 3$  皆可)。

範例 2，只有  $i = j = 2$  是納許均衡。舉例來說，倘若  $i = 1, j = 2$ ，則  $E$  將自己的選項從 1 改到 2 得到的回報會從 0 增加到 1。

範例 3，不論選項是什麼，都有一個人沒有選到最佳反應。

# 問題 7 – 裝水問題 (FillingWater)

---

(20 分)

## 問題描述

在一很長很長的，寬度為  $1\text{ m}$  的長方形容器中，每隔  $1\text{ m}$  有一個隔板，共  $N$  個隔板，但每個隔板高度不見得相同。

若我們只能保留兩個板子（其它板子抽掉），並在兩個板子中間倒水，請問最多能裝多少  $\text{m}^3$  的水？

## 輸入格式

本題輸入僅有一行（行尾沒有換行，以 EOF 結束），有  $N$  個以空白間隔的數字  $x$ ，代表每個隔板的高度（單位  $\text{m}$ ）

## 輸出格式

一個正整數，代表能裝的最大體積為多少  $\text{m}^3$

## 資料範圍

$$2 \leq N \leq 1,000,000$$

$$1 \leq x \leq 1,000,000$$

## 輸入範例 1

3 5 2 9 7

## 輸出範例 1

15

## 輸入範例 2

12 7 5 3 6 9 3 5 4 1

## 輸出範例 2

45

## 輸入範例 3

5 1 6 3 2 9 1

## 輸出範例 3

25

## 範例說明

範例 1，若保留 3, 5 兩個板子，兩板距離為 1，則裝水容量為  $3\text{m}^3$ 。若保留

3, 2 兩個板子，兩板距離為 2，則裝水容量為  $4\text{m}^3$ 。若保留 5, 7 兩個板

子，兩板距離為 3，則裝水容量為  $15\text{m}^3$ ，為最大體積，所以輸出 15。

# 問題 8 – 找出最多的字 (Find Max Char)

(20分)

## 前言

假設有個句子「沙發大財富」

0	1	2	3	4
沙	發	大	財	富

其中有三個詞，分別是「沙發」、「發大財」、「財富」。

在「沙發大財富」這個句子中，不重疊的詞的組合有很多組，例如，「沙發 大財富」「沙 發大財 富」「沙發 大 財富」... 等，但不能用沙發和發大財的組合，因為“發”這個字重疊了！

以上組合中「沙發 大 財富」用到了 4 個字，是所有組合中不重疊且最多字的組合。

## 問題敘述

在上述情境中，我們已經把句子預處理過，輸入的資料會告訴你句子中那些部份可以成為詞，以“沙發大財富”而言，詞的部分可以依位置表示為，

0 1 沙發，位置是 0 ~ 1

1 3 發大財，位置是 1 ~ 3

3 4 財富，位置是 3 ~ 4

假設，在一長度  $n$  個字的句子中，有  $m$  個不同的詞，分別座落於句子中不同的位置並可能重疊。請找出不重疊且最多字的組合是多少字？

## 輸入格式

第一行有一個正整數  $n$ ，代表句子的長度

第二行有一個正整數  $m$ ，代表有  $m$  個詞

第三行起的  $m$  行，每行有兩個正整數  $x, y$ ，以空白分隔，分別表示每個詞的開始位置和結尾位置

## 輸出格式

一個整數，代表不重疊且最多字的組合是幾個字

## 資料範圍

$1 \leq n \leq 10,000$

$1 \leq m \leq 10,000$

$0 \leq x \leq y < n$

## 輸入範例 1

5

2

0 2

3 4

## 輸出範例 1

5

## 輸入範例 2

5

2

0 2

2 4

## 輸出範例 2

3

## 輸入範例 3

5

3

0 1

3 4

1 3

## 輸出範例 3

4

## 範例解釋

範例 1，第一個詞的位置是 0~2 佔了 3 個字，第二個詞的位置是 3~4 佔了 2 個字，兩個詞彼此不重疊，共佔 5 個字。

範例 2，第一個詞的位置是 0~2 佔了 3 個字，第二個詞的位置是 2~4 佔了 3 個字，但兩個重疊故只能選其一，但兩個長度相同，佔 3 個字。

範例 3，（可以參考「沙發大財富」的例子）第一個詞跟第三個詞重疊到，第二個詞也跟第三個詞重疊到，若選擇了第一個跟第二個，長度為 4；若選擇第三個詞，長度為 3。故最大長度為 4。

