

A. 字串解壓縮

問題描述

運行長度編碼（run-length encoding）為一字串壓縮方式，針對一給定的原始字串 S ，若 S 中有一子字串由 t 個字元 x 構成且該子字串的前一個與後一個字元皆非 x ，則會以 tx 表示該子字串。若要壓縮的子字串僅為一個字元（即 $t = 1$ ），則會將 1 省略以讓儲存空間更節省。舉例來說，`aaabcccccccccba` 從最前面開始 a 連續出現 3 次，b 出現 1 次，c 出現 11 次…，因此經過長度編碼後可以表示為 `3ab11cbaba`。假設構成原始字串的字元僅有 26 個小寫英文字母，請撰寫一解碼器，將長度編碼過後的字串 C 還原為原始字串 S 。

輸入格式

輸入為一行字串 C 代表長度編碼過後的字串，長度至多 1000。原始字串僅由 26 個小寫英文字母構成，每個字母連續出現的次數至多 100。

輸出格式

輸出一行字串 S ，為輸入運行長度編碼對應的原始字串，測資保證解壓縮後的字串長度不超過 10^5 。

測資限制

- $1 \leq |C| \leq 1000$ 。
- C 僅由小寫英文字元 a-z 以及字元 0-9 組成。
- C 是一個合法的長度編碼字串，意即 C 是由 S 經過長度編碼壓縮所得的結果。
- S 僅由字元 a-z 組成。
- S 中任意字母連續出現次數不超過 100。

範例測試

Sample Input	Sample Output
<code>mi4si2si3pi</code>	<code>mississippi</code>
<code>14ops</code>	<code>ooooooooooooops</code>

評分說明

本題共有二組測試題組，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	40	編碼前字串每個字元連續出現的次數至多為 9。
2	60	無額外限制。

B. 建設人工島

問題描述

棕櫚島是杜拜城著名的人工島，也是世界知名的觀光勝地，島的形狀就像一棵棕櫚樹，每年都吸引了上千萬人前往旅遊。為了促進觀光產業的發展，踢歐埃大公國也計畫建立一座人工島，叫做「樹狀結構島」。整座島是由 n 個景點及 $n - 1$ 條互不相交的雙向平面道路連接起來，每條道路連接兩個相異的景點。若把這些景點當成節點且道路當成邊來看的話，可以發現整個島恰是一個樹狀結構的圖。也就是說，任兩點之間都是連通的，並且其間只存在唯一一條路徑。並且兩景點間的距離，定義為其間路徑的長度（即該路徑上所有道路的長度總和）。

為了發展觀光產業，人工島預計要將某些道路改建成高速公路，使得這些道路串連後恰好是連接某兩個最遠景點的路徑。然而因為經濟的動蕩，要搭建時發現預算不足，無法建立太長的高速公路，因此踢歐埃大公國決定建立一條高速公路連結島上「次遠」的兩景點。所謂「次遠」的兩景點是指一對景點，其間的距離為「比最遠距離小」的所有可能裡最大的。以下面的範例測試一為例，所有景點對的距離由大至小排列為 $2, 2, 2, 1, 1, 1$ ，所求之次遠距離為 1。

請寫一支程式，幫助踢歐埃大公國，計算樹狀結構島上次遠的兩點距離。

輸入格式

輸入共 n 行，第一行有一個正整數 n 代表樹狀結構島上景點的個數；景點編號由 0 至 $n - 1$ 。接下來有 $n - 1$ 行輸入，每一行有 3 個整數 u_i, v_i, w_i ，代表景點 u_i 與景點 v_i 之間有一條雙向道路連接，且這條道路的長度為 w_i 。同一行的連續兩整數間以一個空白分隔。

輸出格式

輸出為一正整數，代表所求之次遠距離。

測資限制

- $3 \leq n \leq 10^5$ ，且 n 為整數。
- 對所有 $1 \leq i \leq n - 1$ ，滿足 u_i, v_i 為 0 到 $n - 1$ 之間的整數。
- 對所有 $1 \leq i \leq n - 1$ ，滿足 w_i 為整數，且 $1 \leq w_i \leq 100$ 。
- 紿定的圖為一棵樹。

範例測試

Sample Input	Sample Output
4 0 1 1 0 2 1 0 3 1	1
8 7 0 2 0 1 8 0 5 6 6 5 10 2 4 10 3 4 18 5 4 2	30
4 0 1 1 1 2 2 3 2 3	5

評分說明

本題共有三組測試題組，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	$n \leq 100$
2	34	任意 $w_i = 1$ 。
3	47	無額外限制。

C. 銀河捷運

問題描述

在某個不知名的外星王國中，外星人發展了一套獨特的數字系統與銀河捷運系統。根據科學家的研究，外星人的數字系統可對應到我們所使用的「模 M 運算」，其中 M 為一個質數。以下用模運算的術語來描述外星人的數字系統。以 $M = 7$ 為例，下列的算式在外星球的數學系統是成立的：

- $2 \times 3 = 6$
- $2 \times 5 = 3$
- $2 + 3 = 5$
- $2 + 6 = 1$
- $3 - 1 = 2$
- $3 - 5 = 5$

我們的研究人員還發現了銀河捷運系統與他們的數字系統存在著高度的關聯。首先，銀河捷運系統和我們在地球上常見的直線形軌道或是環形軌道不一樣，外星人自己有一套獨特的站台分佈機制以及捷運運行技術。再來，我們發現銀河捷運系統的每一條路線都可以用一個方程式 $y^2 = x^3 + ax + b$ 和一質數 M 來表示，而這個方程式上的每個整數點 (x, y) 都分別代表一個這條路線上的站台。以下面這條捷運路線為例：

$$y^2 = x^3 - x + 1, M = 13$$

座標 $(3, 8)$ 是這條路線的站台，因為 $8^2 \equiv (3^3 - 3 + 1) \equiv 12 \pmod{13}$ 。除此之外， $(0, 1), (1, 1), (3, 5)$ 也是這條路線上的站台。

外星王國中對直線的定義也和我們類似，例如在地球上，我們說三個 **x** 座標相異的點 $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ 共線代表他滿足：

$$\frac{y_1 - y_2}{x_1 - x_2} = \frac{y_2 - y_3}{x_2 - x_3}$$

雖然我們並不知道外星人是如何做除法的，但我們知道交叉相乘的法則仍然通用。因此外星球中三點共線的條件可以寫成：

$$(y_1 - y_2)(x_2 - x_3) \equiv (y_2 - y_3)(x_1 - x_2) \pmod{M}$$

外星人要搭乘銀河捷運的時候，要先輸入自己所在的站台座標 (x_1, y_1) ，接著輸入另外一個站台的

座標 (x_2, y_2) 作為參考值，並再找出第三個站台座標 (x_3, y_3) ，使得這三個點共線。這第三站台就是外星人要前往的地方。

在給出站台座標 (x_1, y_1) 與參考值站台座標 (x_2, y_2) 的情況下，研究人員告訴我們一個可以求出目的地座標 (x_3, y_3) 的方法：首先我們可以假設一條直線 $y \equiv mx + k \pmod{M}$ 作為同時通過三個點的直線方程，直線方程式中的斜率 m 可以利用費馬小定理 $x \cdot x^{M-2} \equiv 1 \pmod{M}$ 得出 $m \equiv (y_2 - y_1) \cdot (x_2 - x_1)^{M-2} \pmod{M}$ 的結論。接著，將這條直線方程代進捷運路線的方程式可以得到 $(mx + k)^2 \equiv x^3 + ax + b \pmod{M}$ 。並且從根與係數的法則可以得出 $x_1 + x_2 + x_3 \equiv m^2 \pmod{M}$ 進而求出目的地的 x 及 y 座標。

