

編號 \_\_\_\_\_

## 2014 年國際資訊奧林匹亞競賽研習營初選 程式設計能力測驗

### 作答說明

1. 測驗時間為 180 分鐘。
2. 限用C/C++/PASCAL語言作答。
3. 共有 5 道試題，每題 20 分。
4. 測驗期間，若要飲水或上洗手間，需經監試人員同意始能離開，也應儘速返回試場。
5. 手機請關機，考試中若有學生之手機響鈴，該生該科以零分計算。
6. 測驗結束時請勿離開座位，並請同學將每題的原始程式及執行檔備份於官方隨身碟中。（主檔名為 P1, P2, ..., P5。）
7. 如需計算或作圖請利用計算紙或題目卷空白處。
8. 本次測驗缺考者，一律以 0 分計算，且不得以任何理由要求補考。

**請聽候指示再翻頁！**

教育部國民及學前教育署 委託

國立臺灣師範大學資訊工程系、資訊教育所 編製

中華民國一〇三年三月八日

## 測驗說明

此競賽使用國際資訊奧林匹亞競賽系統線上自動評分。選手以現場核發的帳號/密碼登入系統 (<http://tojjudge.csie.ntnu.edu.tw>)，將每道試題的解題程式上傳。

1. 針對每道試題，請將解題程式碼撰寫於一個檔案 (.c/.cpp/.pas) 中，並上傳至競賽系統。  
檔案大小不可超過 100KB，編譯所需記憶體不可超過 256 MB。
2. 所有試題一律使用標準輸入輸出，每一行輸出完畢後，均需列印一個換行字元。請嚴格依照題目指定格式輸出，多餘空白字元或換行字元均視為錯誤。
3. 解題順序可自由決定，不需依題號作答。
4. 每道試題可上傳多次，唯需間隔五分鐘。
5. 每道試題均設有執行時限，超過時間則視為解題失敗。
6. 每道試題的得分以該題所有上傳解題程式版本之最高得分計算。(注意：並非取各筆測資的最高得分總和。) 例如總共上傳兩次，第一次上傳程式的各筆得分為 4, 0, 0, 4, 4，第二次上傳程式的各筆得分為 0, 4, 4, 0, 0，則此題的得分為 12 分，非 20 分。
7. 上傳解題程式後，可在線上查閱得分或編譯/執行錯誤訊息。(可能需要等待一段時間。)
8. 若有任何舞弊(抄襲、連接非競賽系統之網路)或蓄意侵害競賽系統之行為，一經發現，將即刻失去參賽資格。
9. 對測驗問題有疑問，請利用線上提問系統 (Communication) 發問。
10. long long 型態的輸入輸出格式請使用 %lld。

## 第一題：神算絕配問題

執行時間: 1 秒

ABC 電視舉辦一場神算絕配競賽節目，規則如下：

1. 競賽者需兩人為一組報名，其中一人分至 A 區，另一人分至 B 區。
2. 各區分別給予一個很大的數字，A 區為  $m$ ，B 區為  $n$ 。
3. 令函數  $f(x) = 1^1 + 2^2 + 3^3 + \dots + x^x$ ，A 區需計算  $f(m)$  的值，B 區需計算  $f(n)$  的值，得到答案後，將答案傳至評量區，當評量區接收到 A 區與 B 區同一組人員都送出的答案時，評量區才開始檢測該組兩位成員答案是否正確。
4. 為節省評量時間，評量區將  $f(m)$  的個位數與  $f(n)$  的個位數相加，比對是否與兩區標準答案的個位數和相同。若答案不正確，將會退回該組答案，競賽者可重新計算，重送答案。最快算出正確答案的一組即得神算絕配競賽的冠軍。

請你幫評量區寫一個程式，計算  $f(m)$  的個位數與  $f(n)$  的個位數相加所得的值。

### 輸入說明

輸入兩個正整數  $m$  與  $n$  ( $1 \leq m, n \leq 10^{50}$ )， $m$  為給予 A 區的數字， $n$  為給予 B 區的數字，兩個數字以一個空白分開。

