

A. 原始人排序

問題描述

布朗博士搭乘著時光機器回到了原始人出現的年代，他打算教導原始人現代的資訊技術。首先，布朗博士教原始人們數字的二進位表示法，他由小而大列下了幾個數字：

000, 001, 010, 011, 100, 101, 110, 111

剛列完，原始人的首領就糾正他，說下面的列法才是正確由小到大的順序：

000, 001, 010, 100, 011, 101, 110, 111

爭吵了半天，布朗博士才發現，原來原始人在計數時是用石頭在牆上刻劃一道道的痕跡，每道痕跡就算計數一次，因此在二進位的表示法下，1 的次數愈多，在原始人心中認定的數值愈大。

請你寫一支程式幫助布朗博士，將 n 個十進位表示的數字依原始人認定的順序（即二進位表示法中 1 的個數）由小而大排序；倘若兩數字的二進位表示中 1 的個數相同，則視為相同，依輸入時的順序輸出。

輸入格式

n
$a_1 \ a_2 \ \dots \ a_n$

- n ：輸入數字的個數。
- a_i ：第 i 個需要排序的數字，以十進位整數表示。

輸出格式

$b_1 \ b_2 \ \dots \ b_n$

- b_i ：排序完後，順序第 i 的數字，以十進位整數表示。

測資限制

- $1 \leq n \leq 10^3$ 。
- $1 \leq a_i \leq 1024 = 2^{10}$ 。 ($i \in \{1, 2, \dots, n\}$)
- 輸入給的 n 個數字皆相異，也就是對整數 $1 \leq i < j \leq n$ ，滿足 $a_i \neq a_j$ 。
- 上述所有變數均為整數。

範例測試

Sample Input	Sample Output
5 3 2 4 1 5	2 4 1 3 5
3 10 11 12	10 12 11

評分說明

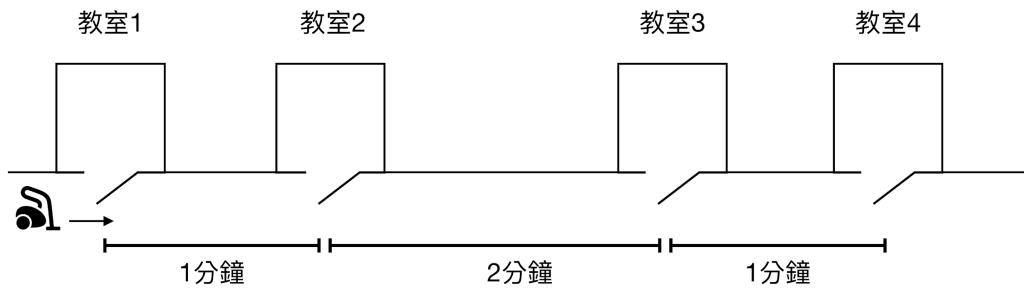
本題共有二組子任務，條件限制如下所示。每一子任務可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	40	$a_i \leq 7$ 。
2	60	無額外限制。

B. 掃地機器人

問題描述

大麥集團最近研發了新式的掃地機器人，可利用歷史資料（如特定區域過往的平均髒亂程度、地面材質是否容易清掃等）來增加掃除工作的效率。今有 n 間位於同一走廊的教室需要打掃，這些教室由左而右分別編號為 1 至 n ，每間教室的髒亂程度各有不同。如下圖為一 $n = 4$ 的例子：



掃地機器人因掃除力有限，因此對於同一間教室可能需要反覆多次的清掃才能將該教室掃除乾淨。在進行掃除時，剛開始因為灰塵較多比較容易吸除，但隨著掃除的時間增加，剩下的灰塵會愈來愈難被吸除。為了簡化問題，我們假設機器人在同一間教室必須打掃整數分鐘的時間，並且每分鐘能吸除的灰塵量會隨時間線性遞減，減至 0 則不會再減少。更詳細地說，對於教室 i 會有兩個參數 s_i 及 d_i ， s_i 為此教室掃除第一分鐘能吸到的灰塵量，而 d_i 則代表每隔一分鐘能吸到灰塵的遞減量。因此，掃地機器人在教室 i 清掃的第 x 分鐘能吸到的灰塵量為 $\max\{s_i - d_i \cdot (x - 1), 0\}$ 。

今天有 m 分鐘的時間供大麥掃地機器人進行打掃，在這 m 分鐘內機器人可以自由往返及打掃各教室。假設大麥掃地機器人一開始位於教室 1。給定每間教室第 1 分鐘能吸除的灰塵量、單位時間的遞減量以及相鄰兩間教室移動所需的時間，請幫大麥掃地機器人計算最多可吸除的灰塵量。

輸入格式

n	m		
t_1	t_2	\dots	t_{n-1}
s_1	s_2	\dots	s_n
d_1	d_2	\dots	d_n

- n 、 m ：依序為教室數及可打掃的時間（單位分鐘）。
- t_i ：掃地機器人從教室 i 移動到教室 $i + 1$ 的時間（單位分鐘）。
- s_i ：掃地機器人在教室 i 的第 1 分鐘能吸到的灰塵量。
- d_i ：掃地機器人在教室 i 每分鐘能吸到的灰塵遞減量。

輸出格式

answer

- *answer*：為一整數，代表 m 分鐘內掃地機器人能掃除最多的灰塵量。

測資限制

- $1 \leq n \leq 1000$ 。
- $1 \leq m \leq 10^9$ 。
- $0 \leq t_i \leq 10^9$ 。 ($i \in \{1, 2, \dots, n - 1\}$)
- $1 \leq s_i \leq 10^9$ 。 ($i \in \{1, 2, \dots, n\}$)
- $0 \leq d_i \leq 10^9$ 。 ($i \in \{1, 2, \dots, n\}$)
- 上述所有變數均為整數。

範例測試

Sample Input	Sample Output
4 9 0 0 0 3 1 6 3 1 0 3 2	21
4 9 1 1 6 3 1 6 3 1 0 3 2	17

範例解釋

第二筆範例中，機器人可以依照以下策略清掃到最多的灰塵：

- 第 1 到第 2 分鐘末：在教室 1 清掃，得到 $3 + 2 = 5$ 單位灰塵。
- 第 3 分鐘：從教室 1 移動到教室 2。
- 第 4 到第 6 分鐘末：在教室 2 清掃，得到 $1 + 1 + 1 = 3$ 單位灰塵。
- 第 7 分鐘：從教室 2 移動到教室 3。
- 第 8 到第 9 分鐘末：在教室 3 清掃，得到 $6 + 3 = 9$ 單位灰塵。
- 以上 9 分鐘，共蒐集到 $5 + 3 + 9 = 17$ 單位的灰塵。

評分說明

本題共有二組子任務，條件限制如下所示。每一子任務可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	30	$m \leq 1000$ 。
2	70	無額外限制。

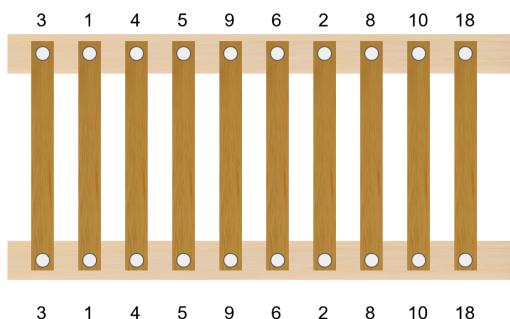
C. 粉刷護欄

問題描述

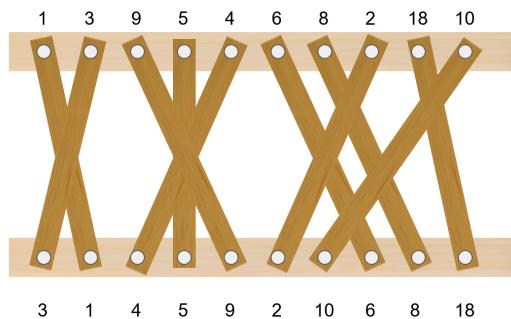
紅心國的黑桃國王為了促進國民身心健康，大力在國內修繕河濱自行車道，然而河道並不筆直，騎車時在彎道處不減速很容易衝出車道導致事故。為了減少事故，黑桃國王決定在彎道處搭建設護欄，並找了方塊設計公司與梅花人力公司來完成這項工程。

護欄由多片長條形的木板並排而成（如下圖一）。方塊設計公司將設計圖交給了梅花人力公司，並且要求梅花人力公司將木板標記兩兩相異的編號。不巧的是，梅花人力公司的工人們把藝術大學護欄的設計圖和車道護欄的設計圖搞混了，藝術大學護欄如下圖二。已知藝術大學護欄滿足：

1. 上下有 2 條水平的木樑，所有木板均釘在這兩條木樑上。
2. 每條木樑上等距離分成 n 個點，其中上面木樑第 i 個點釘著編號 a_i 的木板，下面木樑第 i 個點釘著編號 b_i 的木板。
3. 每根木板只會恰好釘在一個上木樑以及一個下木樑的點上。並且 n 根木板釘的點都互相相異。



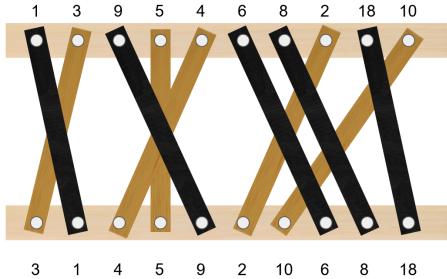
圖一



圖二

可想而知，這樣的成品是無法通過驗收的。方塊設計公司想了一個補救的辦法：從目前搭建成好的護欄中，找出若干個從側面看上去兩兩不相交的木板，並將選出的木板漆上深的顏色。如此一來，視覺上也達到了和原設計護欄類似的效果。

方塊設計公司希望兩兩不相交的木板愈多愈好，請你寫一支程式來找出需要上漆的木板。以圖二為例，可選出 5 片木板上漆，一個可能的上漆方式如圖三。



圖三

輸入格式

n
$a_1 \ a_2 \ \cdots \ a_n$
$b_1 \ b_2 \ \cdots \ b_n$

- n 為一整數代表每條木樑上點的數量。
- a_i 為一整數代表上面木樑第 i 個點釘著的木板的編號。
- b_i 為一整數代表下面木樑第 i 個點釘著的木板的編號。