給定一條銀河捷運路線的參數 a 、 b 、 M 以及外星人的出發站台和中間所有的參考站台之座標，請撰寫一支程式算出最後外星人所抵達站台的座標。

輸入格式

輸入第一行有一個正整數 t ，代表測資的筆數。接下來 t 行每行有 7 個整數 $M, a, b, x_1, y_1, x_2, y_2$ 。其中 M 為模數， a 和 b 為捷運路線方程式的二參數， (x_1, y_1) 及 (x_2, y_2) 分別為起點及參考點的座標。同一行的兩數字間以一空白分隔。

輸出格式

對於每筆測資請輸出一行含兩數字 x_3, y_3 ，以一空白分隔。 x_3 與 y_3 皆為 0 以上 $M - 1$ 以下(包含 0 與 $M - 1$)的整數，代表外星人將前往的站台座標。

測資限制

- $1 \leq t \leq 10^5$ ，且 t 為整數。
- a, b, x_1, y_1, x_2, y_2 都是 0 以上 $M - 1$ 以下(包含 0 與 $M - 1$)的整數。
- $x_1 \neq x_2$ 。
- $2 \leq M < 2^{31}$ ， M 為一質數。
- 保證存在第三個站台 (x_3, y_3) ，使得 $x_3 \neq x_1$ 且 $x_3 \neq x_2$ ，並且與 (x_1, y_1) 和 (x_2, y_2) 共線。

範例測試

Sample Input	Sample Output
3	6 1
13 1 0 4 4 2 7	7 12
13 3 1 0 12 6 12	12 6
13 3 1 7 12 10 11	

評分說明

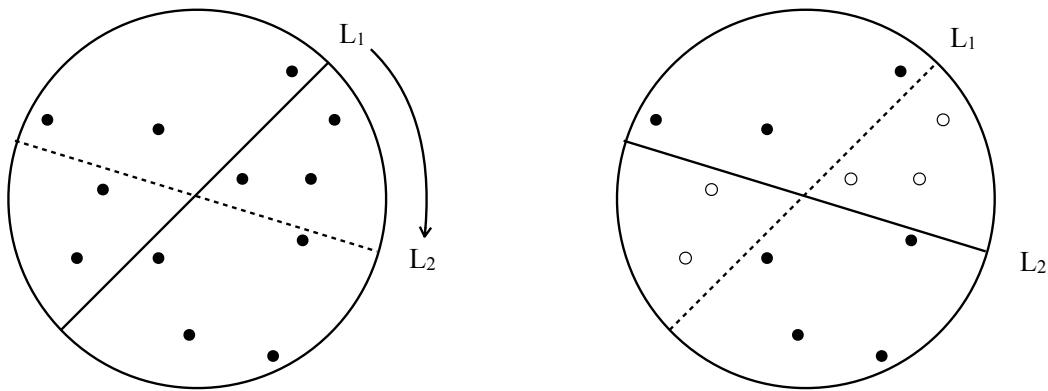
本題共有三組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	36	$t \leq 100, M \leq 100$ 。
2	32	$t \times M \leq 10^5$ 。
3	32	無額外限制。

D. 質感測試

問題描述

法國設計師「西當・普里斯」設計了一款新潮的燈具，這盞燈具的造型是一個圓盤，上面佈設了許多單顆的 LED 燈泡，開啟燈具上 LED 的方式，是利用一支與圓盤直徑等長的棒狀感應器，此感應器的中心點固定於圓盤圓心，且可順時針以圓心為固定點旋轉，旋轉時它所接觸到的 LED 皆會被開啟。以下圖為例，黑點代表未開啟的燈泡，開關由 L₁ 移動至 L₂ 後，燈泡開啟的狀況如右。



每次開啟的 LED 燈泡會落在棒狀感應器經過的兩個扇形區域。另外，我們假設旋轉開始或結束時，若感應器下方有 LED 燈泡，則它們也會被開啟。

然而，廠商在製造燈具時沒有控管材料，做出來的燈具上所佈設的 LED 亮度與顏色不一，導致整體的整體質感受到了影響。西當普里斯請廠商把每顆 LED 開啟時的呈現的質感量化為一個整數，稱為質感係數；整盞燈具的質感可由開啟的 LED 之質感係數加總而得，若一個 LED 都沒有開啟，則質感為 0。已知我們可以選擇感應器初始擺放的角度，在初始所有 LED 燈泡皆是關閉的狀態下，請寫程式幫助西當普里斯，計算這盞燈具最大可能的質感。

輸入格式

輸入共 $n + 1$ 行，第一行有一個整數 n ，表示 LED 燈泡的個數。假設圓心位於原點 $(0, 0)$ 。接下來 n 行，第 i 行有三個整數 x_i, y_i, w_i ，其中 (x_i, y_i) 表示第 i 個 LED 燈泡的座標，任二 LED 燈泡的座標皆不同且燈泡座標不會在原點上， w_i 表示該燈泡的質感係數。

輸出格式

輸出為一整數，表示此燈具的最大質感。

測資限制

- $1 \leq n \leq 3 \times 10^5$ ，且 n 為整數。
- x_i, y_i 皆為 -10^5 到 10^5 之間的整數 (包含 -10^5 與 10^5)。
- $-1000 \leq w_i \leq 1000$ ，且 w_i 為整數。
- 對所有 $1 \leq i < j \leq n$ ，滿足 $x_i \neq x_j$ 或 $y_i \neq y_j$ 。
- 對所有 $1 \leq i \leq n$ ，滿足 $x_i \neq 0$ 或 $y_i \neq 0$ 。

範例測試

Sample Input	Sample Output
4 2 1 -1 -5 1 3 -3 1 -2 0 1 4	6
7 1 -1 -1 1 3 1 0 -1 -4 0 4 5 -1 1 1 -1 -3 1 -2 2 -2	3

評分說明

本題共有四組測試題組，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	21	所有 LED 燈泡的 y 座標相同， $n \leq 100$ 。
2	35	所有 LED 燈泡的 y 座標相同， $n \leq 10^4$ 。
3	11	$n \leq 10^4$ 。
4	33	無額外限制。

E. 鋪地磚

問題描述

開心國的歡樂市正在進行都市更新，其中的愉快路旁的人行道有一項重鋪地磚的工程。愉快路附近的住戶希望鋪設好的人行道能夠展現地方特色，所以委託都市規畫的設計團隊進行人行道的設計。設計團隊在這條人行道預留了一列共 n 塊地磚的空格，打算鋪設一些顏色獨特的地磚。獨特的地磚顏色有紅色、綠色以及藍色，三種顏色的地磚分別以 r 、 g 、 b 表示。設計團隊將這 n 個空格由左而右依序編號為 1 至 n ，每一種可能的地磚鋪法可用一長度為 n 的字串 S 表示，字串的第 i 個字元 $S[i]$ 代表第 i 個空格要鋪設的地磚顏色。

為了顧及群眾的喜好，歡樂市長蒐集了愉快路附近居民的意見，並彙整成一份「美觀意見」清單，清單裡共有 k 項美觀意見。其中第 i 項美觀意見由一個「美觀顏色組合」 c_i 和「美觀加權」 w_i 組成， c_i 是一個由 rgb 三種字元所組成的字串， w_i 是一個正整數。

我們可以根據這個清單為每一種地磚鋪法評定一個「美觀總分」 t ，定義為

$$t = \sum_{i=1}^k (\text{count}(S, c_i) \times w_i)$$

其中 $\text{count}(x, y)$ 代表字串 y 在字串 x 中出現的次數，例如 $\text{count}(\text{aaabaa}, \text{aa}) = 3$ ，因為由 aaaba 中的第 1、2、5 個字元分別作為字串開頭，都能找出子字串 aa ；故 aa 在 aaaba 中共出現三次。由上式可得知總分 t 會等於所有「美觀顏色組合」 c_i 在鋪法字串 S 中出現的次數乘上該項美觀加權」的總和。

