

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
 $a_1 a_2 \cdots a_n$

- 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

$b_1 b_2 \cdots b_n$

- 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

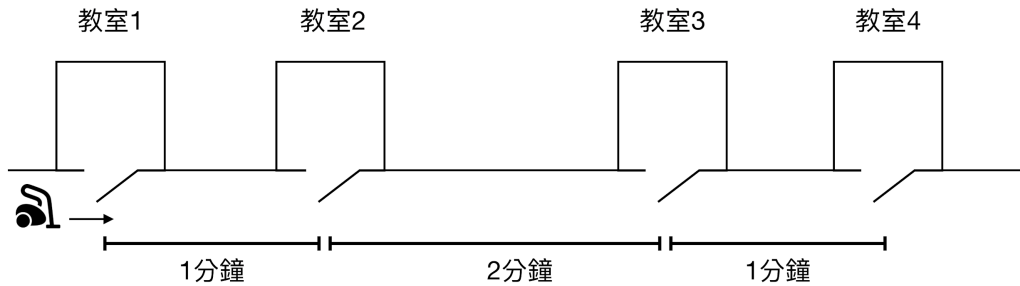
本題共有二組子任務，條件限制如下所示。每一子任務可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

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
t1 t2 ⋯ tn-1
s1 s2 ⋯ sn
d1 d2 ⋯ dn
```

- 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。

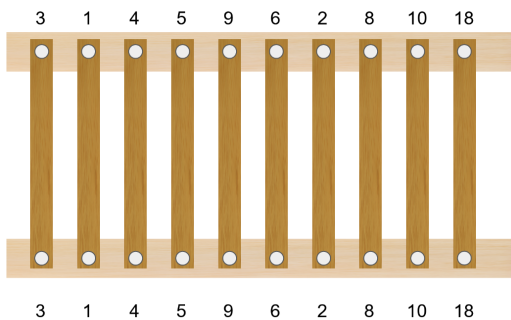
C. 粉刷護欄

問題描述

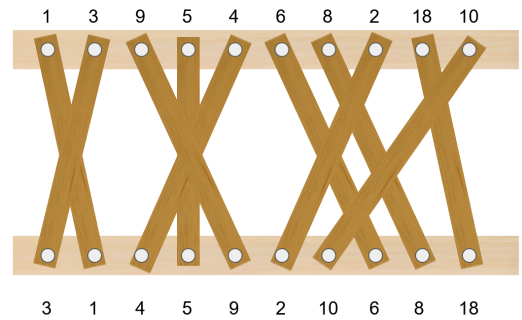
紅心國的黑桃國王為了促進國民身心健康，大力在國內修繕河濱自行車道，然而河道並不筆直，騎車時在彎道處不減速很容易衝出車道導致事故。為了減少事故，黑桃國王決定在彎道處搭建護欄，並找了方塊設計公司與梅花人力公司來完成這項工程。

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。



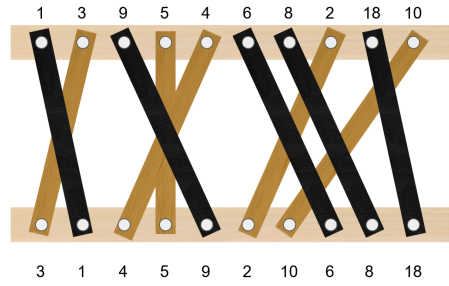
圖一



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
a1 a2 ⋯ an
b1 b2 ⋯ bn
    
```

- 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

```
c1 c2 ⋯ ck
```

- 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
u1 v1 w1 a1 b1 p1
u2 v2 w2 a2 b2 p2
⋮
un-1 vn-1 wn-1 an-1 bn-1 pn-1
h1 m1 s1 t1
h2 m2 s2 t2
⋮
hq mq sq tq

```

- 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 $s_i \neq t_i$ 。 ($i \in \{1, 2, \dots, q\}$)
- 給定的 $n-1$ 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 給予一個 $n \times n$ 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。