# 問題 9 – 地牢鑽石 (Dungeon Diamonds)

---

(20 分)

## 問題敘述

有一個地牢，裡面有著  $N$  個房間。

這些房間之間以地道所連接著。

任意兩個房間之間，恰好只會有一條簡單路徑相通。

在其中一些房間之中，會有一些鑽石。

你可以派遣  $K$  個機器人去蒐集這些鑽石。

如果一個房間被經過兩次以上，機器人還是只能蒐集到一次鑽石。

每個機器人，可以從一號房間開始，沿著一條簡單路徑，蒐集路徑上的所有的鑽石。

請問你最多可以得到多少鑽石？

(“簡單路徑”為一條由不重複的邊所構成的路徑。)

## 輸入格式

第一行包含兩個整數， $N$   $K$ ，代表房間的數量、機器人的數量。

第二行包含著  $N$  個整數  $a_1, a_2, \dots, a_N$ ，代表 1 號房間到  $N$  號房間分別有多少鑽石。

接下來  $N - 1$  行，每行會有兩個介於 1 到  $N$  的整數  $u_i$   $v_i$ ，代表  $u_i$  號房間和  $v_i$  號房間之間有一條地道連接著。

## 輸出格式

請輸出一個整數，代表你最多可以得到多少鑽石。

## 資料範圍

$$1 \leq N \leq 10^5, \quad 0 \leq K \leq N, \quad 0 \leq a_i \leq 10^9$$

$$1 \leq u_i, v_i \leq N, \quad u_i \neq v_i$$

### 輸入範例 1

1 1  
7

### 輸出範例 1

7

### 輸入範例 2

3 1  
1 2 3  
1 2  
1 3

### 輸出範例 2

4

### 輸入範例 3

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

### 輸出範例 3

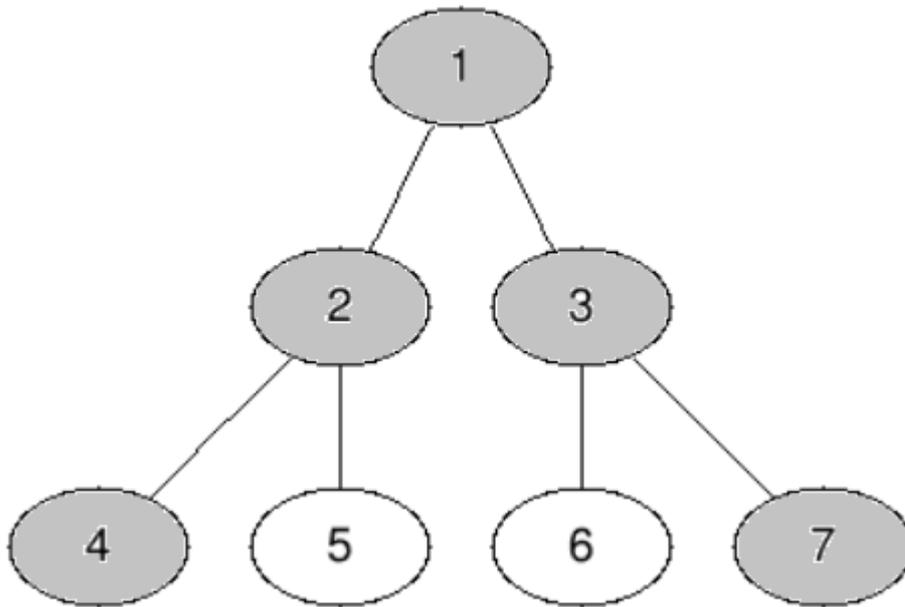
25

## 範例說明

範例 1：機器人的最好路徑為“1”，因此你能得到 7 個鑽石。

範例 2：機器人的最佳路徑為“1 3”，因此你能得到  $1+3=4$  個鑽石。

範例 3：最佳的情況下，機器人一的最佳路徑為“1 2 4”，機器人二的最佳路徑為“1 3 7”。所以你能得到的最多的鑽石數量是  $8 + 2 + 5 + 3 + 7 = 25$ 。