此外，某些地磚的顏色已經被附近的住戶指定，這些地磚的顏色將不能被改變。顏色指定的狀況同樣能以一長度為 n 的字串 T 表示；若第 i 個空格的地磚顏色尚未被指定，則 $T[i] = \text{x}$ ；若第 i 個空格的地磚顏色已經被指定，則 $T[i]$ 將會是 rgb 三個字元其中之一，代表此格地磚被指定的顏色。

請你寫一個程式來協助設計團隊鋪設地磚，使得美觀總分最大。

輸入格式

輸入共 $k + 2$ 行。第一行為兩個正整數 n 與 k ，以一個空白分隔，分別表示待鋪設的地磚數以及美觀意見有幾項。

接下來有 k 行，其中第 i 行 ($1 \leq i \leq k$) 有一字串 c_i 及一正整數 w_i ，兩者間以一個空白分隔，分別代表第 i 個美觀意見的「美觀顏色組合」及「美觀加權」。

最後一行有一個長度為 n 的字串 T ，由 $rgbx$ 四種字元組成，表示目前街上地磚的指定狀況。

輸出格式

輸出一個整數，代表不違反住戶要求的前提下，最大可能的「美觀總分」。

測資限制

- $1 \leq n \leq 10000$ ，且 n 為整數。
- $1 \leq k \leq 100$ ，且 k 為整數。
- 任意美觀顏色組合字串 c_i ，滿足 c_i 長度在 1 到 2000 之間(包含 1 與 2000)，且 c_i 僅由 `rgb` 三種字元組成。
- 所有美觀顏色組合字串 c_i 的長度加總小於等於 2000。
- 對所有 $1 \leq i \leq n$ ，滿足 w_i 為整數，且 $1 \leq w_i \leq 200$ 。
- 對所有 $1 \leq i < j \leq k$ ，滿足 $c_i \neq c_j$ 。
- T 的長度恰為 n ，且 T 僅由 `rgbx` 四種字元組成。

範例測試

Sample Input	Sample Output
7 1 rgbr 1 rxbxgxx	2
18 4 rr 6 gg 5 b 1 rbg 2 xxgxxxxxxxxxxxxgxx	86

評分說明

本題共有四組測試題組，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	$n \leq 8$ 。
2	21	住戶的要求字串中沒有 <code>x</code> 字元。
3	45	c_i 長度至多為 7。
4	18	無額外限制。



第一題：AI 複製機器人 (Copycat)

問題敘述

水原博士想要製作一個 AI 複製機器人，讓他可以利用複製貼上來新增文件內容。不過很可惜的是，受限於現有技術，機器人在任何時刻只能記得兩個字串 X 與 Y 。每一次啟動機器人的時候， X 與 Y 這兩個字串會分別被初始化成字母 A 與字母 B，即 $X = \text{A}$ 與 $Y = \text{B}$ 。要寫入文件時，機器人能進行四種操作：

1. 複製全部到 X ：複製現在文件中所有的文字 S ，並且替換掉目前的字串 X ，即 $X \leftarrow S$ 。
2. 複製全部到 Y ：複製現在文件中所有的文字 S ，並且替換掉目前的字串 Y ，即 $Y \leftarrow S$ 。
3. 貼上 X ：貼上你最近一次複製的 X 字串到文件末端，即 $S \leftarrow S + X$ 。
4. 貼上 Y ：貼上你最近一次複製的 Y 字串到文件末端，即 $S \leftarrow S + Y$ 。

為了展示 AI 聰明又節省能源的特色，水原博士想要讓 AI 機器人用最少的操作數，將指定字串 T 產生於一初始為空白的文件中。舉例來說，如果我們想要生成內容為 $T = \text{ABBABBB}$ 的文件，可經由以下操作序列得到它：

$$3 \rightarrow 4 \rightarrow 4 \rightarrow 1 \rightarrow 3 \rightarrow 4$$

故至多只需要 6 次操作就可以生成字串 T 。經過一番推敲，不難發現最少操作數也為 6。

為了確認 AI 機器人是否有出錯，請你先幫助博士計算所需的最少操作數。

輸入格式

給定一個僅由字母 A 與字母 B 組成的字串 T 。

輸出格式

針對所輸入的字串 T ，輸出一個數字表示最少操作次數。

測資限制

- 字串 T 的長度 $|T|$ 滿足 $1 \leq |T| \leq 2000$ 。

輸入範例 1

ABBABBB

輸出範例 1



輸入範例 2

AAAAA

輸出範例 2

8

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	43	字串 T 僅由字母 A 組成。
2	5	$ T \leq 10$ 。
3	16	$ T \leq 100$ 。
4	36	無額外限制。



第二題：惡魔島死刑犯釋放事件 (Amnesty)

問題敘述

惡魔島上的惡魔監獄關押著 n 個死刑犯，這些死刑犯都被單獨關押在單人牢房之中，犯人的編號為 1 到 n 。由於惡魔島的地形實在是太惡魔了，這些單人牢房是建造在島中央最高的山洞內，且山洞內恰有 n 間牢房。更邪惡的是，這些單人牢房之中，只有一間牢房能直接從山洞外進入。編號為 1 的死刑犯被安排在這間牢房。除了這間能直接由山洞進出的牢房以外，要抵達其他間牢房必須透過牢房與牢房之間的通道。而且，如果要從山洞外進入任何一個死刑犯 x 所在的單人牢房，在不重複經過牢房的前提下，通往死刑犯 x 的牢房路徑總是唯一的。

換句話說，所有的單人牢房在山洞內會連接成一個樹狀結構，每一個節點都代表一間單人牢房，而（編號為 1 的）根節點則代表能夠直接從山洞外進入的牢房，夠邪惡了吧。

最近，由於惡魔島為了改名為天使島，政府決定通過法律廢除所有刑事處罰。但該法律不溯及既往，所以這些死刑犯都必須以特赦來處理釋放問題。由於人數眾多，考量釋放對社會所造成影響，所以天使一般的典獄長與政府協商後，決定率先特赦一小部分的死刑犯。這個特赦的人數比例將由一個正整數參數 k 來決定。

決定了參數 k 以後，接下來得決定要釋放哪些死刑犯。典獄長會將惡魔監獄的（樹狀）結構畫在紙上，並遵循「前序走訪」方案來走訪所有的牢房，過程中依序記錄他所走訪到的犯人編號；每個犯人的編號在走訪過程中首次遇到時就會記下，且只會記錄一次。全部走訪完畢以後，典獄長再按照紙上紀錄的順序，每數到第 k 個人，就將該人釋放。亦即，紙上紀錄的依序第 $k, 2k, 3k, \dots, \lfloor \frac{n}{k} \rfloor k$ 個人會獲得釋放。

值得注意的是，雖然牢房與牢房之間的連接情形是固定的，但將其樹狀結構畫在紙上卻有許多種不同的畫法，也導致了典獄長的前序走訪方案可能會有所不同。至於典獄長會怎麼在紙上畫下惡魔監獄的結構完全是根據他當天的心情而定，我們只知道走訪的起點（樹根）為死刑犯 1 號所在的牢房。

在無法預知典獄長當天心情的情況下，你有沒有辦法判斷出，哪些死刑犯是有機會被釋放的？而哪些是不可能被釋放的？

輸入格式

輸入的第一列包含兩個正整數 n 以及 k 。第二列開始，總共有 $n - 1$ 列，每一列都有兩個正整數 u_i 與 v_i ，代表編號為 u_i 的死刑犯所在的牢房，有一條通道連接著編號為 v_i 的死刑犯所在的牢房。

輸出格式

請輸出一個長度為 n 的字串，該字串的第 i 個字元如果是 Y 代表編號為 i 的死刑犯有機會被釋放，如果是 N 則代表編號為 i 的死刑犯無論如何都不可能被釋放。

測資限制

- $1 \leq n \leq 40000$ 。



- $1 \leq k \leq n$ 。
- 對於所有 $1 \leq i \leq n$ ，都有 $1 \leq u_i, v_i \leq n$ 。
- 對於所有 $1 \leq i \leq n$ ，都有 $u_i \neq v_i$ 。
- 輸入的牢房連接情形保證形成一個樹狀結構。