輸出格式

$c_1 \ c_2 \ \cdots \ c_k$

- k 為一整數代表從側面看上去兩兩不相交的木板的最大數量。
- c_i 為一整數代表由左到右第 i 個需要上漆的木板編號。
- 若有多個可上漆的方式，則輸出 $c_1 c_2 \dots c_k$ 字典序最大的。字典序是先按照 c_1 以升序排列，如果 c_1 一樣，那麼比較 c_2 、 c_3 乃至 c_k 。

測資限制

- $1 \leq n \leq 2 \times 10^5$ 。
- a_1, a_2, \dots, a_n 為 n 個相異非負整數。
- b_1, b_2, \dots, b_n 和 a_1, a_2, \dots, a_n 為同一群數字，僅順序不同。
- $0 \leq a_i, b_i \leq 10^9$ 。 $(i \in \{1, 2, \dots, n\})$
- 上述所有變數均為整數。

範例測試

Sample Input	Sample Output
10 1 3 9 5 4 6 8 2 18 10 3 1 4 5 9 2 10 6 8 18	3 9 6 8 18

範例 1 說明：此範例對應圖二。最多可上漆的木板數量為 5，可能的上漆方式有 $(1, 4, 6, 8, 18)$, $(1, 5, 6, 8, 18)$, $(1, 9, 6, 8, 18)$, $(3, 4, 6, 8, 18)$, $(3, 5, 6, 8, 18)$, $(3, 9, 6, 8, 18)$ 六種，字典序最大者為 $(3, 9, 6, 8, 18)$ 。

評分說明

本題共有三組子任務，條件限制如下所示。每一子任務可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	22	$n \leq 10$, $\{a_1, a_2, \dots, a_n\} = \{b_1, b_2, \dots, b_n\} = \{0, 1, 2, \dots, n - 1\}$ 。
2	33	$n \leq 100$
3	45	無額外限制。

D. 乘車時間

問題描述

米德加市以 24 小時不間斷運轉的地鐵系統聞名。此地鐵系統由 n 座車站（編號為 $1, 2, \dots, n$ ）與 $n - 1$ 條地鐵線（分別稱為 1 號線至 $n - 1$ 號線）形成，且路網中任意 2 站皆可互相到達。列車運行模式如下：每條地鐵線皆為獨立運轉，且除了端點的 2 座車站，中途沒有任何停靠站。 i 號線連接車站 u_i 與 v_i ，行車時間固定為 w_i 分鐘，每天自 u_i 發的首班車時刻為零時 a_i 分，自 v_i 發的首班車時刻為零時 b_i 分，班距固定為 p_i 分鐘，其中 p_i 為不超過 6 的正整數，而 a_i 與 b_i 為小於 p_i 的非負整數。舉例來說，若 $u_i = 1, v_i = 2, w_i = 10, a_i = 2, b_i = 0, p_i = 5$ ，則每天 0:02、0:07、0:12、...、23:57 各有一班車從車站 1 開往車站 2，到站時刻分別為 0:12、0:17、0:22、...、0:07；回程的發車時刻則為 0:00、0:05、0:10、...、23:55，回到車站 1 的時刻分別為 0:10、0:15、0:20、...、0:05。

交通專家克勞德最近正在研究米德加市的地鐵系統，想知道在某些時間點從某些車站搭車到達另一些車站的所需時間。更精確地說，克勞德有 q 筆詢問，其中第 i 筆詢問可以用四個整數 h_i, m_i, s_i, t_i 表示，代表他想知道在 h_i 點 m_i 分，從車站 s_i 利用地鐵系統到達車站 t_i ，包含等車的所需時間。假定換車（同車站內換乘另一條地鐵線）需要恰好 1 分鐘，請寫一支程式幫助克勞德得到這些詢問的答案。

輸入格式

```

 $n \ q$ 
 $u_1 \ v_1 \ w_1 \ a_1 \ b_1 \ p_1$ 
 $u_2 \ v_2 \ w_2 \ a_2 \ b_2 \ p_2$ 
 $\vdots$ 
 $u_{n-1} \ v_{n-1} \ w_{n-1} \ a_{n-1} \ b_{n-1} \ p_{n-1}$ 
 $h_1 \ m_1 \ s_1 \ t_1$ 
 $h_2 \ m_2 \ s_2 \ t_2$ 
 $\vdots$ 
 $h_q \ m_q \ s_q \ t_q$ 

```

- n ：車站數量。
- q ：克勞德的詢問數。
- $u_i, v_i, w_i, a_i, b_i, p_i$ 表示 i 號線連接兩車站 u_i, v_i ，行車時間 w_i 分鐘，去程首班車為零時 a_i 分發，回程首班車為零時 b_i 分發，且班距固定為 p_i 分鐘。
- h_i, m_i, s_i, t_i 為克勞德的第 i 筆詢問，代表他問你 h_i 點 m_i 分從車站 s_i 到車站 t_i 的所需時間。

輸出格式

c_1
c_2
\vdots
c_q

- c_i 為一整數，代表第 i 筆詢問的答案，單位為分鐘。

測資限制

- $2 \leq n \leq 5 \times 10^4$ 。
- $1 \leq q \leq 2 \times 10^5$ 。
- $1 \leq u_i, v_i \leq n, 1 \leq w_i \leq 1000$ 。 ($i \in \{1, 2, \dots, n - 1\}$)
- $1 \leq p_i \leq 6, 0 \leq a_i, b_i \leq p_i - 1$ 。 ($i \in \{1, 2, \dots, n - 1\}$)
- $0 \leq h_i \leq 23, 0 \leq m_i \leq 59$ 。 ($i \in \{1, 2, \dots, q\}$)
- $1 \leq s_i, t_i \leq n$ ，且 $s_i \neq t_i$ 。 ($i \in \{1, 2, \dots, q\}$)
- 紿定的 $n - 1$ 條地鐵線恰可以連通 n 座車站。
- 輸入的數皆為整數。

範例測試

Sample Input	Sample Output
5 5	26
1 2 10 2 0 5	30
2 3 1 0 0 1	1
2 4 5 2 1 3	15
4 5 5 0 2 4	20
23 35 1 5	
23 35 5 1	
0 1 2 3	
17 30 3 5	
7 20 4 1	

範例解釋

第一筆詢問中，在 23:35 分從車站 1 出發至車站 5 的乘車過程如下：

1. 首先搭乘 23:37 分往車站 2 的列車，於 23:47 抵達。
2. 接著換乘 23:50 分往車站 4 的列車，於 23:55 抵達（注意需要花一分鐘轉車所以無法搭乘 23:47 發的車）。
3. 最後換乘 23:56 分往車站 5 的列車，於隔天 0:01 抵達目的地。

包含等車與換車的所需時間為 26 分。

第三筆詢問中，搭上 0:01 的列車，於 0:02 抵達目的地，需時 1 分鐘。注意出發時不需要 1 分鐘的轉乘時間即可直接乘車。

評分說明

本題共有四組子任務，條件限制如下所示。每一子任務可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	16	$n \leq 500$ 。
2	31	對於所有的 $i = 1, 2, \dots, n - 1$ ，皆有 $p_i = 1$ 。
3	37	對於所有的 $i = 1, 2, \dots, n - 1$ ，皆有 $u_i = i, v_i = i + 1$ 。
4	16	無額外限制。

E. 密室逃脫

問題描述

密室逃脫一般泛指一種特定的遊戲類型。在該類遊戲中，玩家通常被限定在一個近乎完全封閉或者對自身存在威脅的環境內，以第一視角探索週遭環境，不斷地尋找並利用身邊的物品做為工具，完成指定任務，並以最終逃離該區域為目的。這些指定的任務，常常是以解開特定謎題的方式呈現。

在實景遊戲的範疇裡，密室逃脫一般指一種單人或多人在特定場所裡進行的娛樂活動。參與這項活動的玩家，一般會被置身於在一個特定的場所，通過裝修與設計，營造逼真的場景，而後賦予玩家不同的身份、任務、及故事劇情，要求玩家在規定的時間內，通過尋找線索、團隊合作、層層解謎，最終完成任務脫離密室，整個過程一般進行 60 至 120 分鐘。因為該類遊戲是基於電子遊戲裡的密室逃脫的基礎發展而成，故又被稱為真人密室逃脫。（節錄自《維基百科》）

近年來，越來越多實境密室逃脫工作室成立，遊戲設計的難度也越來越高。而任務的設計必須不斷地推陳出新，否則在這個資訊科技發達的時代，任務關卡的破解方式很快就能在網路上找到。踢歐埃工作室是一間以太空探險為主題的實境密室逃脫工作室；他們打算設計一道會隨著參與人數的多寡而改變答案的遊戲關卡，其任務規則設計如下：

1. 依照參與人數 n 紿予一個 $n \times n$ 的棋盤方格，其中每個格子都被填入一個非負整數。
2. 用座標 (i, j) 來表示第 i 列第 j 行的格子，在關卡劇情中為一塊空域，飛空艇可以穿梭其中。
3. 令格子 (i, j) 被填入的非負整數為 $a_{i,j}$ ，代表 (i, j) 空域有質量為 $a_{i,j}$ 的小行星，影響飛航安全。
4. 玩家們有一座高能雷射武器，每次發射可以把一列或一行的所有小行星的質量都減去 1；一顆小行星唯有在質量降為 0 時才算被清除，而此時即便再被發射也不會減少質量（亦即質量不會變成負數）。
5. 由於這座高能雷射武器的耗能巨大，玩家們必須在發射總次數最低的情況下把棋盤上的所有小行星清除乾淨才算通關。

請你幫忙踢歐埃工作室設計一個程式，針對一給定的棋盤，給出一個發射總次數最低的策略。

輸入格式

```
n  
a1,1 a1,2 ... a1,n  
a2,1 a2,2 ... a2,n  
⋮  
an,1 an,2 ... an,n
```

- n 代表玩家數，亦即棋盤的列數與行數。
- $a_{i,j}$ 代表 (i, j) 空域的小行星質量。

輸出格式

```
m  
r1 r2 ... rn  
c1 c2 ... cn
```

- m 為一整數代表清除所有小行星的最低發射次數。
- $\{r_i\}_{i=1}^n$ 與 $\{c_i\}_{i=1}^n$ 為任意一個發射總次數最低的策略，其中
 - r_i 為一非負整數，代表朝著第 i 列發射的次數。
 - c_i 為一非負整數，代表朝著第 i 行發射的次數。
 - 輸出的 r_i, c_i 必須為 0 到 2×10^6 之間的非負整數。
- 若輸出的 m 為正確的最小次數但發射方案錯誤仍可得到部分分數，得分算法請參照下方評分說明一節。

