九十四學年度高級中學資訊學科能力競賽決賽

上機程式設計題

作答注意事項：

一、　對考題有任何疑義，請於考試開始後 2 個小時之內填寫「問題單」，交付監考人員轉送命題委員提出問題，逾時不予回覆。

二、　第一題到第五題每題 16 分，第六題 20 分，共 100 分。

三、　可選擇指定解題語言中任何一種語言解題。

四、　最後繳交編譯後之執行檔限定在 Windows XP 的命令提示字元下執行。

五、　各題執行檔檔名請設定如下：
考生編號_題號.exe
例如：101_1.exe

六、　各題原始碼檔名請設定如下：
考生編號_題號.解題語言附屬檔名
例如：101_1.c

七、　各題輸入資料檔名如下：
in_題號.txt
例如：in_1.txt

八、　各題輸入方式以讀檔方式為之，請以目前工作目錄（Current Working Directory）下的檔案名稱為讀取路徑。

九、　各題輸出方式為標準輸出（螢幕)。

十、　考試結束後，將不再允許更動及重新編譯程式。

十一、所有發展的程式必須在 30 秒以內於試場內的電腦輸出結果，否則不予計分。

1. 城市道路連通網

問題敘述：

若我們在地圖中找出幾個城市之間的道路連通關係，並將其表示為一個無向圖形(undirected graph)；其中，圖形中的節點(node)表示城市，節點與節點間若有直線(edge)連接則代表兩城市間有一直接道路相通。例如：圖一為$\{c_1,c_2,c_3,c_4,c_5\}$五個城市的道路連通圖，圖二為用來記錄此圖一連通圖的相連矩陣(adjacency matrix)，矩陣的 1 代表兩節點有一道路相通，否則則以 0 表示。

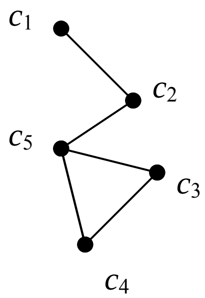


圖 一

	c_1	c_2	c_3	c_4	c_5
c_1	0	1	0	0	0
c_2	1	0	0	0	1
c_3	0	0	0	1	1
c_4	0	0	1	0	1
c_5	0	1	1	1	0

圖 二

假定在某國家的規劃中，每條道路的長度皆為 1 公里。請問如何得知由一城市 c_i 到另一個城市 c_j 最多不超過 N 公里(含 N 公里)的走法有幾種?

例如，圖一中由城市 c_3 到城市 c_5 最多不超過 3 公里的走法有 6 種。包括：1 公里 1 種 $\{c_3\rightarrow c_5\}$, 2 公里 1 種 $\{c_3\rightarrow c_4\rightarrow c_5\}$, 3 公里 4 種 $\{c_3\rightarrow c_4\rightarrow c_3\rightarrow c_5\}$,$\{c_3\rightarrow c_5\rightarrow c_2\rightarrow c_5\}$, $\{c_3\rightarrow c_5\rightarrow c_3\rightarrow c_5\}$,$\{c_3\rightarrow c_5\rightarrow c_4\rightarrow c_5\}$。

請注意，這些走法中可包含中途路經目的地的走法，如：上例中的$\{c_3\rightarrow c_5\rightarrow c_2\rightarrow c_5\}$等。

輸入說明：

第一行為一個正整數 m $(1<m\leq 32)$，代表城市的個數；即地圖上有$\{c_1, c_2, \ldots, c_m\}$共 m 個城市。

接下來的 m 行代表相連矩陣中每一行的內容。例如：相連矩陣中的第 i 行第 j 列的位置代表 c_i 與 c_j 兩城市之間有無直接道路相通，其中 0 表示兩城市無直接道路相通，1 表示兩城市間有一個 1 公里的道路連接。

接下來三行依序為三個正整數 i, j, N $(i\neq j,\ 1\leq i, j\leq m;\ 1\leq N\leq 50)$，表示由城市 c_i 到另一個城市 c_j 最多不超過 N 公里。

