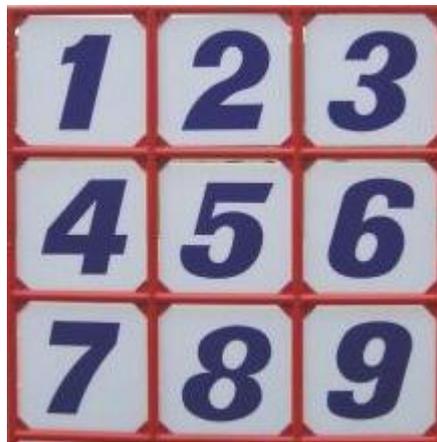


1. 棒球九宮格

問題敘述：

小明很喜歡棒球。某天晚上跟媽媽到家裡附近逛夜市，突然發現多了一個棒球九宮格的遊戲攤位(如圖一)。遊戲規則如下：一開始九宮格內會分別擺上一個木板，玩家站在距離九宮格板數公尺外，總共可以丟出九顆球。被命中的格子裡的木板會倒下；若球丟出範圍外，或是剛好丟在九宮格的框架上，則所有格子都會維持原樣。一旦將球丟出去，可用的球數就減少一，直到將九顆球丟完為止。

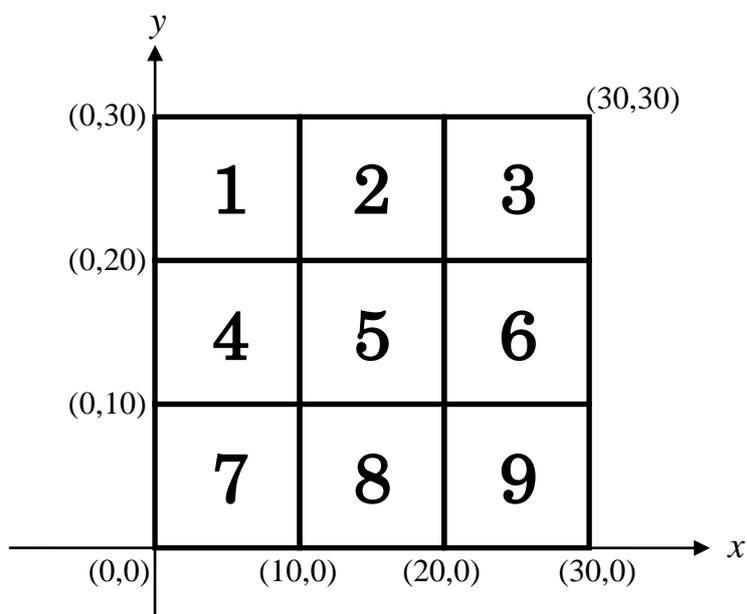


(圖一)

九顆球都丟完之後，夜市的老闆會根據倒下的格子來檢查連成幾條線，以決定獎品的好壞。依照老闆的規定，只要以下其中一組格子裡的木板同時倒下，則算連成一線： $\{1, 2, 3\}$ 、 $\{4, 5, 6\}$ 、 $\{7, 8, 9\}$ 、 $\{1, 4, 7\}$ 、 $\{2, 5, 8\}$ 、 $\{3, 6, 9\}$ 。舉例來說，若 2、3、5、7、8、9 這六個格子中的木板都倒下，則 2、5、8 連成一線，且 7、8、9 也連成一線，總共連成兩條線。(請注意，依規則 3、5、7 不算連成一線。)此外，老闆也會根據丟到的格子號碼，計算"加碼積分"，提供額外的獎賞，詳細規則如下：丟到 5 號得 2 分，丟到 $\{2, 4, 6, 8\}$ 中的任一號碼得 5 分，丟到 $\{1, 3, 7, 9\}$ 中的任一號碼得 8 分。例如若 2、5、8、9 這四格被丟中，則加碼積分為 20 分。請注意，重複丟到的號碼只算一次分數。

請寫一個程式來模擬玩棒球九宮格的遊戲。為了方便起見，我們將用二維座標來定義九宮格的位置，如圖二所示。每一格都是相同大小的正方形，7 號格左下角的座標為 $(0, 0)$ ，3 號格右上角座標為 $(30, 30)$ 。在模擬的過程中，會提供每一顆球丟在九宮格上的座標位置。為了簡化問題，若某顆球丟在九宮格上的座標位於某一格內，則判斷打到該格；如果座標位於某一格的邊上，則判斷為打到框架，