測資限制

- $1 \leq n \leq 500$ 。
- $0 \leq a_{i,j} \leq 2 \times 10^6$ 。 $(i, j \in \{1, 2, \dots, n\})$
- 輸入的數皆為整數。

範例測試

Sample Input	Sample Output
3 1 0 1 0 1 0 0 1 0	2 1 0 0 0 1 0
4 8 5 37 3 4 4 46 4 77 12 23 82 100 59 81 98	233 3 12 79 95 5 2 34 3
10 65 93 72 20 76 24 52 27 74 88 28 80 11 1 29 30 7 14 39 55 88 71 62 51 41 65 62 66 53 4 60 49 11 71 21 78 24 31 10 87 61 38 2 6 64 43 25 14 95 64 54 80 97 64 56 70 73 27 84 52 74 8 78 98 6 56 71 91 17 86 80 80 85 80 21 57 88 68 66 28 36 31 15 88 87 100 99 73 42 4 94 7 39 56 75 1 79 76 81 11	885 46 33 40 45 60 67 68 55 67 46 48 47 30 30 30 33 33 30 35 42

評分說明

本題共有四組子任務，條件限制如下所示。

子任務	分數	額外輸入限制
1	20	$n \leq 20$, 且 $a_{i,j} \in \{0, 1\}$ 。
2	40	$n \leq 70$, 且 $a_{i,j} \in \{0, 1\}$ 。
3	35	$n \leq 70$ 。
4	5	無額外限制。

每一子任務可有一或多筆測試資料。在一子任務中，若所有測試資料的輸出均滿足以下條件，即可獲得滿分：

- 最小總發射次數 m 正確。
- 發射策略 $\{r_i\}_{i=1}^n$ 與 $\{c_i\}_{i=1}^n$ 滿足題目要求。

若子任務中有任一筆測試資料的輸出滿足以下任一條件，將收到 Wrong Answer 且無法在此子任務中獲得任何分數：

- 只輸出一行 m 但沒有輸出 r_i, c_i 。
- 其它輸出格式錯誤，例如 r_i, c_i 不是介於 0 到 2×10^6 之間的情形。
- 最小總發射次數 m 錯誤。

若子任務中所有測試資料的輸出，格式與最小總發射次數 m 皆正確，但發射策略不符合題目要求 ($\{r_i\}_{i=1}^n$ 與 $\{c_i\}_{i=1}^n$ 總和與 m 不相等也可以)，可在此子任務中獲得 40% 的分數。也就是說如果只會計算總和但不會計算解的方案時在 m 後面輸出兩行 n 個 0：

```

m
0 0 ... 0
0 0 ... 0

```

可以得到該子任務 40% 的分數。



第一題：挑選捷徑 (Shortcut)

問題敘述

捷運公司打算要在既有的捷運路網上，新造一條捷運路線。捷運公司希望這條新造的捷運路線能夠最大降低大眾的通勤時間，而通勤時間的估算是依以下原則計算：

- (1) 兩站之間的通勤時間定義為兩站在路網上的最短路線所經過的捷運站個數。如下圖左，從站 5 到站 4，最短的路線需要經過站 2、站 3、站 4，所以站 5 與站 4 之間的距離為 3。
- (2) 大眾的通勤時間定義為所有兩站之間的通勤時間加總。如下圖左，有 4 對捷運站之間的距離為 1、4 對距離為 2、2 對距離為 3，所以大眾通勤時間為 18。

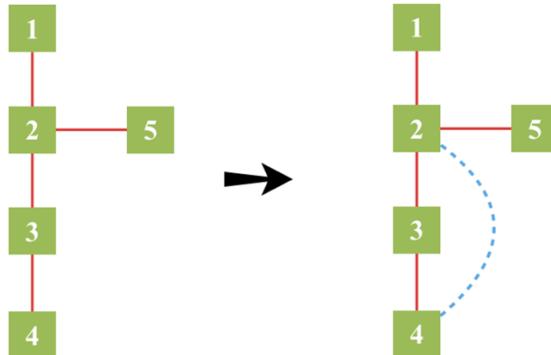


圖 1：捷運路網範例

若新路線選在連接站 2 和站 4，則大眾通勤時間從 18 下降至 15，是下降幅度最大的選擇，也剛好是上圖中唯一一條能下降 3 的新路線選擇。

輸入格式

輸入第一列有兩個正整數 n 以及 m ，兩數字間以一個空白隔開。其中 n 代表捷運車站數量， m 代表有 m 對捷運車站有直接相連，車站編號分別是從 1 至 n 的整數。

接下來的 m 列中每一列包含一對以空白隔開的整數，第 i 對整數 (u_i, v_i) 代表第 i 條連接線所連接的兩個車站編號。

輸出格式

輸出為兩整數，之間用一個空白隔開。第一個整數代表有多少條新路線的選擇，當它加入路網可以最大幅度下降大眾通勤時間。第二個整數代表大眾通勤時間在加入一條新路線後，最多可以下降多少大眾通勤時間。



測資限制

- $3 \leq n \leq 500$ 。
- $n - 1 \leq m \leq n(n - 1)/2 - 1$ 。
- $1 \leq u_i, v_i \leq n$, $u_i \neq v_i$ 。
- 任何一對捷運站只會被至多一條連接線直接連接。
- 整個捷運路網保證是連通的，任何兩個捷運站之間都能夠藉由一些連接線直接或間接相連。

輸入範例 1

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

輸出範例 1

```
1 3
```

輸入範例 2

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

輸出範例 2

```
5 1
```

評分說明

本題共有 4 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。



子任務	分數	額外輸入限制
1	9	$n \leq 40$ 。
2	22	$n \leq 150$ 。
3	18	$m = n - 1$ 。
4	51	無額外限制。



第二題：第 R 大的倍數 (Multiple)

問題敘述

小明剛在學校裡認識二進制，覺得很有趣。一個正整數的二進制表示是由一些 0 和 1 的位元組成的字串，且不含前導零 (leading zero)。他突然想到，在給定正整數 m 、 n 及 k 時，是否可以找到一些 k 的倍數，其二進制表示法剛好有 m 個 0 及 n 個 1。小明發現有些時候可以，如 $m = 2$ 、 $n = 3$ 和 $k = 5$ 時，則 11001 為 25 的二進制表示法。但有些時候則不可，如 $m = 1000$ 、 $n = 1$ 和 $k = 3$ 時，則唯一只有 1 個 1 的 1001 位二進制數字，並不是 3 的倍數。

「啊？原來二進制這麼簡單嗎？」小明不禁在心裡自我對話了起來。

「那如果給定一個順位 R ，有辦法計算出符合條件限制的數字中第 R 大的嗎？」看來是小明的左腦突發奇想，設計出了新的問題。

「這算是什麼問題 R 」小明輕輕吐槽了一下，然後發現其實自己不太會解這個突發奇想的問題。

現在小明拜託你幫忙，對於給定的正整數 m 、 n 、 k 和 R ，回答以上問題，並以其二進制表示法表示之。若答案不存在，則回答 **impossible**。

輸入格式

輸入包含四個以空白隔開的正整數 m, n, k, R 。

輸出格式

請輸出滿足題目條件以二進制表示之數值，若答案不存在則輸出 **impossible**。

測資限制

- $1 \leq m, n \leq 20000$ 。
- $1 \leq k \leq 500$ 。
- $1 \leq R \leq 10^9$ 。

輸入範例 1

2 3 5 1

輸出範例 1

11001



輸入範例 2

1000 1 3 1

輸出範例 2

impossible

輸入範例 3

3 2 6 3

輸出範例 3

impossible

評分說明

本題共有 5 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	13	$m, n, k \leq 10$ 。
2	5	$k = 3$ 。
3	39	$m, n, k \leq 500$ 。
4	24	$R = 1$ 。
5	19	無額外限制。



第三題：細菌 (Bacteria)

本題為互動題。

問題敘述

布朗博士研究水缸細菌，在一個大水缸中，這種細菌打從出生的第二秒開始，每一秒就會分裂產生一隻新一代細菌。在時間 $T = 0$ 秒時，布朗博士放進 A 個新生細菌，在時間 $T = 1$ 秒時，放進 B 個新生細菌，這些細菌在整秒時才會進行分裂。有趣的是，這種細菌除非人為將水缸環境徹底消毒，否則永遠不會死亡。你也可以假設布朗博士準備的水缸足夠大，可以容納得下任意正整數隻細菌。布朗博士架設了一個能自動觀察細菌數量的設備，記錄下每過一秒水缸中的細菌總數。

為了更了解這群細菌，布朗博士做了一些實驗，除了在 $T = 0$ 與 $T = 1$ 時，放入 A 與 B 個細菌之外，布朗博士設定了放置細菌的機器，能夠額外在每個時間 $T = k$ 整秒 ($k \geq 2$) 時，另外放入若干個新生細菌。布朗博士決定進行四場不同的實驗：

- 實驗 1：在每個時間 $T = k$ 秒時，額外放入 1 隻新生細菌。
- 實驗 2：在每個時間 $T = k$ 秒時，額外放入 k^2 隻新生細菌。
- 實驗 3：在每個時間 $T = k$ 秒時，額外放入 $k^2 \times 2^k$ 隻新生細菌。
- 實驗 4：在每個時間 $T = k$ 秒時，如果 k 是奇數則額外放入 2^k 隻新生細菌，否則不放入細菌。

某天，布朗博士在啟動架設好的實驗設備後，就出門運動去了。就在布朗運動回來以後，他卻意外地發現，自己寫的細菌紀錄程式有 bug：

- 不記得現在是第幾秒鐘了。
- 不記得現在進行的是哪一個實驗。
- 由於溢位的問題，自動觀察細菌數量的設備只記錄下目前看到的細菌數除以 $10^9 + 7$ 的餘數。
- 由於記憶體限制的問題，自動觀察細菌數量的設備只記錄最新 10^6 秒內的資料，更久以前的資料會因為被覆蓋而消失。