輸出說明：

輸出一整數，表示由城市 c_i 到另一個城市 c_j 最多不超過 N 公里(含 N 公里)的走法有幾種。**答案保證不會超過 $2^{31} - 1$**。

輸入範例 1：

```
5
01000
10001
00011
00101
01110
3
5
3
```

輸出範例 1：

```
6
```

輸入範例 2：

```
7
0101000
1011000
0100000
1100110
0001010
0001101
0000010
1
7
5
```

輸出範例 2：

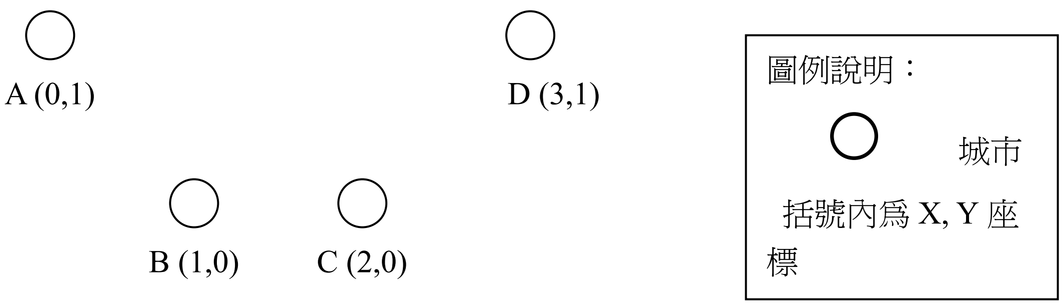
```
11
```

2. 衛星通訊中心

問題敘述：

在一個神奇的王國中，國王規定所有的道路都只能是東西向或是南北向，他認爲這樣才能有整齊的城市。連帶的，所有的水管，電線，電話線，天然氣管線等等也都必須按照這個規則，只能有東西向跟南北向，要轉彎通通都是直角轉彎。

這個國家有許多城市，這些城市的市長會議決議要大家集資建設一個衛星通訊中心，期望能夠用最新的通訊科技與世界接軌。爲了節省成本，衛星通訊中心必須建築在城市中。除了建設通訊中心的經費外，建設通訊中心與各個城市間的通信線路也需要耗費相當多的金錢。因爲通訊線路只能以東西向及南北向的方式架設，通訊中心到一個城市的通信線路長度，是彼此間東西向距離長度，加上南北向的距離長度。通往不同城市通信線路不能夠共用，也就是說雖然通過同樣一個地方，目的地不同仍然要各自計算成本。而通信線路的計價，與長度成正比，也就是說一條兩公里長的通信線路是一條一公里長線路的兩倍價錢。所以選擇建築通訊中心的城市地點就格外重要，如果能夠挑到一個好地點，那麼就能夠花最少的金錢去建設到每個城市的獨立通信線路。例如有四個城市 A、B、C、D，座標位置分布如下圖：



當通訊中心選擇建在 D 城市時：

通訊中心到城市 A 的通信線路長度爲 $|0-3|+|1-1|=3$，

到城市 B 爲 $|1-3|+|0-1|=3$，

到城市 C 爲 $|2-3|+|0-1|=2$，

到城市 D 爲 $|3-3|+|1-1|=0$。

此時通信線路的總長度爲 $3+3+2+0=8$。

在這例子中，選擇將通訊中心建立在城市 B 或 C 均可使得通訊線路的總長度爲最小 (最小值均爲 6)。所以將通訊中心建設在城市 B 或 C 時，能夠將建設費用降到最低。

請你寫一個程式來計算衛星通訊中心應該設置在哪裡，才能使得通訊線路的總長度爲最小。

輸入說明：

輸入的第一行爲一個整數 n ($1 \le n \le 1000$)，代表有多少個城市。接下來有 n 行，每行有一對整數 x_i 與 y_i ($0 \le x_i, y_i \le 500000$)，以空格分隔，代表第 i 個城市的平面座標是 (x_i, y_i)，即城市 i 坐落在離一個固定參考點以東 x_i 公里，以北 y_i 公里的位置。