輸入範例 1

5 3
1 2
1 3
1 4
1 5

輸出範例 1

YYYYY

輸入範例 2

5 3
1 2
2 3
3 4
4 5

輸出範例 2

NNYNN

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	$n \leq 10$ 。
2	11	$n \leq 500$ 。
3	17	$n \leq 2000$ 。
4	24	$n \leq 5000$ 。
5	35	無額外限制。



第三題：電路測試 (Voltage)

問題敘述

在眾多的解謎遊戲中，電路測試是其中最不好玩的遊戲，沒有一之了。

考慮一塊有 2 列 (row) 以及 $n + 1$ 行 (column) 的矩陣電路板。在該矩陣電路板上每一列與每一行的交界處，都有一個接腳，其中我們將第 r 列與第 c 行交界處的接腳以座標 (r, c) 表示。

你的手邊有一堆長度為 1 的電線，可以連接電路板上相鄰的兩個接腳（左右相鄰或上下相鄰）。相鄰接腳之間最多只可以接一條電線，不可以兩條以上並聯。在遊戲開始之前，電路板上頭，任何由連續兩列與連續兩行組成的 2×2 個接腳中間都會有一個數字，這個數字代表了圍繞著這個數字的四組相鄰接腳之間的電線連接計數。也就是說，這個電路板上會浮現總共 n 個數字。舉例來說，如果某個數字為 0，代表圍繞這個數字的四組相鄰接腳皆未被電線連結；若某個數字為 4，則代表圍繞這個數字的四組相鄰接腳，每一組都有電線連結。

遊戲的進行方式如下：有一位主持人會控制著電路板，並會提出若干次「詢問」。每一次詢問中，主持人會在盤面上標記出兩個接腳 $P = (r_1, c_1)$ 以及 $Q = (r_2, c_2)$ 。你的任務便是要計算出所有可能的電線連接方式。每個連接方式皆須滿足電路板上數字的條件，還必須讓 P 和 Q 這兩個接腳透過若干條電線連接在一起。在你回答幾次詢問後，主持人還可能會「更新」電路板上的某些數字，然後再接著提出「詢問」，以增加挑戰性。

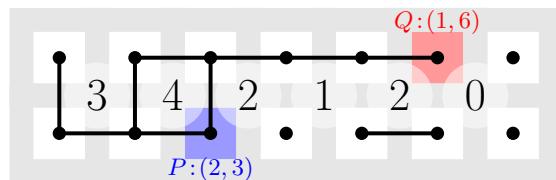


圖 1：這是一塊 2×7 大小的矩陣電路板，相鄰的 2×2 接腳中間的數字對應著周圍四條電線的連接狀況。而圖中繪製了眾多滿足條件的連接狀況之其中一種。

為了能夠快速破關，你決定要寫一個模擬器，來回答主持人所提的各個「詢問」；每個詢問的答案可能是個很大的數字，所以每一次回答時你只要輸出滿足條件的連接電線的方法數，除以 $10^9 + 7$ 的餘數即可。

輸入格式

輸入的第一列包含兩個正整數 n 與 q ，代表矩陣電路板的行數減 1 與「詢問」及「更新」的總次數。第二列包含 n 個整數， a_1, a_2, \dots, a_n ，依序代表目前電路板上由左至右出現的數字內容。

接下來的 q 列，每一列的第一個數字 op 代表該操作是更新或詢問：

- 若 $op = 1$ ，代表更新操作。接下來會有兩個以空白隔開的數字 x 與 v ，代表欲將電路板上由左至右數起來第 x 個數字修改為 v 。
- 若 $op = 2$ ，代表詢問操作。接下來會有四個以空白隔開的數字 r_1, c_1, r_2, c_2 ，代表電路板上面接腳 P 與 Q 的位置。



輸出格式

對於每一個 $op = 2$ 的詢問操作，請你計算出有幾種連接電線的方法數，除以 $10^9 + 7$ 的餘數。

測資限制

- $1 \leq n \leq 10^5$ 。
- $1 \leq q \leq 10^5$ 。
- $a_1, a_2, \dots, a_n \in \{0, 1, 2, 3, 4\}$ 。
- $op \in \{1, 2\}$ 。
- $1 \leq x \leq n$ 。
- $v \in \{0, 1, 2, 3, 4\}$ 。
- $1 \leq r_1, r_2 \leq 2$ 。
- $1 \leq c_1, c_2 \leq n + 1$ 。

輸入範例 1

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

輸出範例 1

```
1
0
```

輸入範例 2

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

輸出範例 2

```
2
```



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	18	$n \leq 5$ 且 $q \leq 100$ 。
2	19	$n \leq 10^5$ 且 $q \leq 10^5$ ，輸入保證至多只有 100 次詢問。
3	22	$n \leq 30000$ 且 $q \leq 30000$ 。
4	41	無額外限制。



第四題：無線網路服務 (Wifi)

問題敘述

快樂國政府正打算興建一條商店街作為首都旅遊的新景點。商店街預計建造於一條長度 L 的筆直街道；同時，快樂國政府還打算在商店街周圍增設 n 座基地台，使得遊客在購物時能使用免費的手機無線網路服務。目前政府已經規劃了數個可能的興建計畫，但在實行之前，必須先仔細評估各項計畫的好壞。

商店街可視作二維平面上長度為 L 、並且與 X 軸重合的線段，其中街道起點重合於原點。每座基地台的位置由平面上一點 (x, y) 表示，且 $x, y \geq 0$ 。假設站在距街道起點距離 x 處時，手機總是接收距離座標點 $(x, 0)$ 最接近的基地台訊號，定義一座基地台的「**有效區域**」為街道上接收該基地台訊號的座標點集合，「**有效長度**」為有效區域的長度。

例如，下圖中 $L = 10$ ，有三座基地台 A, B, C ，分別位於 $(1, 2), (3, 5), (6, 2)$ ，此時街道上任一點都不會接收基地台 B 的訊號，其有效長度為 0。此外，基地台 A 與 C 則以 $(3.5, 0)$ 為分界，分界左側離 A 較接近，總是接收來自 A 的訊號，右側則反之。因此 A, B, C 三座基地台之有效長度分別為 $3.5, 0, 6.5$ 。

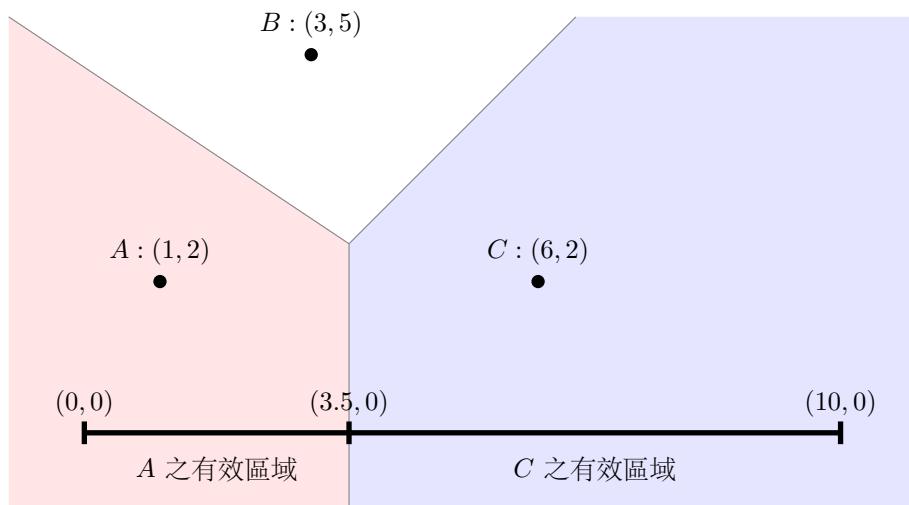


圖 2: 這個是與輸入範例 1 呼應的有效區域示意圖。

給定 n 座基地台的位置和商店街長度 L ，請你寫一支程式，找出 n 個有效長度中的最小值、中位數（第 $\lceil n/2 \rceil$ 小）及最大值。

輸入格式

第一列有兩個整數 n 及 L ，分別代表基地台的數量及商店街的長度。接下來有 n 列，其中第 i 列有兩個整數 x_i 與 y_i ，代表第 i 座基地台位於座標 (x_i, y_i) 。