沒有任何格子被打到。舉例來說，假設一顆球丟到座標位置為(7,13)的地方，由於落在4號格之內，判定打到4號格；假設有另一顆球打在(10,2)(或(10,20))的座標位置，則因為落在框架上，沒有任何格子被打到。當然如果座標落在九宮格外的區域，也沒有任何格子被丟中。



(圖二)

輸入說明：

輸入總共有 9 行。每一行有兩個整數 x 和 y ($-20 \leq x, y \leq 50$ ， x 與 y 由一個空白隔開)，代表球的座標。

輸出說明：

請輸出丟出 9 顆球之後，根據夜市老闆的規則總共連成幾條線；另外請根據丟到的格子號碼，算出加碼積分。

輸入範例 1：

```
7 13
10 2
5 8
-9 19
11 35
```

3 23
18 0
0 18
40 22

輸出範例 1 :

1 21

輸入範例 2 :

12 22
15 11
3 20
3 22
33 27
28 28
16 5
22 -5
35 35

輸出範例 2 :

2 28

2. 線串式主題閱讀系統

問題敘述：

因為網路論壇的流行，吸引大量的內容討論文章。但傳統的閱讀方式是條列式、依照時間出現順序排列，一旦討論數量龐大，很難找到集中的討論主題。線串主題 (subject threading) 就是一種將相同主題 (subject) 集中，以方便閱讀的瀏覽方式。我們希望設計線串主題排列系統，挑選集中相同的主題文章，再依照最早出現主題的時間，依序排列。為了簡化系統，我們將文章的主題與發表時間摘要出來，成為一個集中的索引檔案，請根據此索引檔，輸出線串主題。

文章索引檔表示法說明：

索引檔內為每一篇文章建立三個欄位，分別為主題(Subject)、日期(Date)、與識別碼(ID)。主題表示為：

Subject: xxxxx

其中“Subject:”之後有一空白，“xxxxx”為主題內容。主題內容可含除換行之外的任何字元。主題內容前三個字元若為“Re:”或“RE:”則忽略不計。若內容有多組“Re:”或“RE:”刪除第一組即可，內容字串第一個非空白字元前之任何數量的空白(0x20)皆忽略不計；而主題字串最後一個非空白字元之後的所有空白字元也忽略不計，除此之外，比較兩主題是否相同時，所有字元必須完全相同，例如以下前三個主題皆視為相同主題，但最後一個則視為不同的主題：

Subject: 有關高中程式競賽

Subject: Re: 有關高中程式競賽

Subject: Re: 有關高中程式競賽

Subject: Re: Re: 有關高中程式競賽

日期欄位表示如右：M D H Y

其中，M 為 1 至 12 的整數，分別代表 1 月到 12 月。D 為 1 到 31 的整數，表示當月份第 1 日至 31 日。H 表示時，為 0 到 23 的整數。Y 代表年，為四位整數。數字間以一個空白相隔。例如以下日期表示 2008 年 11 月 3 日 12 時：

Date: 11 3 12 2008

“Date:”之後有一空白。

識別碼則以最多十位整數表示，每一篇文章之識別碼在索引檔中都是唯一。例如：

ID: 1234567

“ID:”之後有一空白。

輸入說明：

輸入為文章索引檔，每一篇文章有三個欄位，每一欄位單獨一行，依序為 Subject: Date: 與 ID:。Subject: 欄位長度最多 1024 個字元(byte)，不同的文章索引欄位以一空白行間隔。連續兩個空白行表示資料結束。每個索引檔最多含有 10,000 篇文章索引（每篇文章三個欄位），至少有兩個線串。

輸出說明：

輸出主題（已濾除 Re: RE: 與空白字元）相同的文章索引欄位為一主題線串，以線串中時間最早的文章為基準，依據時間先後次序輸出主題，倘若時間相同，則以在索引檔中出現的順序輸出。再將此線串中的文章依據出現時間輸出其對應 ID，**若時間相同，以在索引檔中出現的順序排列**，並以逗點隔開。線串之間以一空白行隔開。連續兩空白行表示結束，請輸出最早出現的 2 個線串資料即可，例如：

Subject: 有關高中程式競賽
ID: 1234657,123456,98765

Subject: 高中程式競賽地點
ID: 8888,9999,1000

輸入範例 1：

Subject: 經濟風暴
Date: 11 20 10 2008
ID: 1

Subject: 美國總統當選人
Date: 11 19 12 2008
ID: 12

Subject: Re: 經濟風暴
Date: 11 21 20 2008
ID: 34

Subject: Re: 美國總統當選人
Date: 11 22 13 2008

ID: 98

輸出範例 1：(只需輸出最早出現的 2 個線串)