輸出說明：

第一行輸出設置城市的座標，X 座標跟 Y 座標之間用一個空格隔開，答案如有多組最佳解時，任意輸出一組最佳解。

第二行輸出通信線路總長度。

輸入範例 1：

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

輸出範例 1：

```
0 0
5
```

輸入範例 2：

```
4
0 1
1 0
2 0
```

```
3 1
```

輸出範例 2：

```
1 0
6
```

3. 下界函數

問題敘述：

很久以前，兩個喜歡研究數字的數學家發現了一種有趣又實用的數列，稱爲 Davenport-Schinzel 數列。當一個數列滿足以下 3 個條件：

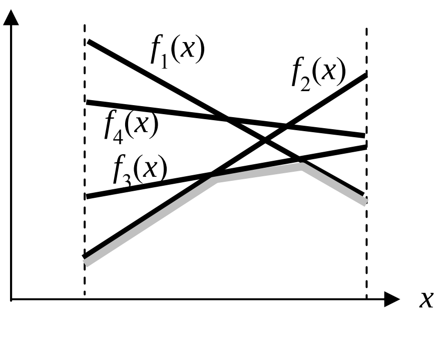
1. 數列裡的任何一個數字都會小於或等於一個給定的正整數 n
2. 任意兩個相鄰的數字都不會一樣
3. 數列裡找不到兩個數字會間隔地連續出現 s+2 次以上(包含 s+2 次)

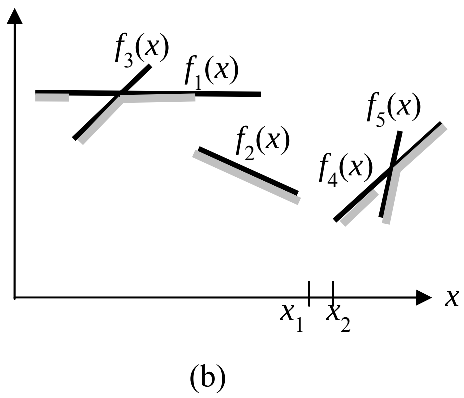
就稱爲 (n, s) Davenport-Schinzel 數列，簡稱 DS(n, s) 數列。例如， (1, 2, 3, 1, 3, 2, 1) 不是一個 DS(3, 3) 數列，當我們從數列中取出 1 和 2 這兩個數字會得到 (1, 2, 1, 2, 1)，這兩個數字間隔地連續出現了 5 次，所以它違反了第 3 個條件。而 (1, 2, 3, 1, 3, 2, 3) 就是一個 DS(3, 3) 數列。

Davenport-Schinzel 數列的性質有許多重要的應用，其中之一就是用來估算一組線性函數 (linear functions) 之下界函數 (lower envelope) 的線段數目。假設 $f_1, f_2, \ldots, f_n$ 是一組線性函數，每一個 f_i 都代表一條線段，它們的下界函數 L 的定義如下:

$$L(x) = \min\{f_i(x) \mid 1 \le i \le n\}$$

其中 L 的定義域 (domain) 是所有 f_i 之定義域的聯集。如果 $f_1, f_2, \ldots, f_n$ 的定義域都相同的話，L 會是一個凸包(convex)形狀的連續函數，如圖一(a)；否則 L 可能會是斷斷續續的線段，如圖一(b)。請注意，在圖一(b)中 $x \in (x_1, x_2)$並不包含在 L 的定義域中。





圖一、下界函數 L (灰色部分)

利用 Davenport-Schinzel 數列來估算下界函式的線段數，需要很複雜的運算。請你利用電腦程式來幫忙計算一組函數的下界函數包含有多少個線段。在這個問題裏，你可以假設任兩個 f_i 和 f_j 不會有重疊或相接成爲一條線段的情形，並且輸入資料中也不會有垂直線的存在。

輸入說明：

測試資料中圖一(a)和圖一(b)的情形都有可能會出現。測試檔的第一行爲一個正整數 n ($1 \le n \le 500$)，代表有 n 個函數。接下來的 n 行，每行會有四個介於 0 到 10000 的整數 x_1、y_1、x_2、y_2，代表一個兩端爲 (x_1, y_1) 和 (x_2, y_2) 的線段函數。