### 輸出說明

請輸出  $f(m)$  的個位數與  $f(n)$  的個位數相加所得的值。

範例一	範例二
輸入	輸入
2 2	3 1
輸出	輸出
10	3

### 計分方式

子任務	分數	額外輸入限制
1	4	$m, n \leq 100$
2	4	$m, n \leq 1000$
3	4	$m, n \leq 10^{30}$
4	4	$m, n \leq 10^{30}$
5	4	$m, n \leq 10^{50}$

## 第二題：最大總和問題

執行時間: 1 秒

給定一個  $N \times N$  的方形矩陣  $A$ ，以及三個正整數  $x$ 、 $y$  和  $S$ ，請找出由  $A$  矩陣中位於第  $x$  列第  $y$  行那一項開始（以下簡稱  $(x, y)$  項），移動  $S$  步（不可重覆經過同一位置）所經過的項之總和最大值。注意移動時只能上下或左右移動，不可走斜邊。例如一開始在  $(x, y)$  項，其下一步只能在以下四項中的一項： $(x+1, y)$ 、 $(x-1, y)$ 、 $(x, y+1)$  和  $(x, y-1)$ 。

以圖 1(a) 為例，若開始在  $(1, 1)$  項，其下一步只能在以下四項中的一項： $(2, 1)$ 、 $(0, 1)$ 、 $(1, 2)$  和  $(1, 0)$ 。若開始在  $(1, 0)$  項，其下一步只能在以下三項中的一項： $(2, 0)$ 、 $(0, 0)$  和  $(1, 1)$ （因為  $(1, -1)$  不合法）。

以圖 1(b) 為例，若  $x=1, y=1, S=3$ ，則最大的總和是 25（經過  $(1, 1), (2, 1), (2, 0)$  和  $(1, 0)$ ）。

$$A = \begin{bmatrix} (0,0) & (0,1) & (0,2) \\ (1,0) & (1,1) & (1,2) \\ (2,0) & (2,1) & (2,2) \end{bmatrix} \qquad A = \begin{bmatrix} 1 & 4 & 7 \\ 8 & 2 & 5 \\ 6 & 9 & 3 \end{bmatrix}$$

(a) (b)

圖 1 輸入矩陣：(a) 矩陣座標 (b) 一個矩陣例子。

### 輸入說明

第一行有兩個整數，代表陣列大小  $N$  ( $N \leq 10$ ) 和步數  $S$  ( $S \leq 10$ )。

接著有  $N$  行，代表  $N \times N$  矩陣。

最後一行有兩個整數值，代表起始點位置  $x, y$ ，其中  $0 \leq x < N, 0 \leq y < N$ 。

### 輸出說明

顯示出由起始點走  $S$  步所經過的數之最大總和。

<p><u>範例一</u></p> <p><u>輸入</u></p> <p>3 3 1 4 7 8 2 5 6 9 3 0 1</p> <p><u>輸出</u></p> <p>21</p>	<p><u>範例二</u></p> <p><u>輸入</u></p> <p>5 5 11 16 21 1 6 7 12 17 22 2 3 8 13 18 23 24 4 9 14 19 20 25 5 10 15 0 0</p> <p><u>輸出</u></p> <p>105</p>
--	---

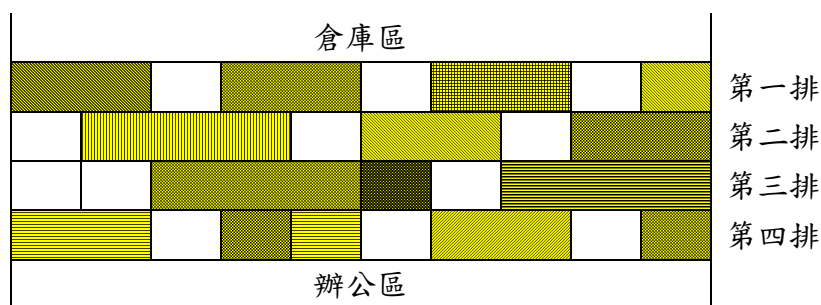
計分方式

子任務	分數	額外輸入限制
1	4	$N \leq 5, S \leq 3$
2	4	$N \leq 5, S \leq 3$
3	4	$N \leq 5$
4	4	無
5	4	無