輸出格式

請輸出以空白隔開的三個實數，分別代表有效長度最小值，中位數及最大值。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 $x_{\text{輸出}}$ ，實際的答案為 $x_{\text{答案}}$ ，那麼只要滿足

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

就算回答正確。

測資限制

- $1 \leq n \leq 10^6$ 。
- $1 \leq L \leq 10^9$ 。
- 對於所有 $1 \leq i \leq n$ ， $0 \leq x_i, y_i \leq 10^9$ 。
- 任兩座基地台座標皆不同，即對於所有 $i \neq j$ 皆有 $(x_i, y_i) \neq (x_j, y_j)$ 。

輸入範例 1

```
3 10
1 2
3 5
6 2
```

輸出範例 1

```
0.0 3.5 6.5
```

輸入範例 2

```
1 100
0 101
```

輸出範例 2

```
100.0 100.0 100.0
```

輸入範例 3

```
2 100
100 17
0 23
```



輸出範例 3

48.800000000 48.800000000 51.200000000

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	16	$n \leq 50$ 。
2	10	$n \leq 10^4$ 而且同一筆測試資料中所有基地台的 Y 座標相同。
3	21	$n \leq 10^4$ 。
4	53	無額外限制。



第一題：數字螢幕小鍵盤 (Numpad)

問題敘述

隨著科技越來越發達，人們追求著以更高的效率，將想要表達的話語輸入至電腦中。而這樣的動機，也發展出了許多取代傳統鍵盤以外的輸入方式，比方說螢幕小鍵盤（On-Screen Keyboard）。使用時，只需要用手指頭點擊螢幕對應的位置，就可以將對應的符號輸入至電腦中了。這類型的螢幕小鍵盤在平板電腦或手機中也有相當的貢獻。

為了讓這類型的虛擬鍵盤更加好用，你決定開發一款不需要讓手指頭離開螢幕的新型螢幕小鍵盤！有別於傳統的制式化鍵盤，這個新鍵盤的按鍵將以氣泡的形式浮動地出現在螢幕上。只要讓手指頭滑過對應的氣泡，就可以輸入對應的符號，不需要將手指移開螢幕。另一個有別於制式鍵盤的特點是，氣泡數可能會多於可輸入的符號個數，意即滑過不同的氣泡可能輸入相同的符號。

在輸入時，一開始所有氣泡會同時顯示在螢幕上，你可以點選任何一個氣泡作為起首字元。當你移動到某個氣泡 b 上時，螢幕上僅會浮現某些特定的氣泡，其他的氣泡會暫時消失。我們稱 b 與這些浮現在螢幕上的特定氣泡有所「關聯」。你可以由當前所碰到氣泡滑到與其有關聯的某個氣泡上；滑過去後，螢幕上浮現和暫時消失的氣泡又會改變。你可以依上述的規則持續滑動，直到手指移開螢幕，表示輸入結束。這個虛擬鍵盤還有一個神奇之處，那就是在任何輸入的滑動過程中，絕對不會重複碰到同一個氣泡。

要完整開發一個全新的螢幕小鍵盤不是一件容易的事情。因此，你想要先從輸入數字這點下手進行實作。如同世界上的人們使用不同的語系一般，世界上的人們也都不見得使用同一個進位制來進行計算。比方說，有些人使用 10 進位制、有些人使用 12 進位制、而有些人習慣的是 -2 進位制。事實上，你想要設計一款適合 p 進位制的人使用的數字鍵盤。

每一個設計出來的 p 進位制數字鍵盤，上頭有 n 個氣泡，每一個氣泡都有不同的編號：姑且編為 $0, 1, \dots, n - 1$ 。而每一個氣泡中則對應到一個 $0, 1, \dots, |p| - 1$ 之間的整數。欲輸入數字時，從任何一個氣泡點下去，然後沿著浮現的氣泡接連滑過去，直到把手指頭移開螢幕為止。假設你滑過了 k 個氣泡，它們上頭的數字依序是 $b_{k-1}, b_{k-2}, \dots, b_0$ ，那麼，移開指頭的那瞬間，電腦解讀的數值等於

$$(b_{k-1}b_{k-2}\cdots b_0)_p = b_{k-1}p^{k-1} + b_{k-2}p^{k-2} + \cdots + b_1p + b_0$$

這邊需要注意的是，這些數字表示的格式必須為正確格式，不能有多餘的前導零（leading zero）。即，滑出來的數字序列需要滿足 $k = 1$ 或 $b_{k-1} \neq 0$ 其中之一的條件。為了找出最佳的鍵盤設置方式，給定一個 p 進位制數字鍵盤的結構，以及一個以 p 進位表示之 m 位數數值 $threshold = (t_{m-1}t_{m-2}\cdots t_0)_p$ ，請你計算有多少種方法可以滑出至少為 $threshold$ 的整數。無論湊出來的數字是否相同，只要氣泡的組合方式不同，就視為不同的方法。這個答案可能很大，請輸出方法數除以 $10^9 + 7$ 的餘數。



輸入格式

輸入的第一列包含三個整數 n, p, m 。第二列有 m 個非負整數 $t_{m-1}, t_{m-2}, \dots, t_0$ ，為以 p 進位制表達之整數 $threshold$ 。第三列有 n 個非負整數 a_0, a_1, \dots, a_{n-1} ，其中 a_i 表示編號為 i 的氣泡上頭標注的數字為 a_i 。第四列有一個正整數 E ，表示氣泡與浮現的氣泡之間的關聯數。接下來有 E 列，其中的第 i 列有兩個非負整數 x_i, y_i ，代表當按下編號為 x_i 的氣泡時，編號為 y_i 的氣泡會浮現出來。

輸出格式

請輸出滑出至少為 $threshold$ 的方法數，除以 $10^9 + 7$ 的餘數。

測資限制

- $1 \leq n \leq 10^5$ 。
- $-999 \leq p \leq 999$ ，且 $2 \leq |p|$ 。
- $1 \leq m \leq 100$ 。
- $t_{m-1} \neq 0$ 或 $m = 1$ 其中一者成立，即輸入的 $threshold$ 保證沒有多餘的前導零。
- $0 \leq E \leq 5 \times 10^5$ 。
- 對於所有 i ，皆有 $0 \leq x_i, y_i < n$ 而且 $x_i \neq y_i$ 。
- 對於所有 $i \neq j$ ，皆有 $(x_i, y_i) \neq (x_j, y_j)$ 。

輸入範例 1

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

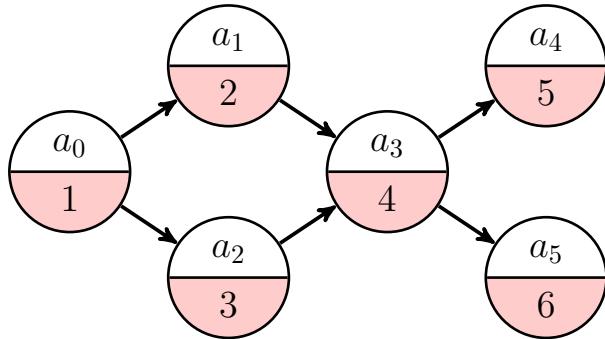
輸出範例 1

6



輸入範例 1 的說明

這六種滑數字的方法其數值分別是：1245、1246、1345、1346、345 以及 346。



輸入範例 2

```
6 -10 3
1 0 1
0 0 1 0 0 1
6
0 1
0 2
1 3
2 3
3 4
3 5
```

輸出範例 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	11	$n \leq 10$ 、 $E \leq n(n - 1)/2$ 、 $m \leq n$
2	18	$n \leq 5000$ 、任兩個氣泡之間至多只有一種方法，從一個氣泡開始並結束在另一個氣泡。
3	30	$2 \leq p \leq 999$ 。
4	41	無額外限制。