Subject: 美國總統當選人

ID: 12,98

Subject: 經濟風暴

ID: 1,34

輸入範例 2：

Subject: 失業率升高

Date: 11 28 10 2008

ID: 1

Subject: 美國總統當選人

Date: 11 19 12 2008

ID: 12

Subject: 經濟風暴

Date: 11 21 20 2008

ID: 34

Subject: Re: 美國總統當選人

Date: 11 22 13 2008

ID: 98

輸出範例 2：(只需輸出最早出現的 2 個線串)

Subject: 美國總統當選人

ID: 12,98

Subject: 經濟風暴

ID: 34

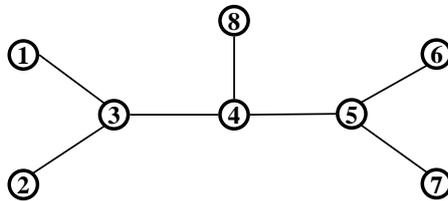
3. 找關鍵人物

問題敘述：

社會學家在研究一個社群時，常會用一個圖形 (graph) 來表示組成該社群人員之間的關係結構。一個標準的表示方式是：社群中每一個成員用圖形的一個頂點 (vertex) 來表示，而社群中的兩個人若有一特定關係則在這兩人相對的頂點間加一個邊 (edge) 來表示這個關係。而有了這樣的圖形表示法，社會學家就可以就這圖形的結構來進行一些研究。

有一種研究是要在社群中找關鍵人物，有關關鍵人物的認定有各式各樣的方式，其中有一種理論認為一個人若是一個關鍵人物那麼在社群中的很多聯繫都要透過這個人才能完成。因此基於這個觀點，有人提出一個假說：一個人若是一個關鍵人物，那麼他在社群關係圖形中相對的頂點就會有最多條最短路徑 (shortest path) 通過。

現有一個社會學家想要檢測這個理論的正確性，因此他希望你能寫個程式幫他找到社群中的可能關鍵人物。很湊巧的這社會學家找來的社群關係圖形都是樹狀的。我們都知道在樹狀圖 (tree) 中任何兩個頂點間只有唯一一條路徑，因此這路徑當然就是最短路徑。舉例來說，考慮以下樹狀圖：



這個圖形共有 8 個頂點，編號從 1 到 8。任何兩個頂點間都有一條路徑，因此這圖形共有 28 條路徑。頂點 1、2、6、7、8 是樹葉節點，因此沒有任何路徑通過。頂點 3 及頂點 5 分別都有 11 條路徑通過，而頂點 4 則有 15 條路徑通過，因此根據這個理論頂點 4 所代表的人物就是關鍵人物，而此點也稱為此圖的關鍵頂點。

輸入說明：

輸入內容的第一行只有一個數字 n ，代表輸入樹狀圖的頂點數。後面會接 $n-1$ 行數字，每一行有兩個數字以空白隔開，代表該圖一個邊的兩個頂點，頂點的編號從 1 到 n 。測試資料中 n 的可能最大值为 20000。

輸出說明：

輸出一行以空白隔開的兩個數字，第一個數字為所找到關鍵頂點的編號，若

好幾個頂點符合要求，請輸出編號最小的。第二個數字為通過該節點的路徑數。

輸入範例1：

8

1 3

3 4

5 4

6 5

5 7

3 2

4 8

輸出範例1：

4 15

輸入範例2：

6

1 5

5 4

2 4

5 6

3 4

輸出範例2：

4 7

4. 工作順序問題

H 教授的實驗室有 n 件工作，這些工作的編號依序為 $1, 2, 3, \dots, n$ 。而 H 教授的行政助理 L 小姐必須把它們一一做完，每次只能完成一件工作。但是 L 小姐很不喜歡按照順序工作（還好，此 n 件工作沒有先做後做的規定。），如果讓 L 小姐完成了編號 i 的工作，依她的行事風格，她鐵定不會接着完成編號為 $i+1$ 的工作 ($1 \leq i \leq n-1$)。聰明的你請幫 L 小姐計算看看，有多少種完成工作的順序可以進行？

輸入說明

輸入僅有一列，包含兩個正整數，分別是 n ($1 \leq n \leq 10,000,000$) 以及 m ($1 \leq m \leq 1,000,000,000$)，中間以一個空白隔開。