### 第三題：機器搬動計畫

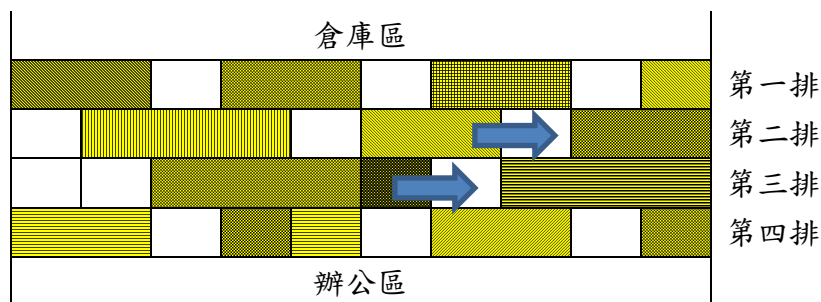
執行時間: 1 秒

工廠老闆是一個非常講究擺放整齊的人，所以他的工廠地上已事先畫上橫向和縱向的長方格，每個長方格的長與寬都是固定的。老闆要求機台的設計必須具有同樣的寬度，以對齊地上畫好長方格的寬度，而機器的長度必須為地上長方格之長度的整數倍。所有機台必須對好方格位置，橫向排列整齊，如下圖中的機器共排列成橫向四排，橫向的機器間可以緊鄰也可以有間隔，但間隔一定是長方格之長度的整數倍。



工廠擺放機台區域的兩邊各當作辦公區及倉庫區。有一天，工廠打算進行清倉的整理，必須將庫存產品從倉庫區用推車穿越機台區運送到辦公區，一輛推車的寬度剛好等於地上長方格之長度。為了讓推車能通過，必須將一些機台向左或向右移動，好空出一條通道讓推車通過。吹毛求疵的老闆還要求推車的運送必須垂直於機台的每一排直線通過，以免不小心撞到機台。

由於要移動機台必須先關機才能移動，而每關掉一座機台，不論關機多久都會造成固定的損失，因此廠長非常傷腦筋應該要關掉哪幾座機台，能夠損失最少，而讓運送庫存的推車順利通過。以下圖為例，將第二排由左算起第二座機台向右移一格，再將第三排由左算起第二座機台向右移一格，就可以在由左算起第 6 個長方格位置，空出一個走道讓推車順利通過。這樣需要關機的機台數為兩座。



請你寫一個程式，能夠由所給機台擺放的情況，算出所需關機搬動的最少機台數目，以及所空出走道由左算起是第幾個長方格位置。

**輸入說明**

第一行輸入兩個整數  $N, L$  並以一個空白區分， $N$  代表擺放機台的排數目， $L$  代表地上長方格從左到右的數目。其中  $1 \leq N \leq 20, 1 \leq L \leq 500$ 。

接下來的  $N$  行，每一行第一個位置有一個正整數  $I$ ，一個空白後有  $I$  個非負整數，皆以一個空白區分，表示在那一排的機台由左而右的依序擺設狀況。以  $S$  表示所讀入整數，若  $S$  為 0 表示空著一個長方格，若  $S$  大於 0 表示接下來有一個機台長度為  $S$  個長方格長度。請注意若機台間隔有多個長方格，則會以連續多個 0 表示，且每一排中都至少會有一個長方格間隔。

**輸出說明**

輸出第一行為所需關機搬動的最少機台數目，第二行為所空出走道由左算起是第幾個長方格位置。若同樣搬動最少機台數目有多個空出走道的答案，請輸出由左算起長方格位置編號最小的答案。

範例一	範例二
<u>輸入</u> 4 10 7 2 0 2 0 2 0 1 6 0 3 0 2 0 2 6 0 0 3 1 0 3 8 2 0 1 1 0 2 0 1  <u>輸出</u> 2 6	<u>輸入</u> 4 5 3 3 0 1 3 2 0 2 3 3 0 1 2 4 0  <u>輸出</u> 3 5