輸出說明：

輸出下界函數所包含的線段數，每組一行。如圖一(a) 輸出 3，圖一(b) 則輸出 7。

輸入範例 1：

```
4
0 100 100 20
0 20 100 80
0 40 100 50
0 70 100 60
```

輸出範例 1：

```
3
```

輸入範例 2：

```
5
0 30 50 30
40 20 70 15
10 20 30 40
80 10 120 40
90 0 110 40
```

輸出範例 **2**：

7

4. 滿漢全席

問題敘述：

滿漢全席是中國最豐盛的宴客菜餚，有許多種不同的材料透過滿族或是漢族的料理方式，呈現在數量繁多的菜色之中。由於菜色眾多而繁雜，只有極少數博學多聞技藝高超的廚師能夠做出滿漢全席，而能夠烹飪出經過專家認證滿漢全席，也是中國廚師最大的榮譽之一。

世界滿漢全席協會是由能夠料理滿漢全席的專家廚師們所組成，而他們之間還細分爲許多不同等級的廚師。爲了招收新進的廚師進入世界滿漢全席協會，將於近日舉辦滿漢全席大賽，協會派遣許多會員當作評審，爲了就是要在參賽的廚師之中，找到滿漢料理界的明日之星。

大會的規則是這樣的，將發給參賽的選手 n 種材料，而選手可以自由選擇將材料利用滿式或是漢式其中一種料理當成菜餚。而大會的評審制度是這樣的，有 m 位評審分別把關。每一位評審對於滿漢全席有各自獨特的見解，心裡頭都覺得在這些材料的限制下，有兩樣菜色是作爲滿漢全席是不能缺少的，如某評審認爲，如果沒有漢式東坡肉跟滿式的涮羊肉鍋，就不能算是滿漢全席。但避免過於有主見的審核，一個評審除非是兩樣心目中必備的菜色都沒有做出來的狀況下，否則是不會淘汰該參賽者的。換句話說，只要參賽者能在這兩種材料的做法中，其中一個符合評審的喜好即可通過該評審的審查。如材料有豬肉，羊肉和牛肉時，有四位評審的喜好如下表：

評審一	評審二	評審三	評審四
滿式牛肉	滿式豬肉	漢式牛肉	漢式牛肉
漢式豬肉	滿式羊肉	漢式豬肉	滿式羊肉

如參賽者甲做出滿式豬肉，滿式羊肉和滿式牛肉料理，他將無法滿足評審三的要求，無法通過評審。而參賽者乙做出漢式豬肉，滿式羊肉和滿式牛肉料理，就可以滿足所有評審的要求。

但大會後來發現，在這樣的制度下如果材料選擇跟派出的評審沒有特別安排好的話，所有的參賽者最多只能通過部分評審的審查而不是全部，所以可能會發生沒有人通過考驗的情形。如有四個評審喜好如下表時，則不論參賽者採取什麼樣的做法，都不可能通過所有評審的考驗：

評審一	評審二	評審三	評審四
滿式羊肉	滿式豬肉	漢式羊肉	漢式羊肉
漢式豬肉	滿式羊肉	漢式豬肉	滿式豬肉

所以大會希望有人能寫一個程式來判斷，所選出的 m 位評審，會不會發生

沒有人能通過考驗的窘境，以便協會挑選合適的評審團。

輸入說明：

測試檔案的第一行包含一個數字 K，代表測試檔案包含了 K 組資料。每一組測試資料的第一行包含兩個數字 n 跟 m，代表有 n 種材料，m 位評審。爲方便起見，材料捨棄中文名稱而給予編號，編號分別從 1 到 n。接下來的 m 行，每行都代表對應的評審所擁有的兩個喜好，每個喜好由一個英文字母跟一個數字代表，如 m1 代表這個評審喜歡第 1 個材料透過滿式料理做出來的菜，而 h2 代表這個評審喜歡第 2 個材料透過漢式料理做出來的菜。每個測試檔案不會有超過 50 組測試資料，而每筆測試資料中材料的種類數跟評審的個數均不超過 15。