# 問題 10 – 優質烤肉 (Barbeque)

(8 分 / 20 分) 此題有部份給分

## 問題敘述

東東是今年 YTP 公司的中秋烤肉負責人，這裡指的並不只是他要負責規劃、採賣和宣傳，更重要的是他要自己一個人幫公司的所有員工烤肉！

因為油花的位置不同，肉在正面和反面烤的情況下味道會不同。在經過漫長的研究之後，東東發現無論翻面多少次，最後呈現的味道只跟出餐前最後一段時間燒烤的方向有關。具體來說，如果把在第  $i$  塊肉正面朝上燒烤的情況下結束料理給他人品嚐，能夠獲得  $a_i$  的滿足度，如果把在第  $i$  塊肉反面朝上燒烤的情況下結束料理給他人品嚐，能夠獲得  $b_i$  的滿足度，跟這塊肉在更之前是以哪一面燒烤沒有關係。

為了確認流程無誤，東東進行了一次演練。一開始所有的肉都是正面朝上，接著他會按照計劃將某些連續的肉翻面（原本是正面朝上就變成背面朝上，原本是背面朝上就變成正面朝上），並且計算假如他此時將某些肉拿給他人品嚐，總共會獲得多少的滿足度。但是肉的數量實在太多了，他根本忘記哪一塊肉現在是正面朝上還是反面朝上，你能幫幫他嗎？

## 輸入格式

輸入的第一行包含兩個正整數  $N, M$ ，表示東東同時要烤  $N$  塊肉，並且在過程中一個發生了  $M$  個事件。

第二行包括  $N$  個正整數  $a_i$ ，代表第  $i$  塊肉以正面朝上方式起鍋時能帶給客人的滿足度。

第三行包括  $N$  個正整數  $b_i$ ，代表第  $i$  塊肉以背面朝上方式起鍋時能帶給客人的滿足度。

接下來的  $M$  行，每行包括 3 個正整數  $t_i, l_i, r_i$ ，如果  $t_i = 0$ ，代表東東把第  $l_i$  到第  $r_i$  塊肉翻面；如果  $t_i = 1$ ，代表東東想詢問如果在此時將第  $l_i$  到第  $r_i$  塊肉起鍋，能帶給大家多少的滿足度。

注意到這題的輸入量十分龐大，建議使用 C++ 作答的同學請在程式碼開頭加上 `#include<cstdio>`，並使用 `scanf` 和 `printf` 作答。

## 輸出格式

對於每筆  $t_i = 1$  的詢問，輸出一個正整數  $k_i$ ，代表如果東東在此時將第  $l_i$  到第  $r_i$  塊肉起鍋，能帶給大家多少的滿足度。

## 資料範圍

- $1 \leq M \leq 10^6$
- $1 \leq N \leq 10^6$
- $1 \leq l_i \leq r_i \leq N$
- $1 \leq a_i, b_i \leq 10^9$