第二題：觀光 (Sightseeing)

問題敘述

ACM 王國有 n 個城鎮，這些城鎮編號為 $1, 2, \dots, n$ 。其中某些城鎮之間建有**雙向通行**的公路。已知任兩個城鎮之間都可以經過一連串公路互相來往，而且對於任一城鎮，若想不重複經過某些公路抵達任何另一個城鎮，其路線是唯一的。換句話說，整個 ACM 王國其城鎮之間的公路連結狀況為一個樹狀結構。

有鑑於城鎮規模大小不同，每天造訪該城鎮的遊客人數也不盡相同。根據歷史資料，ACM 王國的政府掌握了遊客數的數據資料：編號為 i 的城鎮，其一整年下來造訪該地的期望旅客數為 A_i 人。現在 ACM 王國的政府想要推動大旅行計劃。這個大旅行計劃，其目的是希望平衡各城鎮間的觀光收益，打算將各個城鎮的觀光人數，由當前的 A_i 人調整為 B_i 人。因為僅是在各城鎮間調動遊客人數，全國的觀光人潮在大規模移動之後，並不會因此增加或就此減少，所以此大旅行計劃保證了 $A_1 + A_2 + \dots + A_n = B_1 + B_2 + \dots + B_n$ 。

至於具體上，要如何調整觀光人數呢？政府打算與運輸公司簽訂了一項 BOT（建造-營運-轉移）案，簽下這份合約的運輸公司，會從觀光人口過剩的地方，安排計程車、遊覽車、三輪車或是人力車等任何方式，將觀光人潮半推半就地送至其他城鎮觀光。而運輸公司需要花費的成本，可以定義為將所有觀光客在城鎮間移動距離之總和。

最小成本其實不難計算，因此被許多運輸公司認定這個 BOT 案件沒有太多利潤，興趣缺缺。ACM 王國的政府為了增添廠商意願，決定**補助**一些機票錢來引誘運輸公司前來簽約。具體補助方案如下：

首先，政府會事先公布給運輸公司 q 對相異城鎮配對 $(x_1, y_1), (x_2, y_2), \dots, (x_q, y_q)$ 。接下來，運輸公司可以在與政府簽約時，選定一個**非負整數** k 。然後，政府會隨意從 q 對城鎮配對中選定其中一對城鎮 (X, Y) ，並且提供運輸公司 k 張**免費**單人單程機票。然而，運輸公司在領取這些機票時，必須要決定這 k 張機票是「從城鎮 X 飛往城鎮 Y 」還是「從城鎮 Y 飛往城鎮 X 」。這 k 張機票必須有相同的出發城鎮與目的地城鎮。每一張機票可以在任一時刻被使用，但是政府要求這 k 張機票**都必須在這一整年之中被用完**。為了最小化成本，運輸公司此時會根據政府選定的城鎮配對，在「 k 張 $X \rightarrow Y$ 機票」與「 k 張 $Y \rightarrow X$ 機票」兩種選擇之中，選取達成**運輸觀光客且用掉所有機票**之最少成本所對應的機票。

請注意，這些機票不見得總是得給觀光客使用，你可以假設每一個城鎮都有足夠多的居民。為了消耗機票，你其實可以讓這些居民搭上飛機，然後叫計程車再把他們載回居住地的。只不過運輸公司必須負擔將居民送回居住地的成本。

現在，政府已經釋出了標案中的 q 對相異城鎮配對。你身為樂樂運輸公司的頭號員工，任務是要找到一個 k 值來最大化「政府在最差情形下選擇的城鎮配對，運輸公司能夠省下來的費用」，並且輸出在此時公司可以因此**省下多少錢**。若有多個 k 值可選，則輸出最小的 k 值。



輸入格式

輸入的第一列有兩個正整數 n, q ，以一個空白隔開。

接下來有 $n - 1$ 列，每一列分別有三個正整數 u_i, v_i, w_i ，代表城鎮 u_i 與城鎮 v_i 之間有一條長度為 w_i 的公路。

接下來有 n 列，每一列有兩個非負整數 A_i 與 B_i ，代表城鎮 i 目前的觀光人數為 A_i 、目標之觀光人數為 B_i 。

接下來有 q 列，每一列有兩個正整數 x_i, y_i ，代表政府事先公布的相異城鎮配對。

輸出格式

請輸出兩個整數：樂樂運輸公司在簽約時選定的 k 值，以及此時藉由政府補助 k 張機票後可以省下的成本。

測資限制

- $1 \leq n \leq 10^6$ 。
- $1 \leq q \leq 10^6$ 。
- $1 \leq u_i, v_i \leq n$ ， $u_i \neq v_i$ ，而且輸入之公路保證形成樹狀結構。
- $1 \leq w_i \leq 1000$ 。
- $0 \leq A_i, B_i \leq 1000$ 。
- $A_1 + A_2 + \dots + A_n = B_1 + B_2 + \dots + B_n$ 。
- $1 \leq x_i, y_i \leq n$ ， $x_i \neq y_i$ 。
- 對所有的 $i \neq j$ 皆有 $(x_i, y_i) \neq (x_j, y_j)$ 。

輸入範例 1

```
6 2
1 2 2
2 3 1
3 4 3
4 5 2
5 6 1
0 12
32 0
10 0
0 20
0 13
```



3 0
1 4
3 6

輸出範例 1

20 40

輸入範例 2

3 2
1 2 2
2 3 3
50 0
0 80
30 0
1 2
2 3

輸出範例 2

36 72

範例 2 的說明

範例 2 中，一開始沒有機票補助時的總成本是 $50 \times 2 + 30 \times 3 = 190$ 。我們選取 $k = 36$ 後，若政府選擇 $(X, Y) = (1, 2)$ ，則我們選擇機票方向為 $1 \rightarrow 2$ ，這讓我們省下 $36 \times 2 = 72$ 的成本；若政府選擇 $(X, Y) = (2, 3)$ ，則我們選擇機票方向為 $3 \rightarrow 2$ ，這讓我們省下 $30 \times 3 - 6 \times 3 = 72$ 的成本（多出來的 6 張機票必須指派 6 位 3 號城鎮的居民消耗掉，再讓他們每人花 3 單位的成本送回家）。不難發現這個 k 是運輸公司的最佳選擇。好比說，若 $k = 35$ ，則政府選擇 $(X, Y) = (1, 2)$ 時只能省 $35 \times 2 = 70$ 的成本；若 $k = 37$ ，則政府選擇 $(X, Y) = (2, 3)$ 時只能省下 $30 \times 3 - 7 \times 3 = 69$ 的成本。



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	7	$q = 1$ ，且 (x_1, y_1) 是直接以某條公路相連的城鎮配對。
2	16	$n, q \leq 500$ ，且總遊客數 ≤ 500 。
3	32	$n \leq 10^5$ ， $q = 1$ ，且所有城鎮連接成一條鏈（path）。而 (x_1, y_1) 這對城鎮是鏈的兩端。
4	45	無額外限制。



第三題：淨灘問題 (Cleansing)

問題敘述

在一片偌大的沙灘上，有著許多垃圾。沙灘上不只可以舉辦排球比賽，也可以舉辦一邊寫程式一邊享受踢球樂趣的**沙灘踢歪程式設計競賽**。為了舉辦一年一度的沙灘踢歪程式設計競賽，首先要做的事情是將舉辦比賽所需要的一塊正方形範圍清理乾淨。