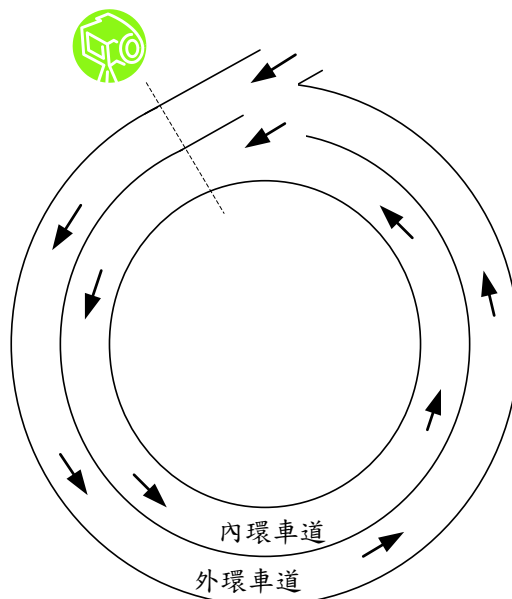
**計分方式**

子任務	分數	額外輸入限制
1	4	$L \leq 10$
2	4	$L \leq 20$
3	4	$L \leq 30$
4	4	$L \leq 30$
5	4	無

## 第四題：F-1 大賽車

執行時間：2 秒

F-1 星人最愛賽車，人人都是參賽者。比賽場地主要有外環車道與內環車道，外環車道和內環車道都只有一個入口，必須先進入外環車道之後才有機會進入內環車道。入口處有一台攝影機拍攝經過兩個車道的車輛，攝影機只能辨識經過的車輛，但無法辨識車輛是在哪一個車道，如下圖所示：



F-1 星球的賽車方式如下：外環車道只能容納  $m$  輛車，而內環車道只能容納  $n$  輛車。在內環車道開得不夠快的車輛（擠不進前  $n$  名）會被擠到外環車道第一名的位置；而在外環車道開得不夠快的車輛（擠不進前  $m$  名）則會被擠到場外去；場外的車輛要開得夠快才有辦法擠到外環車道的入口處，一旦從場外擠入外環車道，會馬上被瞬間移動到外環車道第一名的位置。在內環車道的車輛除非被擠到外環車道，否則會一直在內環車道繞圈圈，並且只要經過內環車道的入口處，就會瞬間移動到內環車道第一名的位置；進入外環車道的車輛若沒被擠出場外，跑完一圈經過入口處時即可擠進內環車道，並且也會被瞬間移動到內環車道第一名的位置。假設在沒有資訊的狀況下，車輛的排名不變。

運彩星球在 F-1 星球旁邊，運彩星人最喜歡以 F-1 星人賽車的結果來下注。下注的方式是選擇你認為比賽結束時，會留在內環車道的車輛，彩金以猜中的車輛數來發給，猜中的越多，彩金越多。給定攝影機拍攝到的比賽過程以及某張運彩星人的簽注單，請你寫一個程式來判斷這張簽注單的中獎情形。

我們以正整數來當作車輛的代號，舉例來說，假設外環車道能容納四輛車 ( $m = 4$ )，而內環車道只能容納兩輛車 ( $n = 2$ )，攝影機拍攝到在比賽過程中經過的車輛依序為：2, 7, 3, 7, 6, 4, 6, 2。由這個順序的前三個數字我們可以知道：



一開始外環車道領先車號變化情況為「2」→「7, 2」→「3, 7, 2」，內環車道此時還沒有車輛進入；接著 7 號車再次經過，表示此輛車超越 3 號車，進入內環車道；然後 6, 4 這兩輛車也擠入外環車道，此時外環車道的領先者依序為 4, 6, 3, 2；接著 6 號車再次出現，表示它超越了 4 號車，進入內環車道；接著 2 號車也擠進內環車道，因為內環車道只能容納兩輛車，所以 7 號車被擠到外環車道第一名的位置。比賽結束，內環車道的領先者依序是 2 號車與 6 號車，外環車道的領先群則依序是 7 號車、4 號車，以及 3 號車。假設簽注單上所下注的車號為 1, 2, 3, 4，則此張簽注單猜中 2 號車。

### 輸入說明

第一行輸入四個正整數  $m$ 、 $n$ 、 $c$ 、 $g$ ，並以一個空白區分， $m$  代表外環車道能容納的車輛數目 ( $m \leq 1024$ )， $n$  代表內環車道能容納的車輛數目 ( $n \leq 512$ )， $c$  代表攝影機拍到車輛的次數 ( $c \leq 30000$ )， $g$  代表簽注車輛數 ( $g \leq 20$ )。