為了準確產出實驗成果，並順利在截稿前將其撰寫成論文投稿至期刊，你能夠幫助焦頭爛額的布朗博士，預測出將來的 t 秒以後，水缸中的細菌數量嗎？從自動觀察細菌數量的設備中，查閱歷史紀錄可能會導致設備延遲，進而造成實驗結果的不準確性。因此焦急的布朗博士希望執行你的程式時，查閱歷史紀錄的次數越少越好。



實作細節

你需要完成以下函式：

```
int predict(long long t);
```

- `predict(t)` 這個函式必須要能正確回答距離現在 t 秒以後水缸中的細菌數量，除以 $10^9 + 7$ 的餘數。
- 每一筆測試資料，其評分程式只會呼叫你實作的 `predict(t)` 函式至多 5 次。

你的程式可以呼叫以下函式：

```
int history(int k);
int get_experiment_type();
int get_a();
int get_b();
```

- `history(k)` 函式會回傳距離現在 k 秒以前水缸中的細菌數量，除以 $10^9 + 7$ 的餘數。輸入的 k 必須滿足 $0 \leq k \leq 10^6$ 。若詢問到了實驗開始之前的時間，該函式會回傳 0。
- `get_experiment_type()` 會回傳一個整數 $M \in \{1, 2, 3, 4\}$ ，代表實驗的編號。
- `get_a()` 會回傳實驗設定之 A 數值。
- `get_b()` 會回傳實驗設定之 B 數值。
- 請注意，`get_experiment_type`、`get_a`、`get_b` 為輔助函式，能夠獲得滿分的程式不需要使用這些函式。
- `get_experiment_type`、`get_a`、`get_b` 在每次的 `predict()` 執行中只能被呼叫至多 1 次。

如果不符合上述條件限制，你的程式會被判為 **Wrong Answer**；否則你的程式會被判斷為 **Accepted**。

測資限制

- $M \in \{1, 2, 3, 4\}$ 。
- $0 \leq A \leq 10^9$ 。
- $0 \leq B \leq 10^9$ 。
- 所有呼叫 `predict` 函式的實驗中，保證該次實驗已進行了至少 $T = 5$ 秒。



互動範例

考慮以下的測試資料：實驗編號 $M = 1$, $A = 1$, $B = 1$ 。

一個被評分程式判斷為 **Accepted** 的互動例子顯示如下：

評分程式端	參賽者端
呼叫 <code>predict(1)</code> 。	呼叫 <code>get_experiment_type()</code> 。
回傳 1。	呼叫 <code>history(5)</code> 。
回傳 1。	呼叫 <code>history(4)</code> 。
回傳 2。	回傳 33。

評分說明

對於每一次的 `predict` 函式執行中，假設你呼叫 `history` 的次數為 C_h 。此外，我們同時定義 C_M, C_A, C_B 這三個數值，其中 C_M 的值為 1 或 0，表示有無呼叫 `get_experiment_type`。 C_A 與 C_B 也相似，分別表示有無呼叫 `get_a` 與 `get_b`。得到的**分數比重** S 值定義如下：

$$S = \min \left(1.0, \sqrt{\frac{\log 6}{\log C_h}} \right) \times (1.0 - 0.5C_M - 0.1C_A - 0.1C_B)$$

本題共有 6 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，你的得分是該組所有測試資料之得分比重 S 中最低者，乘以該子任務的分數。

子任務	分數	額外輸入限制
1	4	總是進行實驗 1，並且傳入之參數 t 滿足 $1 \leq t \leq 10^6$ 。
2	5	總是進行實驗 2，並且傳入之參數 t 滿足 $1 \leq t \leq 10^6$ 。
3	6	總是進行實驗 3，並且傳入之參數 t 滿足 $1 \leq t \leq 10^6$ 。
4	7	總是進行實驗 4，並且傳入之參數 t 滿足 $1 \leq t \leq 10^6$ 。
5	23	保證實驗開始迄今不超過 1000 秒，參數 t 滿足 $1 \leq t \leq 10^{18}$ 。
6	55	實驗開始迄今秒數無上限，參數 t 滿足 $1 \leq t \leq 10^{18}$ 。



範例評分程式

範例評分程式以下列格式讀取輸入：

- 第 1 列： K ($K \leq 5$)，代表有幾場實驗。
- 第 2 至 $K + 1$ 列：每一列有五個以空白隔開的整數： M, A, B, now, t 。其中 now 代表了詢問當下實驗已進行的秒數、而 t 值則是用來傳入 `predict` 函式的引數。

請注意：使用自己上傳的測試資料進行測試時，沒有下列 MSG 描述的情形時你總會得到 Accepted。 如果你的程式被評為 **Accepted**，範例評分程式輸出 Accepted: $<output_1> <output_2> \dots$ ，其中 $output_i$ 對應到各場實驗的 `predict()` 回傳值。如果你的程式被評為 **Wrong Answer**，範例評分程式輸出 Wrong Answer: MSG，其中 MSG 格式與意義如下：

- invalid K: 實驗場數不合法。
- invalid experiment type: 實驗編號不合法。
- invalid A: 輸入之 A 值不合法。
- invalid B: 輸入之 B 值不合法。
- invalid now: 輸入之當前秒數不合法。
- invalid t: 呼叫 `predict` 傳入之引數不合法。
- too many history calls: 呼叫 `history` 超過 10^6 次。
- invalid k for history: 呼叫 `history` 傳入之引數不合法。
- too many get_experiment_type calls: 呼叫 `get_experiment_type` 超過 1 次。
- too many get_a calls: 呼叫 `get_a` 超過 1 次。
- too many get_b calls: 呼叫 `get_b` 超過 1 次。
- corrupted stub: 可能改到 `stub` 了，請重新下載 `stub.cpp`。



第四題：垂直射擊 (Shooting)

問題敘述

有 n 個目標物，每個目標物都是一個水平線段 $[s, t]$ 或者只含一點 ($s = t$ 的情形)，所有目標物的高度 (Y 值) 均大於 0 且互不相同。每一次射擊都是在 X 軸上的某個整數點位置往上垂直發射一砲彈，砲彈路線上第一個碰到的目標物（包含線段端點）就是此發射擊所射中的目標，砲彈射中目標後就與該目標一起消失，並不會穿透。射中每一個目標物即可獲得該目標物的獎分，現在依序發射了 m 發砲彈，請計算出獲得的獎分總和。請注意，每個目標物的高度不同，因此這些目標物都沒有重疊，因此每一發砲彈最多射中一個目標，如果沒有射中任何目標，則該發射擊的獎分為 0。

輸入說明

輸入第一列為 2 個正整數 n 與 m ，分別代表目標物的數量與射擊的砲彈數，接下來 n 列每一列包含一個目標物的資料，順序是由上到下，也就是 Y 值由大到小。每一列目標物的資料包含三個整數 s 、 t 與 w ，代表目標的線段區間是 $[s, t]$ 而獎分為 w 。測資的最後一列包含 m 個以空白隔開的非負整數 x_1, x_2, \dots, x_m ，依序代表每一發射擊的 X 座標。

輸出說明

輸出一個整數，代表最終得到的獎分總和。

測資限制

- $1 \leq n \leq 5 \times 10^5$ 。
- $1 \leq m \leq 5 \times 10^5$ 。
- 對於所有目標物的資料 (s, t, w) ，皆有 $0 \leq s \leq t \leq 10^9$ 而且 $0 \leq w \leq 1000$ 。
- 每一次射擊的 X 座標 x_i 均滿足 $0 \leq x_i \leq 10^9$ 。

輸入範例

```
4 3
0 1 1
3 3 2
5 7 3
0 4 4
3 1 4
```



輸出範例

5

範例說明

第一發砲彈 $x_1 = 3$ ，會射中 $[0, 4]$ ，獲得獎分 4 分；第二發砲彈 $x_2 = 1$ ，射中 $[0, 1]$ ，獲得獎分 1 分；第三發 $x_3 = 4$ ，沒射中任何目標。總共獎分為 5 分。

評分說明

本題共有 5 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	5	$n, m \leq 10^5$ ； X 座標範圍不超過 10000；對於所有目標物皆有 $s = t$ 。
2	7	$n, m \leq 10^5$ ； X 座標範圍不超過 10000；對於每個目標線段皆有 $t - s \leq 10$ 。
3	15	$n, m \leq 10^5$ ；砲彈射擊位置為由左至右，也就是 $x_1 \leq x_2 \leq \dots \leq x_m$ 。
4	25	$n, m \leq 10^5$ 。
5	48	無額外限制。



第一題：石頭 (Stone)

問題敘述

Alice 和 Bob 正在玩拿石頭的遊戲。地上有 n 堆石頭，編號為 1 到 n ，且其中第 k 堆石頭有 A_k 個。兩個人輪流拿石頭，拿到最後一個石頭的人獲勝。每次拿石頭的時候，可以先選擇一堆，然後在該堆中拿取至少一顆石頭。

「這怎麼看都是一個拈 (Nim) 遊戲啊！」Alice 不禁抱怨說道：「要判斷是否必勝的方法實在是太過於簡單了，你把每一堆的石頭數量用互斥或 (Xor) 運算加總起來，如果是非零的話就是先手必勝。」

Bob 點頭同意：「對呀，總覺得生活需要來點變化，不然每一次只剩下一堆石頭的時候，你都整堆拿走直接獲勝。」

「不然這樣吧：我們一開始先決定一個數字 R ，先手必須要從某一堆拿走**不超過 R 個**石頭。接下來的每一回合，拿取石頭的人都不能拿超過前一步對手拿的石頭數量，拿走最後一顆石頭的人獲勝。換句話說，如果每一回合拿走的石頭數依序是 r_1, r_2, \dots, r_t 那麼必須要有 $R \geq r_1 \geq r_2 \geq \dots \geq r_t \geq 1$ 。」Bob 提議。

Alice 哆嗦陷入長考。這種拿石頭的單調性改法，也太單調了吧...不明究理的 Alice 想請你幫忙判斷，如果 Alice 是先手的話，第一步有幾種可能的**石堆編號與拿取的石頭數量**的組合能保證最終的勝利呢？

輸入格式

輸入的第一列有兩個以空白隔開的正整數 n, R ，分別代表石頭的堆數與第一步能拿取的石頭上限。輸入的第二列包含 n 個以空白隔開的正整數 A_1, A_2, \dots, A_n ，代表每一堆石頭的數量。