這個沙灘上可以被劃分成 $n \times 2n$ 個方格區域，為了方便編號起見，左上角的方格定義為 $(1, 1)$ ，右下角的方格定義為 $(n, 2n)$ 。在分析空拍圖以後，你得到一個 $n \times 2n$ 大小的表格 $A[1..n, 1..2n]$ ，其中 $A[i, j]$ 代表了定義在 (i, j) 的方格區域裡面的垃圾量。舉辦程式設計比賽，需要一塊正方形的 $n \times n$ 大小的沙灘區域。換句話說，在這片沙灘上，你恰好有 $n + 1$ 種不同的選擇。要清理出一塊適合舉辦比賽的正方形區域，最重要的任務就是要將該區域內的垃圾全部清運走。只要把所有垃圾**搬移至比賽舉辦範圍外**，就算是清理完畢了。

至於要如何清運垃圾，則有兩種不同的管道可以同時使用：出動人力清運垃圾、以及出動一次性直升機清運垃圾。

- 採用人工清運垃圾時，將 1 單位的垃圾搬運至**四個方向**的相鄰位置，需要 c 元的花費。換句話說，若要將 1 單位的垃圾從 (x_1, y_1) 處搬移至 (x_2, y_2) 處，需要花費 $c(|x_1 - x_2| + |y_1 - y_2|)$ 元。這裡的 (x_2, y_2) 方格甚至可以是整個沙灘以外的區域。
- 除了以人工方式清運垃圾以外，也可以選擇**出動一次**直升機，將所有堆積在特定方格的垃圾一口氣全部清運走。清運成本是每單位 k 元。

值得注意的是，若清運過程中出動了直升機，則必須先將欲讓直升機載走的垃圾全部集中在一個方格內，再由直升機一口氣清運走。剩餘的垃圾必須要以人工方式搬運至範圍外。

在選定比賽辦理區域之前，你想要評估這 $n + 1$ 種不同的舉辦位置，其清運所有垃圾需要之最小花費為何。請撰寫一個程式輸出每種情形需要的花費。

輸入格式

輸入的第一列包含三個整數 n, c, k ，以一個空白隔開。第二列開始的 n 列，每一列有 $2n$ 個非負整數。第 i 列的第 j 個數字代表數字 $A[i, j]$ 。

輸出格式

請輸出 $n + 1$ 個以一個空白隔開的數字，依序代表由左至右，清理每一個 $n \times n$ 正方形範圍內的垃圾所需要的最小花費。



測資限制

- $1 \leq n \leq 500$ 。
- $1 \leq c \leq 10^4$ 。
- $0 \leq k \leq 10^4$ 。
- 對於所有的 $A[i, j]$ 皆有 $0 \leq A[i, j] \leq 10^4$ 。

輸入範例 1

```
5 1 0
1 1 1 1 1 0 0 0 0 0
1 1 1 1 1 0 0 0 0 0
1 1 1 1 1 0 0 0 0 0
1 1 1 1 1 0 0 0 0 0
1 1 1 1 1 0 0 0 0 0
```

輸出範例 1

```
28 23 16 9 4 0
```

輸入範例 2

```
3 10 1
0 0 0 0 0 0
0 0 0 1 5 0
1 0 0 0 0 1
```

輸出範例 2

```
1 1 16 25
```



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	17	$c = 1$ 且 $k = 10^4$ 。
2	5	$n \leq 30$ 。
3	23	$n \leq 100$ 。
4	41	$n \leq 300$ 且 $k = 0$ 。
5	14	無額外限制。



第四題：數列搜尋問題 (Sequence)

問題敘述

小明在練習費氏數列時學到另一個數列 a ，其定義如下：

$$\begin{cases} a_0 &= 1 \\ a_1 &= 1 \\ a_{2i+1} &= a_i \\ a_{2i+2} &= a_i + a_{i+1} \end{cases}$$

這一個數列有許多有趣的特性，小明決定要把這個數列的每一項依序計算出來，然後節錄其中連續的一個序列片段。小明最終將得到的（節錄後的）序列片段記錄為 $r = \langle r_1, r_2, \dots, r_n \rangle$ 。

過了十秒鐘以後，小明意識到他忘記這個片段出現在序列中的位置了。你能幫幫小明找回該序列片段對應到原本數列 a 中第一次出現的位置嗎？如果這個片段不可能出現在數列 a 中，請輸出 -1 。

輸入格式

第一列有一個正整數 n 。第二列有 n 個正整數 r_1, r_2, \dots, r_n ，數字之間以空白隔開。

輸出格式

請輸出第一次出現的位置，除以 $10^9 + 7$ 的餘數。如果該片段不可能出現在數列 a 中，請輸出 -1 。

測資限制

- $2 \leq n \leq 10^6$ 。
- $1 \leq r_1, r_2, \dots, r_n \leq 10^{15}$ 。

輸入範例 1

```
3
2 3 1
```



輸出範例 1

5

輸入範例 2

7
1 1 2 3 5 8 13

輸出範例 2

-1

輸入範例 3

3
5 2 5

輸出範例 3

10

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	10	$n = 2$ 且 $r_i \leq 22$ 。
2	33	$n \leq 20$ 且 $r_i \leq 10^6$ 。
3	57	無額外限制。



第一題：面板亮點 (Pixel) [此題為 Output Only]

問題敘述

「能力」有限公司是一家製作超大面板的公司，他們生產面板的尺寸為高 n 個像素、寬 n 個像素，但是在面板上的其中 k 個像素是亮點，無法正確顯示該像素顏色。由於下游廠商希望面板上沒有亮點，能力公司打算根據面板上面的亮點分布來裁切面板，以滿足下游廠商的需求：每一片面板從原本高 n 個像素、寬 n 個像素，裁切成若干個大小不一的正方形面板，而且上頭沒有任何亮點。

為了避免浪費材料，小明已經用**粉筆**在板子上描繪出了 m 個正方形，正方形的每一個邊都與原本面板的邊平行或垂直。這些正方形彼此不重疊，而且剛好涵蓋了所有非亮點的像素。此外，小明也在每一個描繪出的正方形範圍內，選取一個像素，使用**麥克筆**在背後做標記並且寫上編號，以便裁切後識別使用。

可惜的是，一晚狂風暴雨之後，小明在面板上描繪的 m 個正方形輪廓通通消失了。只剩下用來標記的 m 個像素點。小明發現粉筆痕跡通通消失了以後，腦中不斷驚恐地想著生氣的上司可能會脫口而出的台詞：「不准浪費太多面板！不然等一下裁面板的時候連你一起裁掉好了。」

為了避免小明從能力有限公司被迫離開，加入另一間「人生有限公司」，或是「抄能力有限公司」，請你幫幫小明在儘量不浪費面板的情況下，切出 m 個不重疊的正方形面板，使得每一塊切出來的面板背後恰有一個標記，而且被浪費掉的面板越少越好。裁切的方式和原本相同，即切出的正方形每邊都必須與原本面板的邊平行或垂直。

本題為一 output only 的任務，並且會部分給分。你將會拿到 10 個輸入檔，說明每一個超大面板的狀況。對於每一個輸入檔，你應該繳交一個輸出檔，該檔案描述一組裁切方式。對於一個正確的面板裁切方案，你的成績將依照面板的總面積做相應的評分。

輸入格式

每一個輸入檔之格式如下：

輸入的第一列有三個正整數 n, k, m ，以一個空白隔開。

接下來有 k 列，每一列有兩個正整數 x_i, y_i ，代表亮點座標。

接下來有 m 列，每一列有兩個正整數 p_i, q_i ，代表第 i 個正方形區域中，小明做標記的像素座標。



輸出格式

每一個輸出檔必須遵循以下格式：

輸入的第一列包含一個正整數 m 。

接下來有 m 列，每一列有三個正整數 a_i, b_i, s ，代表第 i 個正方形範圍內的最小 X 座標與最小 Y 座標、且正方形之邊長為 s 。正整數之間以一個空白隔開。