接下來有  $c$  行，代表比賽過程中攝影機依序所拍到的車輛代號（代號均小於  $2 \times 10^9$ ），每一行記錄一輛車的車號。最後有  $g$  行，每一行代表下注的一部車輛代號。

### 輸出說明

輸出簽注單的中獎情形，若簽注的車輛中獎（比賽結束時在內環車道），則輸出 Y，否則輸出 N，以空白作為間隔（最後還有一個空白）。

<u>範例一</u>	<u>範例二</u>
<u>輸入</u>	<u>輸入</u>
4 2 8 4	2 1 8 4
2	50
7	918
3	5566
7	5566
6	918
4	246810
6	7749
2	8519
1	5566
2	6103
3	7749
4	8519
<u>輸出</u>	<u>輸出</u>
N Y N N	N N N N

計分方式

子任務	分數	額外輸入限制
1	2	$m \leq 4, n \leq 2$
2	4	$m \leq 50, n \leq 25$
3	4	$m \leq 100, n \leq 50, c \leq 1,000$
4	4	$c \leq 20,000$
5	6	(無)

## 第五題：作業問題

執行時間：3 秒

身在資工系的小傑，學期中常有寫不完的作業，每份作業都有繳交期限。認真的小傑對課業成績非常在意，總是盡力完成所有作業，可是到了期中他發現，就算不吃飯、不上課、不睡覺，也無法在期限內完成所有作業，於是他必須採取棄保策略，他已經預算好每個作業所要花的時間，請你算出他最多能在期限內交出幾份作業。

作業	花費時間	期限
作業一	2	2
作業二	1	5
作業三	2	4

以上述範例而言，若小明在時間 0~2 完成作業一，時間 2~3 完成作業二，則無法完成在期限內完成作業三，如圖 2(a) 所示。另一種較好的策略是時間 0~2 完成作業一，時間 2~4 完成作業三，接著完成作業二，則全部的作業都可以在期限內完成，如圖 2(b) 所示。

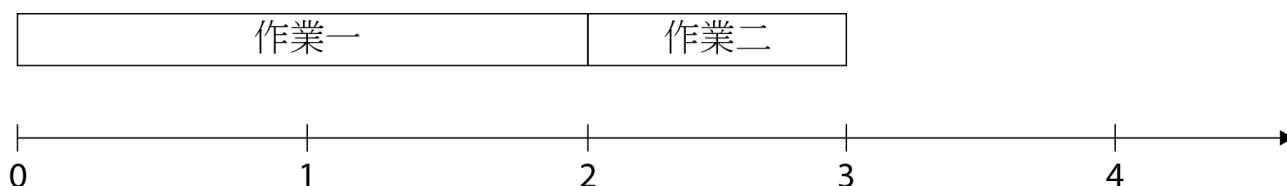


圖 2 (a)



圖 2 (b)

請你撰寫一個程式，讓小明輸入作業的花費時間和繳交期限，幫他計算出最多有多少份作業可以在期限內完成。

**輸入說明**

第 1 行輸入一個正整數  $n$ ，代表作業份數， $n \leq 1,000,000$ 。

第 2 ~  $n + 1$  行，每行兩個正整數，分別表示這個作業的繳交期限  $d$  ( $0 \leq d \leq 2 \cdot 10^9$ ) 和完成它所需的時間  $p$  ( $0 \leq p \leq 2 \cdot 10^9$ )，單位皆為一小時，可以假設小傑能從時間 0 開始不眠不休拼命地趕作業。

**輸出說明**

輸出一個整數代表最多能完成的作業份數。

<u>範例一</u>	<u>範例二</u>
<u>輸入</u>	<u>輸入</u>
3	5
2 2	5 2
3 1	4 2
4 2	3 1
<u>輸出</u>	<u>輸出</u>
2	2 2
	1 1
	<u>輸出</u>
	3

**計分方式**

子任務	分數	額外輸入限制
1	4	$n \leq 10$
2	4	$n \leq 100$
3	4	$n \leq 1,000$
4	4	(無)
5	4	(無)