輸出格式

請輸出 Alice 擔任先手的時候，第一步有多少種石堆編號與數量的組合，能保證最終的勝利。

測資限制

- $1 \leq n \leq 10^5$ 。
- $1 \leq R \leq 10^9$ 。
- 對於所有 $1 \leq i \leq n$ ，都有 $1 \leq A_i \leq 10^9$ 。



輸入範例 1

5 1
1 2 3 4 5

輸出範例 1

5

輸入範例 2

5 1
1 1 1 1 2

輸出範例 2

0

輸入範例 3

2 8
5 8

輸出範例 3

4

評分說明

本題共有 5 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	24	$1 \leq n \leq 3$; $1 \leq R \leq 100$; 對於所有 i 皆有 $1 \leq A_i \leq 100$ 。
2	13	$1 \leq R \leq 2$ 。
3	12	$n = 1$ 。
4	23	$1 \leq n \leq 5$; $1 \leq R \leq 100$; 對於所有 i 皆有 $1 \leq A_i \leq 100$ 。
5	28	無額外限制。



第二題：道路鋪設 (Paving)

問題敘述

在喵喵國喵喵縣，有一個發展較為落後的小鄉村，由於交通十分的不便，因此鄉村沒有辦法很好的進行發展，為了振興此鄉村的經濟，喵喵縣政府決定對此鄉村的一條重要對外道路進行建設，在這條道路上鋪設柏油。這條道路可以被劃分為 n 個區段，由西向東依序編號為 $1, 2, \dots, n$ ，每一階段的工程都會選定一些區段然後整段鋪上柏油。由於這條道路如同人生一般漫長，經過了很多階段以後，喵喵縣政府居然不記得那些路段已經鋪設過柏油了！

由於政府希望可以知道仍需要多少材料才能完成工程，因此派出了 m 名調查員去確認當前工程的情況。湊巧的是，最近聘用的調查員們都是**喜歡數論的狂熱分子**，他們利用所學到的數論知識，將調查結果用非常精簡的方式回報給喵喵縣政府。所有回傳的情報都有著相同的格式：

『我是調查員 i 。我看到了從第 ℓ_i 區段到第 r_i 區段之間，所有編號與 x_i 互質的區段皆已鋪上柏油。』

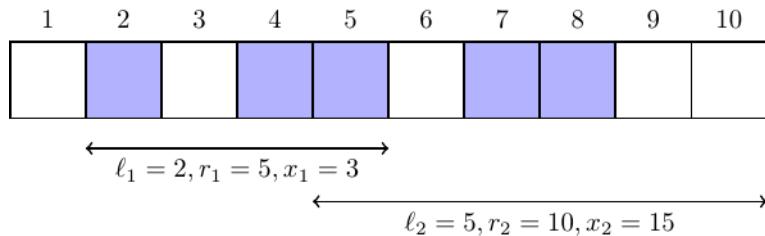


圖 1：考慮一個被分為 10 個區段的漫長道路。根據其中兩名調查員回報的資料，回溯出已鋪設柏油的位置。值得注意的是，儘管 5 這個區段上面已鋪設柏油，2 號調查員卻只有回報 $\{7, 8\}$ 這兩個區段的柏油資訊。5 這個區段則是由 1 號調查員回報的。

已知調查員回報的情況不會有錯，且不會有遺漏。換句話說，對於任何已鋪設柏油的區段，都會被至少一名調查員記錄在回傳的情報中。請你寫一個程式計算還需要鋪設多少單位的柏油路。

輸入格式

輸入的第一列包含兩個正整數 n 、 m ，分別代表區段的數量與調查員的人數。接下來的 m 列，每一列包含三個以空白隔開的正整數 ℓ_i, r_i, x_i ，代表調查員 i 回傳的情報。

輸出格式

請輸出一個整數，代表有多少道路區段是沒有鋪上柏油的。



測資限制

- $1 \leq n \leq 10^9$ 。
- $1 \leq m \leq 1000$ 。
- $1 \leq \ell_i \leq r_i \leq n$ 。
- $1 \leq x_i \leq 100$ 。

輸入範例 1

```
10 2
2 5 3
5 10 15
```

輸出範例 1

```
5
```

輸入範例 2

```
100 1
1 100 64
```

輸出範例 2

```
50
```

評分說明

本題共有 3 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	17	$n, m \leq 1000$ 。
2	29	$1 \leq x_i \leq 20$ 。
3	28	任兩調查員看到的區間並無任何交集。
4	26	無額外限制。



第三題：配對問題 (Pairing)

問題敘述

阿明是一個線上配對專家。《線上配對》是一款有趣的線上遊戲——在數線上進行的遊戲。這個遊戲進行的方式如下：首先，在一條數線上有著 n 個點，由左而右依序編號為 $1, 2, \dots, n$ 。對於每一個 i ，編號為 i 的點帶有一個整數標籤 a_i 。阿明的任務便是要將從中選出許多點對進行匹配，並且獲得最高的總分。一個點只能被匹配至多一次。當兩個點 i 與 j 被配對在一起的時候，阿明就會獲得 $a_i + a_{i+1} + \dots + a_j$ 的分數（這邊我們假設 $i < j$ ）。進行配對的時候，任何兩個配對都不能出現部分相交的情況。換句話說，如果 (i, j) 與 (k, ℓ) 是兩個被選中的配對，其中 $i < j$ 且 $k < \ell$ ，那麼不能出現 $i < k < j < \ell$ 的情形。

匹配結束後，所有沒有被匹配到的點 i ，如果 $a_i > 0$ ，阿明也能夠獲得額外的 a_i 分作為獎勵。而阿明獲得的最終得分就是所有選中的匹配得分，與上述獎勵得分的總和。

為了證明阿明真的是線上配對的專家，阿明的朋友小美決定請你幫忙寫一隻程式，針對每一場遊戲，找出最大的最終得分。

輸入格式

輸入的第一列包含一個正整數 n 。第二列有 n 個以空白隔開的整數 a_1, a_2, \dots, a_n 。

輸出格式

請輸出符合配對規則的配對中最大的最終得分。

測資限制

- $1 \leq n \leq 10^5$ 。
- $-10^9 \leq a_i \leq 10^9$ 。

輸入範例 1

```
3
2 -1 3
```

輸出範例 1

```
5
```



輸入範例 2

4
1 2 3 4

輸出範例 2

15

評分說明

本題共有 4 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	9	$1 \leq n \leq 20$ 。
2	16	$1 \leq n \leq 500$ 。
3	33	$1 \leq n \leq 3000$ 。
4	42	無額外限制。

後記

小美眼見阿明配對獲得了最高總分，不禁感嘆說道：「How is it possible!」



第四題：稜角測試 (Tangent)

本題記憶體限制為 128MB。

問題敘述

在二維平面上給定 n 個相異的點 $p_1 = (x_1, y_1), p_2 = (x_2, y_2), \dots, p_n = (x_n, y_n)$ 。好奇的麗塔想要知道，如下圖的範例所示，若是從上方無窮遠處垂直落下一個線段的話，會先碰到哪個點？如果是線段端點碰到的話也算數。

線段在垂直下落的過程中斜率保持不變，故一線段可以 (a, w, h) 表示，其中 a 代表線段左端點的 x 座標， w 代表左右端點的 x 座標差， h 代表左右端點的 y 座標差。更精確地說，針對線段 (a, w, h) ，若在下落的某時刻左端點的座標為 (x_ℓ, y_ℓ) ，右端點的座標為 (x_r, y_r) ，則 $a = x_\ell$ 、 $w = x_r - x_\ell$ 、 $h = y_r - y_\ell$ 。

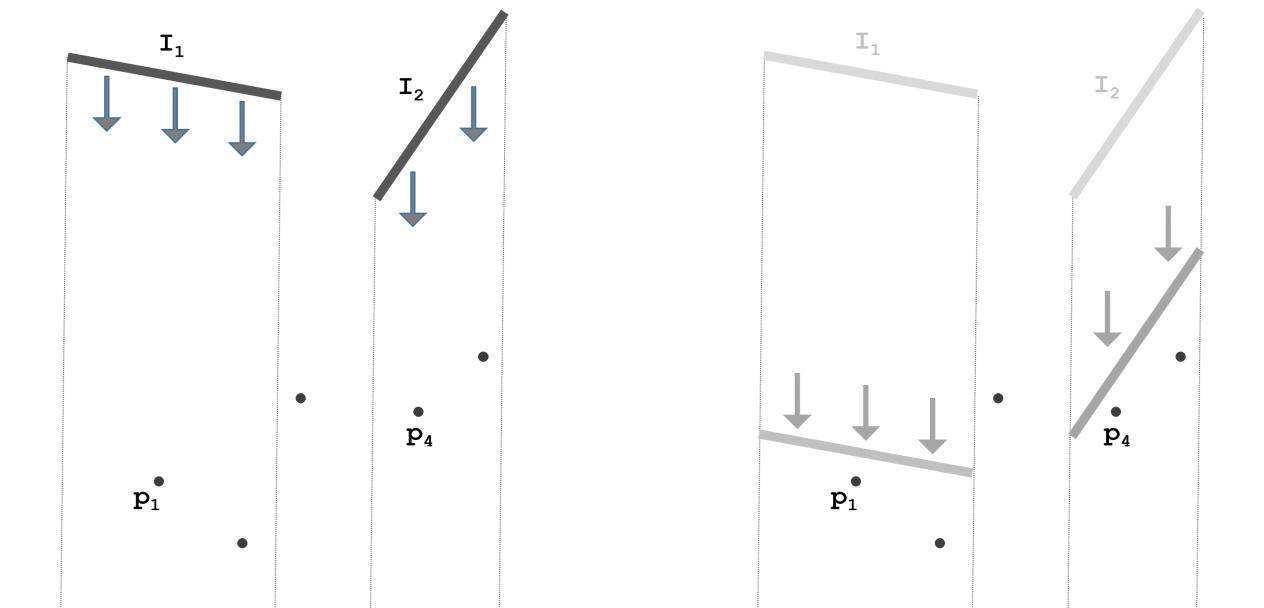


圖 2: 當線段 I_1 垂直落下時，會先碰到 p_1 ；而當線段 I_2 垂直落下時，則會先碰到 p_4 。

輸入說明

測試資料的第一列包含一個正整數 n ，代表給定的點數。接下來有 n 列，每一列有兩個以空白隔開的整數 x_i, y_i ，依序代表 n 個點的座標。第 $n + 2$ 列包含一個正整數 m ，代表欲詢問的線段個數。接下來有 m 列，每一列有三個整數 a, w, h 表示一條落下的線段。