## 子任務

- 第一部份 (8 分) 對應的測試資料，滿足  $N, Q \leq 1000$
- 第二部份 (20 分) 對應的測試資料，無特殊限制

### 輸入範例 1

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

### 輸出範例 1

```
2
```

### 輸入範例 2

```
3 7
3 3 3
6 1 2
```

```
0 2 3
1 2 2
1 1 3
0 2 3
0 1 1
1 1 1
1 2 3
```

## 輸出範例 2

```
1
6
6
6
```

## 輸入範例 3

```
9 4
5 4 7 4 3 5 2 7 3
9 10 5 2 3 6 4 9 6
0 4 9
1 1 8
0 1 5
1 5 6
```

## 輸出範例 3

```
40
9
```

## 範例說明

在說明中以 O 代表正面朝上，X 代表背面朝上。

在範例 1 中，以下事件依序發生：

- 第 1 塊肉被翻面，狀態變為 X。
- 第 1 塊肉的滿足度為 2。

在範例 2 中，以下事件依序發生：

- 第 2 塊肉到第 3 塊肉被翻面，狀態變為 OXX。
- 第 2 塊肉的滿足度為 1。
- 第 1 塊肉到第 3 塊肉的滿足度總和為  $3 + 1 + 2 = 6$ 。
- 第 2 塊肉到第 3 塊肉被翻面，狀態變為 OOO。
- 第 1 塊肉被翻面，狀態變為 XOO。
- 第 1 塊肉的滿足度為 6。
- 第 2 塊肉到第 3 塊肉的滿足度總和為  $3 + 3 = 6$ 。

在範例 3 中，以下事件依序發生：

- 第 4 塊肉到第 9 塊肉被翻面，狀態變為 OOOXXXXX。
- 第 1 塊肉到第 8 塊肉的滿足度總和為  $5 + 4 + 7 + 2 + 3 + 6 + 4 + 9 = 40$ 。
- 第 1 塊肉到第 5 塊肉被翻面，狀態變為 XXXOOXXXX。
- 第 5 塊肉到第 6 塊肉的滿足度總和為  $3 + 6 = 9$ 。

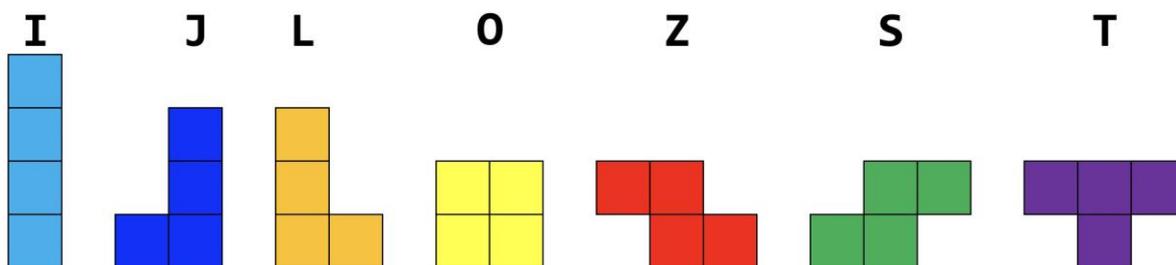
# 問題 11 – 俄羅斯方塊 (Tetris)

(16 分 / 25 分) 此題有部份給分

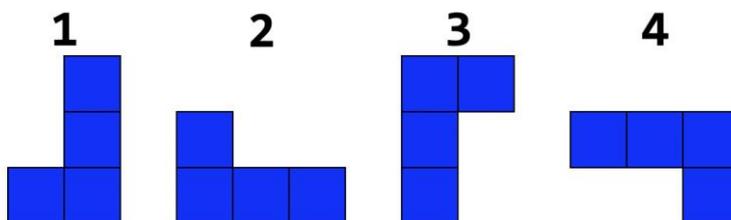
## 問題敘述

小宇是一位《俄羅斯方塊》愛好者，每天有空的時候就會玩個幾局消磨時間。這天，他發現他的鍵盤壞掉了，所以他除了開始遊戲以外什麼事都做不了，只能看著方塊自己落下。你能幫小宇計算一下他總共能消掉幾行嗎？

具體來說，這個遊戲會在一個左右寬  $W$  行 (Column)、上下高  $H$  列 (Row) 的區域裡進行。遊戲裡總共有 7 種不同的方塊，每個方塊會佔據恰好四個格子，如〈圖一〉所示。另外每種方塊還有四種不同的旋轉，其中「旋轉 1」即為〈圖一〉中所示的長相。而「旋轉 2~4」則分別是將前者 順時針 旋轉 90 度 後所得，如〈圖二〉所示。