輸出說明：

每筆測試資料輸出一行，如果不會發生沒有人能通過考驗的窘境，輸出 GOOD；否則輸出 BAD。

輸入範例 1：

```
2
3 4
m3 h1
m1 m2
h1 h3
h3 m2
2 4
h1 m2
m2 m1
h1 h2
m1 h2
```

輸出範例 1：

```
GOOD
BAD
```

輸入範例 **2**：

```
2
3 3
h1 m3
h3 m2
m3 m1
3 3
h1 m2
h2 m3
h3 m1
```

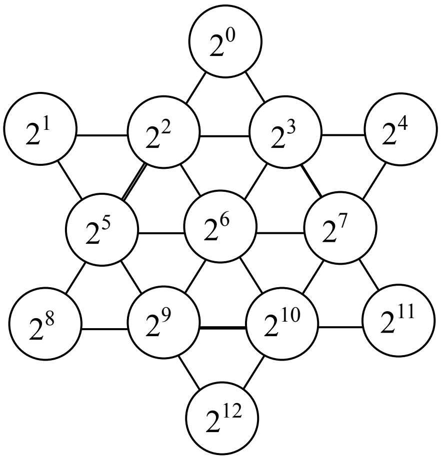
輸出範例 **2**：

```
GOOD
GOOD
```

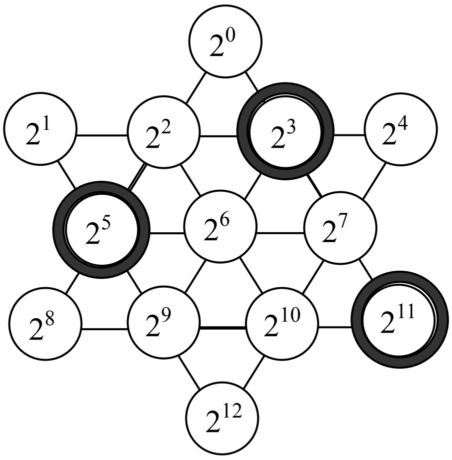
5. 六芒星棋遊戲：先還是後比較有利？

問題敘述：

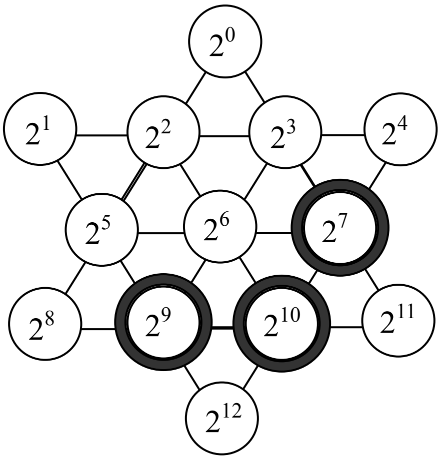
在一個稱爲『六芒星』的星球上，流行著一種『六芒星棋』的遊戲。這種遊戲是由兩個人來比賽，道具是一個棋盤（如圖一）和 13 個鐵圈。遊戲的起始狀態是由兩人互相協調或隨機決定，使用 m 個（$1 \le m \le 13$）鐵圈在棋盤上圈住 m 個數字。接下來雙方輪流移除棋盤上的鐵圈，每人每次只能移除 1 個鐵圈或 2 個相鄰的鐵圈（有直線直接相連的位置視爲相鄰），被迫移除最後一個鐵圈的人算輸。



圖一



圖二



圖三

其實如果仔細思考，可以證明不管起始狀態如何，遊戲中先移除鐵圈的人（先手）與其對手（後手）中必定恰有一人，只要他每次都用對自己最有利的方法來移除鐵圈，最後一定百分之百得勝。以起始狀態如圖二爲例，因爲所有鐵圈均不相鄰，故一次只能移除一個鐵圈，先手不管移除哪一個圈，後手只要再移除一個圈，就可以逼迫先手移除最後一個圈，因此後手有百分之百得勝的把握，我

們說成『圖二的起始狀態是後手有利』。若起始狀態如圖三就不同了，如果先手懂得先移除相鄰的兩個鐵圈（如 2^7 與 2^{10}），就可以迫使後手去移除最後一個圈，因此先手就有百分之百得勝的把握，我們說成『圖三的起始狀態是先手有利』。每一種起始狀態，我們用一個整數來表示它，這個整數就是在該起始狀態中所有被鐵圈圈住的數字總和。例如圖二的起始狀態可以用 $2^3+2^5+2^{11}=2088$ 表示，圖三則可用 $2^7+2^9+2^{10}=1664$ 來表示。