輸出說明

對於每一條落下的線段，請輸出線段垂直落下時，率先碰到的點的索引值。如果有一個以上的點滿足條件，請輸出 x 座標最小的那一個。如果沒有碰到任何點，請輸出 -1 。

測資限制

- $1 \leq n \leq 5 \times 10^5$ 。
- $1 \leq m \leq 10^5$ 。
- $-10^9 \leq x_i, y_i \leq 10^9$ 。
- 對於所有的 $i \neq j$ ，皆有 $(x_i, y_i) \neq (x_j, y_j)$ 。
- $-10^9 \leq a \leq 10^9$ 。
- $0 \leq w \leq 2 \times 10^9$; $-10^9 \leq h \leq 10^9$; $(w, h) \neq (0, 0)$ 。

輸入範例

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

輸出範例

```
1
4
-1
```



評分說明

本題共有 3 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	19	$n, m \leq 1000$ 。
2	30	$n, m \leq 10^5$ 。
3	51	無額外限制。



第一題：摘花 (Flower)

問題敘述

彼得是個愛花的工程師，他的花圃中培育著各式各樣的花朵，這些花朵共有 n 株，且在花圃中整齊地排成一列。

最近彼得打算去花圃中摘取 k 種**不同品種**的花做成花束，卻遇到一個困難：一朵花被摘掉後，花圃就缺了一塊，彼得不希望摘完花後整個花圃變得太過稀疏，更精確來說，彼得不希望同時摘取在花圃中**相鄰**的花。請問在這樣的條件下，彼得是否能順利從花圃中摘取 k 朵花？

若我們用整數來代表花的品種，則花圃中的花可用一個整數序列 F 表示，其中 F_i 代表由左數來第 i 朵花的品種。一個符合條件的摘花方式 P 則可由長度 k 的整數序列表示，其中 $P_i, 1 \leq i \leq k$ ，為由左數來第 i 朵被摘取的花之位置。例如，當 $n = 6, k = 3$, $F = (2, 4, 1, 2, 1, 3)$, F 中的整數 1、2、3、4 分別代表鬱金香、向日葵、玫瑰、蘭花，則彼得可以選擇摘取位置 $P = (2, 4, 6)$ 的花，此時花圃將變為 $(2, \text{空}, 1, \text{空}, 1, \text{空})$ 。注意到摘取 $P = (1, 3, 5)$ 的花並不合條件，因為摘下的花中有兩朵都同樣是鬱金香（即整數 1）；摘取 $P = (1, 2, 5)$ 的花也不合條件，因為前兩朵被摘取的花在序列 F 中相鄰；摘取 $P = (2, 4)$ 的花也不合條件，因為彼得需要 $k = 3$ 朵花。最後注意到，若 $k = 4$ 則 F 中不存在任何符合條件的摘取方式。

請寫一個程式幫助彼得找出一種符合條件的摘花方式 P 。若有多種方式，請輸出所有合法的 P 中字典序最小者，字典序的詳細定義請參照附註。例如，在前一段的例子中 $(2, 4, 6)$ 和 $(1, 3, 6)$ 都是符合條件的摘花方式，但 $(1, 3, 6)$ 的字典序較小。

附註

字典序之定義如下，令 A, B 是兩個長度 k 的相異整數序列，令 i 為滿足 $A_i \neq B_i$ 的最小位置 i ，則我們說 A 之字典序小於 B 之字典序若且唯若 $A_i < B_i$ 。

輸入格式

輸入的第一列包含一個整數 T ，表示測試資料的組數。對於接下來的每一組測試資料，第一列有兩個以空白隔開的整數 n 和 k ，分別代表花圃中花的數量以及彼得需要的花朵數量。第二列包含 n 個以空白隔開的整數，第 i 個整數代表 F_i 。

輸出格式

對於每一組測試資料，請輸出一列包含 k 個以一個空白隔開的整數，代表要摘的花的位置序列 P （若有解，請輸出字典序最小者）。若該組測試資料無法在給定條件下摘取 k 朵花則輸出 0。



測資限制

- $1 \leq n \leq 10^6$ 。
- $1 \leq k \leq \min(22, n)$ 。
- $1 \leq F_i \leq n$ 。

輸入範例

```

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

輸出範例

```

1 3 6
0
1
  
```

評分說明

本題共有 3 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	16	$1 \leq k \leq n \leq 18$ 。
2	41	$1 \leq n \leq 100$ 、 $1 \leq k \leq 20$ 、 $k \leq n$ 。
3	43	無額外限制。



第二題：貓咪大集合 (Neko Atsume)

問題描述

喵喵國有 n 個城市，城市與城市之間以雙向隧道來連接。喵喵國王很辛苦的開闢了 $n - 1$ 條隧道讓任一城市都可以經由一或多條隧道抵達其他所有城市。現在是舉行宴會慶祝的時候了！

喵喵國王想舉辦一場宴會，這場宴會一開始邀請 m 位住不同地方的貴賓來參加。每一場宴會的賓客們會從居住的城市移動到舉辦宴會的城市，結束後再移動回居住的城市。為了體貼遠道而來的朋友們，喵喵國王想把宴會辦在最方便的城市，也就是讓距離最遠的朋友的移動距離最短。（喵喵國王是無私的，他不考慮自己的移動距離。）

然而，喵喵國王的朋友們大多公務繁忙，不一定有空出席。喵喵國王宴會前會依序收到 d 封朋友的訊息，說他已經排了其他行程無法前來，或是把其他行程排開了因此可以前來。

在出席名單一直變動的情況下，宴會的地點也得跟著更改。請你寫一個程式，根據喵喵國的地圖和出席名單的變化，告訴喵喵國王每收到一封訊息後若舉辦在最方便的城市，則住最遠的人需要移動多遠。

輸入格式

輸入的第一列包含一個正整數 n 、 m ，代表喵喵國有幾個城市以及一開始出席名單的人數。接下來的 $n - 1$ 列，每一列包含三個以空白隔開的正整數 u_i, v_i, w_i ，代表有一條隧道連接城市 u_i 和 v_i ，而這條隧道的長度是 w_i 。接下來一行有 m 個相異整數 x_1, x_2, \dots, x_m ，分別代表一開始要參加宴會的人住的位置。接下來的一列有一個正整數 d ，代表訊息有幾封。接下來的 d 列每行為一則訊息，第 i 行為第 i 封收到的訊息並有一個整數 p_i 。若 $p_i > 0$ 表示居住在編號 p_i 城市的朋友說他可以前來，若 $p_i < 0$ 則表示居住在編號 $-p_i$ 城市的朋友無法前來。

輸出格式

一開始請先輸出一列，代表未收到任何訊息時宴會參加者最遠的移動距離。接著請輸出 d 列，第 j 列有一個整數代表收到第 j 封訊息後宴會參加者最遠的移動距離。若任意時間點宴會沒有賓客參加的話請輸出 0。

測資限制

- $2 \leq n \leq 10^5$ 。
- $0 \leq m \leq n$ 。
- $1 \leq x_i \leq n$ 。
- $0 \leq d \leq 2 \times 10^5$ 。
- $-n \leq p_i \leq n$ ，且 $p_i \neq 0$ 。
- $1 \leq w_i \leq 1000$ 。
- $1 \leq v_i, u_i \leq n$ 。



- 紿定的圖 n 個點互相連通。
- 任意時間點，參加者名單所住的城市不重複。
- 訊息保證沒有矛盾，意即：
 - 若第 i 個訊息為不參加，則此人原本的狀態必為參加。
 - 若第 i 個訊息為參加，則此人原本的狀態必為不參加。

範例輸入 1

```
7 1
1 2 1
1 3 1
2 4 1
2 5 1
3 6 1
3 7 1
2
6
4
6
-2
1
-6
5
```

範例輸出 1

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

**範例輸入 2**