輸出說明

請輸出 L 小姐可能完成工作的順序方法數，再除以 m 的餘數。為了避免答案過大，只要輸出答案除以 m 的餘數就可以了。

範例一

輸入
3 1000

輸出
3

範例二

輸入
4 7

輸出
4

5. 蛋糕師傅的煩惱

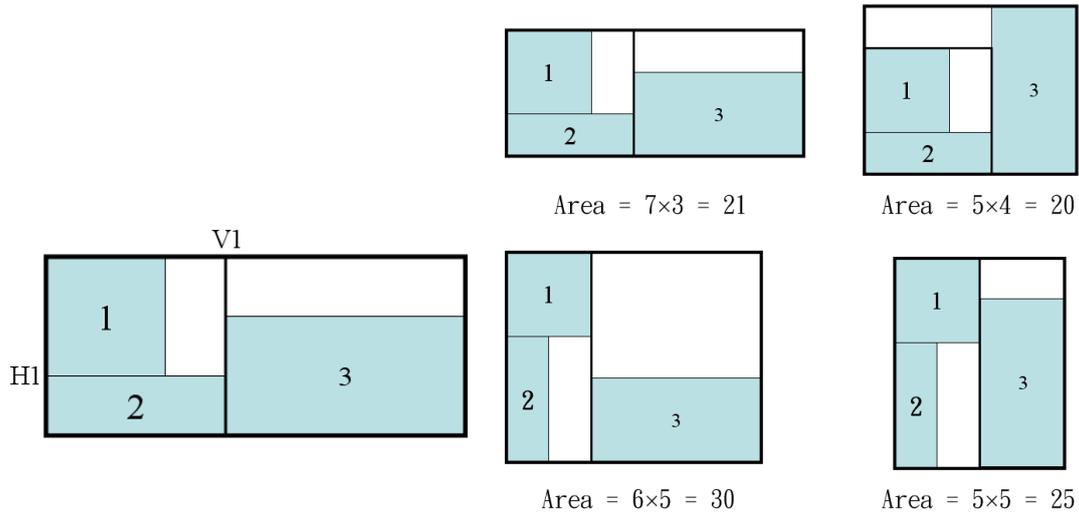
問題敘述：

有個蛋糕師傅作蛋糕的方式是先收集每位顧客訂購小蛋糕的大小，然後依照隨機順序將小蛋糕依序組合來決定最後要烘培的蛋糕大小，然後再一刀一刀的切出每個顧客所需的小蛋糕。

這裡有兩點要注意的是：(一)、師傅切蛋糕的習慣是每一刀都是以一條直線(水平線或垂直線)將蛋糕一切為二；(二)、是由於前述習慣，所以兩塊小蛋糕所組合出來的形狀一定要是矩形不可以是五邊以上的多邊形。例如有三位顧客訂購蛋糕的大小為 1 號 2×2 ，2 號 3×1 ，3 號 4×2 ，師傅依順序決定 1 號排在 2 號的上面， $(1,2,H)$ 表示 1 號排在 2 號上面所組成的新矩形，H 表示要將 1 號 2 號蛋糕分開必須切一刀水平線， $(1,2,H)$ 排在 3 號的左邊，圖一顯示三塊蛋糕的一種排法，值得注意的是，上下排列時，兩塊小蛋糕的左邊界一定要對齊，左右排列時，兩塊小蛋糕的下邊界一定要對齊，當大蛋糕烘培完成時，師傅垂直切第一刀後(如圖一 V1 所示)，3 號小蛋糕就獨立成型了，再水平切第二刀後(如圖一 H1 所示)，1 號和 2 號蛋糕也分別成型了，最後剩下的就是把多餘的部份切掉。

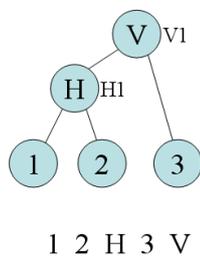
在此問題中假設上下或左右的關係已經決定，師傅要煩惱的是蛋糕要不要轉 90 度擺放，因為不同角度擺放會產生不同的面積，圖二顯示出前述例子因為小蛋糕有否轉 90 度擺放所產生四種不同面積，其中最小面積為 20。

師傅想到用樹狀圖來計算最小面積的小蛋糕擺法，以圖一為例，此擺法的二元樹狀圖如圖三所示，每個內部節點(internal node)旁邊的符號是相對於圖一裡的切蛋糕切法。針對樹狀圖的節點作後置順序拜訪所得到的節點次序如圖三所示。圖四左圖中最外層由最粗線寬所圍繞的矩型為蛋糕烘培時的面積，經過六刀(例如 H1,V1,V2,H2,H3,H4)的切割後，形成 7 塊各自獨立小蛋糕，切割的流程可用二元樹來表示，圖四右圖下方的表示式為對此二元樹作後置順序拜訪所得到的節點次序，注意此樹狀圖的中間節點只能是 H 或 V，代表蛋糕切法只能是垂直切或是水平切，而所有的小蛋糕都是落在樹葉節點(leaf node)。因此本問題的輸入為一後置順序拜訪所得到的樹狀圖節點次序與每個小蛋糕的大小尺寸，你必須在兩秒鐘之內幫蛋糕師傅算出所須最小的烘培蛋糕面積，所算出的面積不是最小或者是超過十秒鐘才算出答案者都要算失敗沒有通過測試。