請你寫一個程式來判斷在不同的起始狀態下，到底是先手還是後手有利。

輸入說明：

第一行爲一個整數 n，$1 \le n \le 10$，代表接下來的 n 行中每一行有一個以整數表示的起始狀態。

輸出說明：

對每一個輸入的起始狀態，程式必需依順序輸出它們是先手有利（以 1 表示）還是後手有利（以 0 表示）。也就是說，輸出爲 n 個 0 或 1 的數字，數字間請用一個空白字元隔開。

輸入範例 **1**：

```
3
32
1024
1664
```

輸出範例 **1**：

```
0 0 1
```

輸入範例 **2**：

```
5
6
13
19
2088
```

```
2305
```

輸出範例 **2**：

```
1 1 0 0 0
```

6. 下棋問題

問題敘述：

eToy 發明了一個新的益智遊戲，該遊戲由 A 和 B 兩人輪流在一個 1,000,000 x 1,000,000 的方格棋盤上的格線交點下棋，格線交點的座標以（x, y）, $0 \le x$, $y \le$ 1,000,000 表示之，(0, 0)代表棋盤最左下角那點。

每一個棋子放置的位置不可以與任何其它棋子在同一 X 座標或 Y 座標上，棋盤上新增加一個棋子時，棋盤上的計數器會自動算出以目前棋盤上棋子所能夠圍成的「無障礙四方形」數。

「無障礙四方形」是指以任意兩個棋子所定義出的四方形內部不含其它棋子，每下一個棋子後所算出的「無障礙四方形」數即爲下該棋子的得分數。每位下棋者的總分即是該下棋者每個棋子的得分數總和。

請寫一個程式計算 A 和 B 兩位下棋者的累計總分。

輸入說明：

第一行輸入只有一個整數 n，代表此盤棋共下了 n ($1 \le n \le 5{,}000$)個棋子。接下來的 n 行，每一行有兩個整數，依序代表這 n 個棋子所放置的位置。

請注意，由於測試資料中確實包含 n=5000 的輸入，你的程式必須非常的有效率才會通過所有的測試資料。

輸出說明：

請輸出兩個整數，分別代表該盤棋兩位下棋者的累計得分數。先下棋者(A)的分數在前，後下棋者(B)的分數在後，中間用一個空白隔開。

輸入範例 1（不包含<-以及後面的字）：

4	<- 此盤棋共下了四步棋
2 3	<- 第一步棋 A 下在 (2, 3) *這一步棋 A 得 0 分
3 4	<- 第二步棋 B 下在 (3, 4) *這一步棋 B 得 1 分
1 2	<- 第三步棋 A 下在 (1, 2) *這一步棋 A 得 2 分
4 1	<- 第四步棋 B 下在 (4, 1) *這一步棋 B 得 5 分

輸出範例 1：

2 6	<- 這盤棋累計得分爲 A 棋者 2 分，B 棋者 6 分

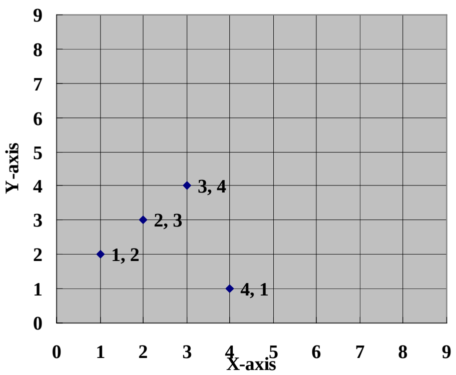
輸入範例 2（不包含<-以及後面的字）：

```
7       <- 此盤棋共下了七步棋
1 5     <- 第一步棋 A 下在 (1, 5) *這一步棋 A 得 0 分
2 7     <- 第二步棋 B 下在 (2, 7) *這一步棋 B 得 1 分
3 8     <- 第三步棋 A 下在 (3, 8) *這一步棋 A 得 2 分
5 1     <- 第四步棋 B 下在 (5, 1) *這一步棋 B 得 5 分
6 2     <- 第五步棋 A 下在 (6, 2) *這一步棋 A 得 9 分
7 3     <- 第六步棋 B 下在 (7, 3) *這一步棋 B 得 13 分
4 4     <- 第七步棋 A 下在 (4, 4) *這一步棋 A 得 10 分
```

輸出範例 2：

```
21 19   <- 這盤棋累計得分為 A 棋者 21 分，B 棋者 19 分
```

範例一



範例二