```
7 4
1 2 1
1 3 1
2 4 1
2 5 1
3 6 1
3 7 1
2 4 6 1
1
-2
```

範例輸出 2

```
2
2
```

評分說明

本題共有 5 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	7	$n \leq 100, d \leq 100$ 。
2	12	任意時間點參加的人數只有 2 人。
3	22	每個時間點參加人數總和 $\leq 4 \times 10^6$ 。
4	24	$p_i > 0$ 。
4	35	無額外限制。



第三題：產品製造 (Production)

問題描述

現有 n 台機器，機器之間有輸送帶互相相連結，前面機器的產出就是後面機器的原料。這台機會依據一組固定的比例來分配其產出送給其相鄰的機器，我們將此固定的比例視為一機率分布。編號 1 的機器是唯一一台可直接由原料倉庫取料的機器，是整個生產線的起點；當然，在生產的過程中，編號 1 的機器的原料來源也可以是其他機器。如果一台機器後面沒有接任何機器，這台機器被視為一個終點，它的產出就是最後的成品，生產線即停止。一台機器可以從不只一台機器獲得原料，不過無論是從哪台機器獲得的原料，這台機器的產出以及向後輸送的機率都是不變的。

每台機器有一個固定的執行時間，不會根據原料的來源或產出的去向而改變。輸送產出到下一台機器的時間可以忽略。我們在意的是從編號 1 的機器開始生產，到最後產出成品所需的時間。由於這個時間會依據生產過程中經過的機器不同而有所改變，我們可以計算這個時間的期望值；精確來說，我們稱一個成品在生產過程中依序經過的所有機器為該成品之「生產路徑」（可能包含重複的機器），該路徑的機率是其上每條邊的機率乘積，時間是其上所有機器的生產時間加總（重複走到的點要累計）。

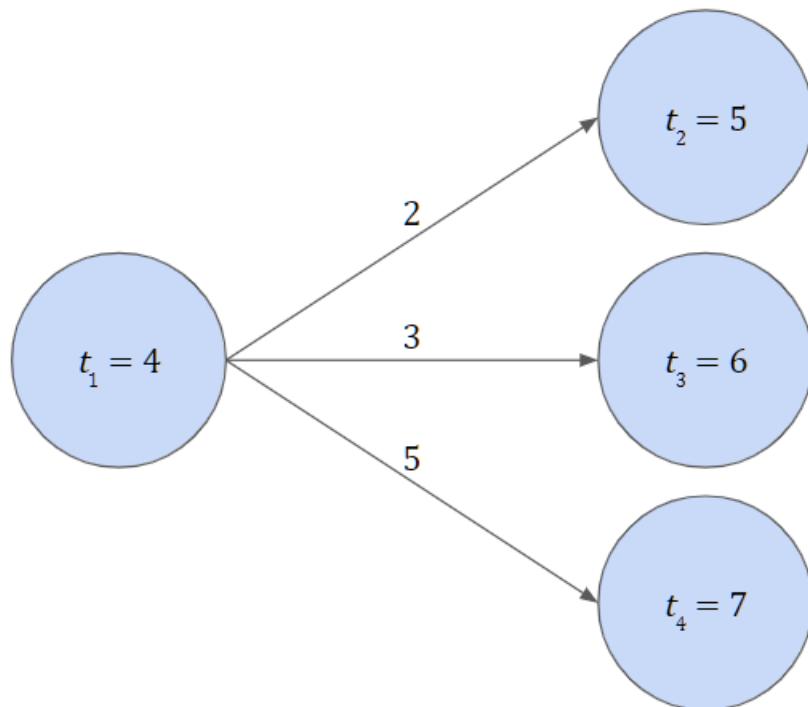


圖 1：機器與生產線的示意圖。

以上圖為例，機器 1 的產出有 $\frac{2}{10}$ 的機率輸送到機器 2 、 $\frac{3}{10}$ 的機率到機器 3 、 $\frac{5}{10}$ 的機率到機器 4。整個生產時間的期望值是 $\frac{2}{10}(4 + 5) + \frac{3}{10}(4 + 6) + \frac{5}{10}(4 + 7) = 10.3$ 。

我們現在有 k 個加速器可以加在任意的機器上。如果在編號為 i 的機器加上 x 個加速器，那麼該機器的執行時間會從 t_i 加快至 $t_i/(1+x)$ 。也就是說，一個機器加上第一個加速器，則它的執行時間會減半。加上



第二個加速器，則它的執行時間會變成最原先的三分之一，以此類推。請決定將 k 個加速器分別加在哪些機器上，使生產時間的期望值最小。

輸入格式

輸入的第一列包含三個正整數 n, m, k ，分別代表機器的數量、輸送帶的數量、和加速器的數量。下一列有 n 個正整數 t_1, \dots, t_n ，代表第 i 台機器的執行時間。接下來有 m 列，每一列包含三個以空白隔開的正整數 u_i, v_i, w_i ，代表有一條輸送帶連接機器 u_i 和 v_i ，而輸送的比重是 w_i 。

輸出格式

請輸出加上所有加速器後，最小的生產時間期望值。（由於浮點數精確度有限，輸出的數值和正確答案的絕對或相對誤差不超過 10^{-6} 都會被視為正確。）

測資限制

- $1 \leq n \leq 10^5$ 。
- $1 \leq m \leq 2 \times 10^5$ 。
- $1 \leq k \leq 10^5$ 。
- $1 \leq t_i \leq 10^9$ 。
- $1 \leq w_i \leq 10^6$ 。
- 所有測資保證從編號 1 的機器可以經由若干條輸送帶抵達任一台機器。
- 所有測資保證任一台機器都可以經由若干條輸送帶抵達至少一個終點。

範例輸入 1

```
5 4 8
2 2 2 2 2
3 5 1
1 3 1
1 4 1
3 2 1
```

範例輸出 1

1.833333



範例輸入 2

```
5 4 73
1 4 2 4 2
1 3 1
1 5 1
5 4 1
5 2 1
```

範例輸出 2

0.320833

範例輸入 3

```
8 7 8
3 10 5 4 7 6 4 6
7 6 757756
7 5 242244
1 3 1000000
2 8 76562
2 4 568826
2 7 354612
3 2 1000000
```

範例輸出 3

10.089585

範例輸入 4

```
4 4 2
1 1 1 1
1 2 1
1 3 1
2 4 1
3 4 1
```

**範例輸出 4**

2.000000

範例輸入 5

```
4 4 2
1 1 1 1
1 2 1
2 3 1
3 1 1
2 4 1
```

範例輸出 5

4.000000

評分說明

本題共有 4 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	9	$n \leq 100$ 、 $m = n - 1$ 、 $k \leq 1000$ 。
2	49	$m = n - 1$ 。
3	23	測資保證每一台機器的產出沒有辦法回到同一台機器當原料。
4	19	$n \leq 100$ 。



第四題：麵包板 (Breadboard)

問題描述

麵包板是一種不需要焊接、只需要將導線連接起來就能實現電路設計初步成果的電路板。由於只需要將導線插入不同的座標孔，就能夠透過預設的基底將線路連接起來，技術較差的連線方式將會導致麵包板上面的線路亂七八糟，影響外觀與除錯時的心情。

今天的電子學實驗課需要在麵包板上安置元件和連接導線，在上實驗課之前，教授特別提醒不想看到導線連接混亂的情形。良好的連接需要同時滿足以下兩點規則：

1. 每個元件都需要放置在麵包板上的某個座標為 (x, y) 的位置上，其中 x 座標只能是正整數， y 座標只能是 1 或是 0。每個座標上只能放至多一個元件。
2. 每條導線連接兩個元件，連接方式是從其中一個元件拉一條直線線段至另一個元件上。每條導線連接的元件需要放置在不同的 y 座標上，且任兩條導線不能有交叉，若兩條導線連接到同一元件，不視為交叉。

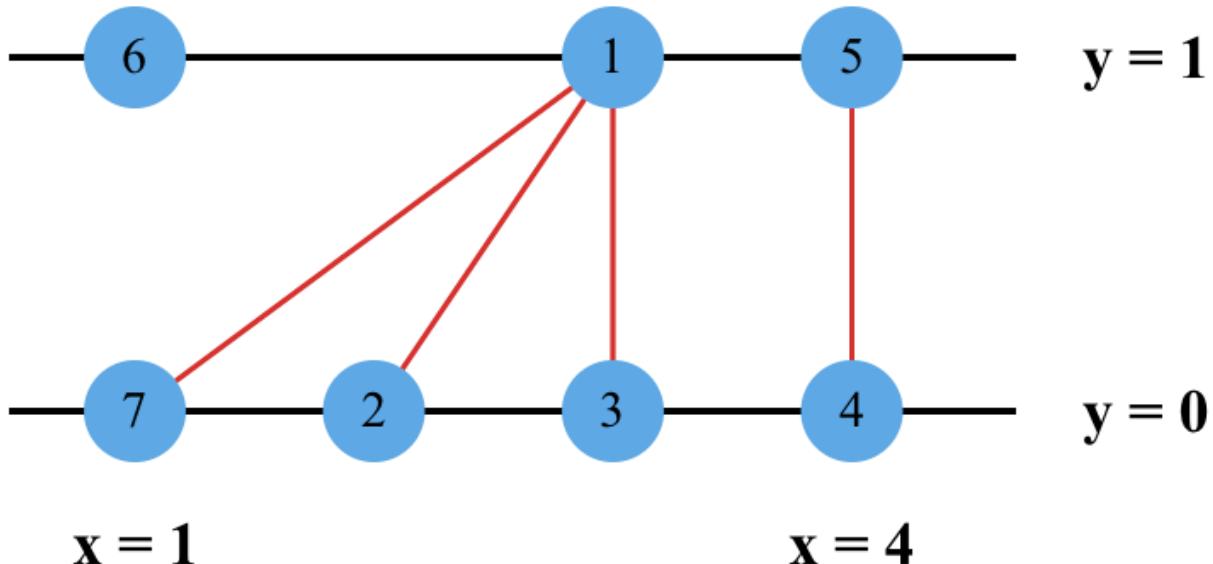


圖 2：輸入範例 1 的其中一種排版方法。

除了以上的規則外，為了讓實驗課的分數可以高一點，想自我要求以最清爽的方式完成，而清爽的定義是所有元件放置的 x 座標都在某個 k 值以下且 k 值盡可能地小。



輸入格式

輸入第一列為兩個整數 n 與 m ，其中 n 代表元件的個數（編號為 $1, 2, \dots, n$ ）、 m 代表導線的個數。接下來的 m 列，每一列包含一對以空白隔開的整數，第 i 列的整數 u_i, v_i 代表第 i 條導線所連接的兩個元件編號。

輸出格式

輸出為一整數，如果輸入的電路無法以滿足規則的方式完成排版，則輸出 -1 。若可以滿足，則輸出最清爽排版所需的 k 值。

測資限制

- $1 \leq n \leq 10^5$ 。
- $0 \leq m \leq 10^6$ 。
- 導線只會用來連接兩個不同的元件，而且同一對元件只會被至多一條導線所連接。也就是說對於所有的 i ，皆有 $1 \leq u_i, v_i \leq n$ 並且 $u_i \neq v_i$ 。此外對於所有的 $i \neq j$ 皆有 $(u_i, v_i) \neq (u_j, v_j)$ 且 $(u_i, v_i) \neq (v_j, u_j)$ 。

範例輸入 1