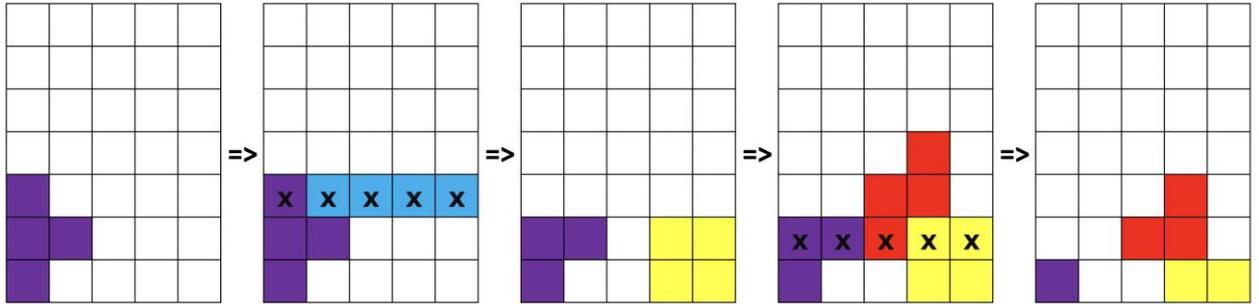


〈圖一〉七種方塊的長相，此為這些方塊的「標準旋轉」



〈圖二〉方塊 J 的四種旋轉

遊戲進行時，每次會有其中一個方塊的某種旋轉，從某幾行 (Column) 的正上方往下墜落，直到再往下會「超出遊戲盤面」或「與已經在盤面上的方塊重疊」為止。此時如果有某一列 (Row) 的所有格子都被方塊佔領，則這整列 (Row) 的格子都會被清除，而所有在這之上、已經被佔領的格子都會往下降恰好一格。〈圖三〉是一個可能的遊戲範例。



〈圖三〉一個可能的遊戲盤面。此與「輸入範例 1」相同。

## 輸入格式

輸入的第一行有三個正整數  $W, H, N$ ，分別代表遊戲盤面的寬度、高度、以及落下的方塊數量。

接著有  $N$  行，每一行有一個字元  $c_i$ ，以及兩個正整數  $p_i, r_i$ ，分別代表方塊的類型、方塊最左邊的格子所在的行 (Column) 以及方塊的旋轉。

同一行的所有東西皆以恰好一個空白分隔。

## 資料範圍

- $1 \leq W \leq 5000$
- $1 \leq H, N \leq 10^5$
- $c_i \in \{I, J, L, O, Z, S, T\}$
- $1 \leq p_i \leq W$ ，保證方塊不會超出盤面的左右界
- $1 \leq r_i \leq 4$

你如果答對所有滿足  $W \leq 100$  以及  $H, N \leq 1000$  的測試資料，你將可以獲得其中的 16 分。

## 輸出格式

如果在過程中，有某個方塊在落下時，沒辦法完全進入畫面內，請輸出：

TOP OUT  $x$

其中  $x$  代表此方塊是第  $x$  個方塊。請注意，TOP、OUT、以及  $x$  之間都恰好有一個空白。如果所有的  $N$  個方塊都可以順利落下，請輸出：

FINISH  $y$

其中  $y$  代表在過程中總共消除了  $y$  列 (Row)。同樣的，在 FINISH 跟  $y$  有恰好一個空白。

### 輸入範例 1

```
5 6 4
T 1 4
I 2 2
O 4 1
Z 3 2
```

### 輸出範例 1

```
FINISH 2
```

### 輸入範例 2

```
2 5 3
I 1 1
T 1 2
T 1 4
```

### 輸出範例 2

```
TOP OUT 2
```

### 輸入範例 3

```
5 10 11
I 1 1
J 2 2
S 4 2
T 4 2
Z 1 1
L 3 1
O 1 1
T 4 4
O 1 1
L 4 3
I 3 1
```

### 輸出範例 3

```
FINISH 8
```

### 範例說明

輸入範例 1 請見〈圖三〉。

輸入範例 2 中，第二個方塊 (T) 有剛好一格超出了遊戲區域。請注意，「超出」的判定是在清除疊滿的列 (Row) 之前。

輸入範例 3 總共消除了 8 行。