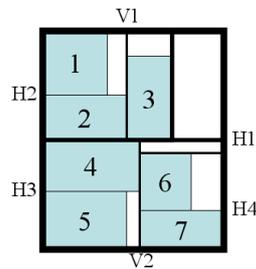


圖一

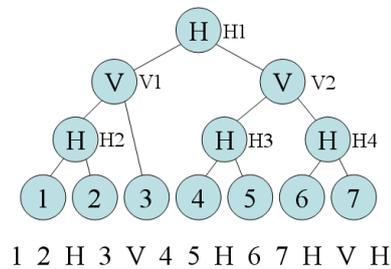
圖二



圖三



圖四



輸入說明：

輸入資料包含後置順序拜訪所得到的樹狀圖節點次序與每個小蛋糕的尺寸大小，第一行為描述樹狀圖節點次序，相鄰節點的內容以一個空白字元隔開，接著每一行描述一個小蛋糕的尺寸，語法為“ $n\ l\ w$ ”， n 表示第 n 號顧客所訂的蛋糕， l 和 w 分別代表此蛋糕的長與寬，每兩個值中間由空白字元隔開。其中 n ， l 和 w 為小於50的正整數。

輸出說明：

只須印出最小面積的值。

輸入範例 1：以圖一為例，其輸入檔案如下：

```
3 1 2 H V
1 2 2
2 3 1
3 4 2
```

輸出範例 1：輸入範例 1 的輸出結果如下：

輸入範例 2：以圖二為例，其輸入檔案如下：

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

輸出範例 2：輸入範例 2 的輸出結果如下：

```
221
```

請注意：輸出錯誤結果以及程式未在 10 秒之內完成的皆視為失敗。

6.耕者有其田

問題敘述：

在一個遙遠國度的國王有一塊形狀為凸多邊形的田地，長年以來由兩個辛勤的農夫幫忙整地施肥與耕種。這凸多邊形的任兩個頂點所連成的線段一定落於此多邊形之內。由於農夫的努力，此田地的收益帶給了國家許多的財富。為了感謝農夫對王國的貢獻，國王決定將此田地依其面積平均送給這兩個農夫，並把這個任務交由程式設計師來幫忙實現。

程式設計師在此田地的每個頂點依據 xy 平面實數直角座標系標示了座標。已知所有的頂點都落在 $x>0$ 的平面上。程式設計師想找出一條通過原點的直線 $y=ax$ (a 是一實數係數)，使這直線剛好等分割過這塊農地。

輸入說明：

輸入檔中所包含之測試資料的第一行是在測試資料裡所列出落在該凸多邊形邊上之點(包含頂點)的個數 ($N \leq 100$)，接下來則是這 N 個點的座標資料。每一行有兩個整數表示一個點的 x 、 y 座標 (x 、 y 的絕對值均小於一百萬)，且以一個以上(含一個)的空格分開。在輸入檔中的點座標資料並不一定照順時鐘方向列出。

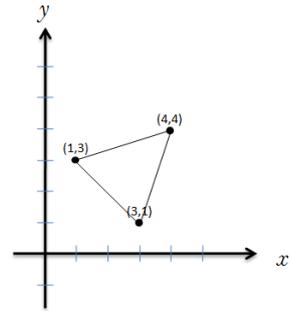
輸出說明：

針對所輸入的每組測試資料，每行輸出對應的實數係數 a 的值。精確度到小數點後第四位 (第五位以下四捨五入)。誤差在 ± 0.0001 之內都算正確。

注意：針對每組測資，產生答案的時間限制為十秒鐘。

輸入範例 一：

3
1 3
3 1
4 4

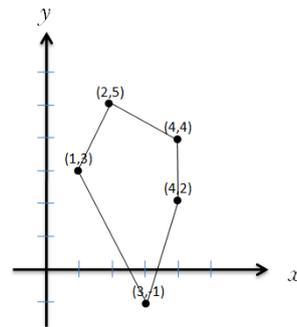


輸出範例 一：

1.0000

輸入範例 二：

5
4 4
4 2
2 5
3 -1
1 3



輸出範例 二：

0.9367