```
7 4
1 2
1 3
1 7
5 4
```

範例輸出 1

```
4
```

範例輸入 2

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



範例輸出 2

-1

評分說明

本題共有 4 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	5	$n \leq 20$ 。
2	10	$n \leq 100$ 。
3	31	$n \leq 10000$ 。
4	54	無額外限制。



第一題：聚光燈 (Spotlight)

問題敘述

提歐埃國最近舉辦了盛大的夏日慶典，為了增添節目熱鬧的氣氛，夏日慶典的規劃部門在舞台上方架設了許多特殊的聚光燈。每一盞聚光燈投射在舞台平面上，都會形成一個凸多邊形的發亮區域。更特別的地方則是，聚光燈在設計的時候，是可以調整為等速移動的，也就是說，如果給定了初始位置和每秒移動的向量，那麼這個發亮的區域會等速沿著該向量移動，**直到永遠**（特別吧！？）。

『夏日慶典的安全，就是你我的安全！』身為專業的活動企劃，這種高強度聚光燈的架設方案是需要經過嚴加考慮的。如果有許多發亮的區域重疊在一起、或是重疊時間過久，可能會導致該區域冷熱不均，影響舞台表演人員的身心健康安全。因此，請你寫一支程式，判斷每一盞聚光燈與任一其他聚光燈探照的區域接觸到的時間總長。請注意，如果兩盞聚光燈的探照區域邊界有重疊或是碰觸到一點，也算是重疊。

輸入格式

輸入的第一列包含一個正整數 n ，代表聚光燈的數量。接下來的 n 列，每一列會有 $2k_i + 3$ 個以空白隔開的整數，其中 k_i 是描述第 i 個聚光燈探照出的多邊形區域的頂點數。這些整數的格式如下：

$$dx_i \ dy_i \ k_i \ x_{1,i} \ y_{1,i} \ x_{2,i} \ y_{2,i} \ \dots \ x_{k_i,i} \ y_{k_i,i}$$

其中 (dx_i, dy_i) 代表探照區域在單位時間內移動的方向向量；緊接在 k_i 後的則是 k_i 個座標 $(x_{1,i}, y_{1,i}), \dots, (x_{k_i,i}, y_{k_i,i})$ ，依序為在第 0 秒鐘的時候，以**逆時針順序**描述探照區域的頂點位置。

輸出格式

對於每一組測試資料，請輸出 n 列，其中第 i 列包含一個小數 t_i ，代表第 i 盞聚光燈與其他任何聚光燈探照區域接觸的時間總長。如果永遠有交集，請輸出 **infinity**。

你的輸出必須與實際值之絕對或相對誤差，不超過 10^{-6} 才算回答正確。也就是說，如果你的輸出值為 $t_{\text{輸出}}$ ，實際的答案為 $t_{\text{答案}}$ ，那麼只要滿足

$$\min \left(|t_{\text{輸出}} - t_{\text{答案}}|, \frac{|t_{\text{輸出}} - t_{\text{答案}}|}{t_{\text{答案}}} \right) \leq 10^{-6}$$

就算回答正確。



測資限制

- $2 \leq n \leq 100$ 。
- 對所有的 $1 \leq i \leq n$ 與 $1 \leq j \leq k_i$ 皆有 $-10000 \leq dx_i, dy_i, x_{j,i}, y_{j,i} \leq 10000$ 。
- $3 \leq k_1, k_2, \dots, k_n \leq 100$ 。
- 每一個輸入的多邊形保證都是凸的。並且輸入的點保證在多邊形頂點上，不會在邊上。

輸入範例 1

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

輸出範例 1

```
2.0
2.0
```

輸入範例 2

```
2
0 0 4 0 0 2 0 2 2 0 2
0 0 4 2 0 2 -2 4 -2 4 0
```

輸出範例 2

```
infinity
infinity
```

輸入範例 3

```
2
0 1 3 1 1 3 1 2 2
0 0 3 1 3 2 3 2 4
```

輸出範例 3

```
2.0
2.0
```



評分說明

本題共有 4 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	15	每個多邊形都是每邊皆平行於 X 軸或 Y 軸的矩形。
2	11	對於所有的 i ，皆有 $dx_i = dy_i = 0$ 。
3	17	對於所有的 i ，皆有 $dx_i = 0$ 或 $dy_i = 0$ 。
4	57	無額外限制。



第二題：疫情下的排隊大考驗 (Line Up)

問題敘述

你是一間餐廳的經理。這間餐廳在防疫期間，只做現點現做的外帶生意。所有等候外帶的客人必須要在結帳櫃台前面排成一排。可惜的是，由於防疫的關係，餐廳只能容納 k 名顧客排隊。服務任何一位客人所需要的時間，恰好都是 s 秒鐘。

根據你的神預測，今天將有 n 名顧客出現在你的餐廳，第 i 名顧客會準時在時間 t_i 抵達你的餐廳。如果成功服務了第 i 名顧客，這名顧客將會為餐廳帶來 p_i 的利潤。

如果一位顧客出現在餐廳的瞬間，你剛好服務完一位客人，那麼這位新顧客與舊顧客不會同時出現在隊伍裡。如果這位顧客加入隊伍時，他不巧是第 $k + 1$ 名顧客，那麼他會被擠在門外，然後悻悻然地離開，然後給你們餐廳負評。

請你勸退一些顧客，使得剩下準時出現在你餐廳的顧客都能夠成功獲得餐點。請問餐廳可能的最大收益為何？

輸入格式

輸入的第一列有三個以空白隔開的整數 n, k, s 。接下來的 n 列，每一列有兩個以空白隔開的整數 t_i 和 p_i 。

輸出格式

請輸出一個整數代表餐廳可能的最大收益。

測資限制

- $1 \leq k \leq n \leq 4000$ 。
- $1 \leq s \leq 10^6$ 。
- $1 \leq t_i \leq 10^9$ 。
- $1 \leq p_i \leq 10^6$ 。



輸入範例 1

2 1 58
40 1
74 1

輸出範例 1

1

輸入範例 2

3 2 10
2 100
7 200
12 100

輸出範例 2

400

評分說明

本題共有 4 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	14	$n \leq 20$; $s \leq 1000$; $t_i \leq 2000$ 。
2	9	$k = 1$ 。
3	24	$n \leq 100$; $s \leq 1000$ 。
4	53	無額外限制。



第三題：蚯蚓的排字大考驗 (Earthworm)

問題敘述

畢業季即將來臨的時刻，大家總是能集結三五好友，約定在同一時間，跳躍起來排成若干英文字母或符號並且拍照，留作紀念。

蚯蚓們也是如此。蚯蚓學校今年流行的畢業照拍法很特別，他們選定一棵樹（對，就是資訊界常用的那種樹），然後每一隻蚯蚓分別佔據一部分連續的樹枝（對應到樹上的一條路徑），而且剛好填滿整棵樹。由於樹枝相當細，因此每一條樹枝都恰好被一隻蚯蚓的一部分覆蓋住。而這組蚯蚓的得分就是每隻蚯蚓覆蓋住的樹枝（邊）數量的**平方和**。

蚯蚓們相當有默契，每一隻都長得一模一樣。所以只有在兩種覆蓋方法中，覆蓋出的路徑集合不同的時候，才會被視為兩種不同的覆蓋方法。蚯蚓們想要計算出所有覆蓋的方法的得分總和，除以 $10^9 + 7$ 的餘數。由於不同的樹會得到不同的總和，蚯蚓希望透過湊出最幸運的總和數字，留下最完美的畢業照！聰明的你，可以幫幫蚯蚓們計算，給定一棵樹的情形下，所有可能覆蓋方法的得分總和，除以 $10^9 + 7$ 的餘數嗎？

輸入格式

輸入的第一列包含一個正整數 n ，代表樹上的節點數，節點的編號為 $0, 1, \dots, n - 1$ 。接下來有 $n - 1$ 列，每一列包含以空白隔開的兩個整數 a_i, b_i ，代表一條連接編號為 a_i 與 b_i 節點的樹枝。

輸出格式

請輸出所有蚯蚓覆蓋方法得到的得分總和，除以 $10^9 + 7$ 的餘數。

測資限制

- $2 \leq n \leq 5000$ 。
- 對於所有 i 皆有 $0 \leq a_i, b_i \leq n - 1$ 。

輸入範例 1

```
4
0 1
0 2
0 3
```



輸出範例 1

18

輸入範例 2

4
0 1
1 2
2 3

輸出範例 2

22

評分說明

本題共有 5 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	13	$n \leq 12$ 。
2	19	$n \leq 100$ 。
3	7	$n \leq 5000$ ，並且所有節點的度數不超過 2。
4	9	$n \leq 5000$ ，並且所有節點的度數不超過 3。
5	52	無額外限制。



第四題：小朋友的牌組大考驗 (Cards)

問題敘述

今天老師要教 m 位小朋友數學中全序 (total order) 的重要性。老師首先發給每一位小朋友一張空白卡片，要每一位小朋友在卡片上隨意地從集合 $\{x_1, x_2, \dots, x_n\}$ 中挑選兩個變數 x_i 與 x_j ，並且寫下 $x_i < x_j$ 、 $x_i = x_j$ 、或 $x_i > x_j$ 三者其中之一的敘述。然後在卡片上留下自己的座號。

接著，老師會不斷地隨意挑選兩個座號 a 與 b ，並且指定「帶著座號 a 卡片的小朋友」將手上所有的卡片，拿給「帶著座號 b 卡片的小朋友」。合併完兩堆卡片以後，同學們要齊心推理，看看是否存在一些正整數代入這些變數後，滿足所有卡片上的敘述。重複以上動作直到所有卡片都被收整齊在同一個小朋友身上為止。

請你寫一個程式判斷小朋友們的推論是否正確。

輸入格式

- 第 1 列包含兩個正整數 n 與 m ，代表變數的數量與小朋友的數量。
- 第 2 至 $m + 1$ 列：每一列包含一個以空白作為間隔的敘述 $i_k \ op_k \ j_k$ ，表示第 k 個小朋友在卡片上寫的敘述。其中 i_k 和 j_k 為變數的編號，而 $op_k \in \{<, =, >\}$ 表示兩個變數的關聯。
- 第 $m + 2$ 至 $2m$ 列：每一列包含兩個以空白隔開的整數 a_k, b_k ，代表老師依照時間順序，指定將帶有座號 a_k 的那疊卡片與帶有座號 b_k 的卡片合併。

輸出格式

對於每一次的合併，如果存在一些正整數代入變數後滿足所有的敘述，請輸出 Yes，否則輸出 No。

測資限制

- $1 \leq n \leq 10^5$ 。
- $1 \leq m \leq 5 \times 10^5$ 。
- 對於所有 k 皆有 $i_k, j_k \in \{1, 2, \dots, n\}$ ； $i_k \neq j_k$ 。
- 對於所有 k 皆有 $op_k \in \{<, =, >\}$ 。
- 對於所有 k 皆有 $a_k, b_k \in \{1, 2, \dots, m\}$ 。
- 每一次合併的時候，保證 a_k 與 b_k 所在的卡片不是同一疊。

輸入範例 1

```
4 4
1 = 2
2 > 3
```



3 > 4
4 > 1
1 2
3 4
1 3

輸出範例 1

Yes
Yes
No

輸入範例 2

4 4
1 = 2
2 > 3
3 > 4
4 < 1
1 2
3 4
1 3

輸出範例 2

Yes
Yes
Yes

評分說明

本題共有 3 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	12	$m \leq 10^5$; $n \leq 50$ 。
2	31	$m \leq 10^5$; $n \leq 1000$ 。
3	57	無額外限制。