測資限制

- $5 \leq n \leq 10^5$ 。
- $0 \leq k \leq \min(3 \times 10^5, n^2)$ 。
- $1 \leq m \leq \min(10^5, n^2)$ 。
- $1 \leq x_i, y_i, p_i, q_i, a_i, b_i \leq n$ 。
- 所有單一輸入檔內提及之像素座標皆不相同。
- 所有輸入保證存在一種裁切正方形的方式，涵蓋了所有非亮點像素。

輸入範例 1

```
5 0 1
2 3
```

輸出範例 1

```
1
1 1 5
```

輸入範例 2

```
5 6 5
1 4
1 5
2 1
2 4
2 5
5 4
1 1
2 2
3 3
```



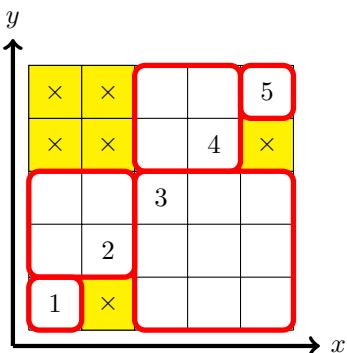
4 4
5 5

輸出範例 2

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

範例 2 的說明

如下圖所示，黃底打叉處代表亮點像素的位置。而編號處代表了小明做上標記的位置。此外，所有切出來的正方形的面板（由紅色圓角框線表示之）剛好不重疊地用完非亮點部分的面板。





評分說明

一個被視為「合法」的輸出檔，必須滿足以下所有條件：

- 輸出的正方形數量必須恰好有 m 個。
- 輸出的第 i 個正方形，必須包含第 i 個編號點。
- 輸出所有正方形內不得出現亮點。
- 輸出的所有正方形中，任兩個正方形的交集面積為 0。

對於每一個合法的輸出檔，你最高可以得到 10 分。令 $D = n^2 - k$ ，即最大可涵蓋的正方形面積。若你輸出的正方形面積總和等於 $area$ ，那麼你將根據以下規則得分：

- 若 $area < D/2$ ，得 0 分。
- 若 $D/2 \leq area < D$ ，得 $S \cdot (2 \cdot area - D)/D$ 分。
- 若 $area = D$ ，得 10 分。

其中 S 是一個隨著測試資料有不同數值的得分參數，詳見下表。本題共有 10 個測試資料檔案，條件限制如下所示。

測試資料	分數	S	n	k	m	額外輸入說明
1	10	3	20	18	42	所有輸入座標 (x, y) 皆滿足 $\min(x, y) \leq 11$ 。
2	10	3	50	41	100	所有輸入座標 (x, y) 皆滿足 $\min(x, y) \leq 21$ 。
3	10	3	500	409	1212	所有輸入座標 (x, y) 皆滿足 $\min(x, y) \leq 21$ 。
4	10	3	500	0	2871	所有輸入座標 (x, y) 皆滿足 $\min(x, y) \leq 21$ 。
5	10	6	20	137	7	所有標記的像素座標 (p_i, q_i) 皆滿足 $p_i = q_i$ 。
6	10	6	100	2418	65	所有標記的像素座標 (p_i, q_i) 皆滿足 $p_i = q_i$ 。
7	10	6	500	106492	141	所有標記的像素座標 (p_i, q_i) 皆滿足 $p_i = q_i$ 。
8	10	9	88572	0	99997	無其他限制。
9	10	9	99840	5800	90936	無其他限制。
10	10	9	71760	0	99996	無其他限制。



第二題：信號塔 (Sigtower)

問題敘述

手機在手，信號要有。踢歐埃共和國為了要迎接更新、更快、更全面的 5G 電信網路，他們設置了高達 2^n 座訊號收發使用的**通訊電塔**。這些電塔肩負著廣大的踢歐埃共和國民眾，於日常生活中最不可或缺的部分。因此，為了避免某些通訊過載，平時僅有其中的某些通訊電塔啟動「傳遞遠程訊號」的功能。我們稱啟動傳遞遠程訊號功能的電塔為**信號塔**。除了能夠傳遞遠程訊號的功能外，通訊電塔（包含信號塔）的最主要工作還有被動接收訊號，使得附近的民眾能夠使用。

通訊電塔在建置時即被賦予了編號，依序為 $0, 1, \dots, 2^n - 1$ 。遠程訊號的傳送與接收，皆會佔據通訊電塔的**頻寬**。有趣的是，踢歐埃共和國當初在賦予電塔編號時做了一個巧妙的設計：若通訊電塔 i 要傳遞遠程訊息給通訊電塔 j ，則該訊息佔據的頻寬為 $(\text{popcount}(i \oplus j))^2$ 。

其中 $\text{popcount}(x)$ 函數，是指把 x 以二進位表示的時候，數值等於 1 的位元數。此外， \oplus 則是將 i 以及 j 分別以二進位表示以後，逐一將位元以「互斥或」運算 (exclusive or) 得到的數值。

對任意一座通訊電塔 i ，其**接收總頻寬** $C(i)$ 等於來自所有信號塔的遠程訊息傳遞佔據的頻寬總和，即 $C(i) = \sum_{j \text{ 為一信號塔}} (\text{popcount}(i \oplus j))^2$ 。現在踢歐埃共和國政府想請你協助，模擬啟動傳遞遠程訊號功能的若干電塔，然後對於國境內感興趣的所有通訊電塔，分別找出接收總頻寬。由於感興趣的通訊電塔數量實在太多，你需要將所有答案雜湊之後輸出。請詳閱輸出格式。

輸入格式

輸入的第一列包含三個正整數 n, q, p 。

輸入的第二列包含一個長度為 $[2^n / 6]$ 的 base64 編碼字串 A 。解碼後可以得到一個長度至少為 2^n 的二元字串 A' ，對於所有註標 $0 \leq i \leq 2^n - 1$ ，位元 $A'[i]$ 等於 1 若且唯若編號為 i 的通訊電塔開啟了傳遞遠程訊號的功能。

輸入的第三列包含一個長度為 $[2^n / 6]$ 的 base64 編碼字串 B 。解碼後可以得到一個長度至少為 2^n 的二元字串 B' ，對於所有註標 $0 \leq i \leq 2^n - 1$ ，位元 $B'[i]$ 等於 1 若且唯若踢歐埃共和國政府想知道編號為 i 的通訊電塔的接收總頻寬。

輸出格式

令 $h = \left(\sum_{i=0}^{2^n-1} B'[i] \times C(i) \times q^i \right) \bmod p$ 。

請輸出這個整數 h 。



測資限制

- $1 \leq n \leq 23$ 。
- $2 \leq q \leq 500$ 。
- $2 \leq p \leq 10^9 + 7$ 。

輸入範例 1

```
6 123 456
WEQAAAAAAA
AAQAAGAAAA
```

輸出範例 1

180

輸入範例 2

```
5 123 456
AAAAAA
QWERTY
```

輸出範例 2

0



關於 Base64 編碼

這邊採用的是非填充 Base64 編碼。也就是說，編號為 $6i + j$ 的電塔，是由第 i 個字元解碼後寫成二進位時對應至 2^j 這一項的位元。換句話說，解碼後的二元字串，是將每一個字元解碼後以最低有效位（Least Significant Bit, LSB）在前的方式串接起來。以下是擷錄自維基百科的編碼表：

Index	Binary (LSB First)	Char									
0	000000	A	16	000010	Q	32	000001	g	48	000011	w
1	100000	B	17	100010	R	33	100001	h	49	100011	x
2	010000	C	18	010010	S	34	010001	i	50	010011	y
3	110000	D	19	110010	T	35	110001	j	51	110011	z
4	001000	E	20	001010	U	36	001001	k	52	001011	0
5	101000	F	21	101010	V	37	101001	l	53	101011	1
6	011000	G	22	011010	W	38	011001	m	54	011011	2
7	111000	H	23	111010	X	39	111001	n	55	111011	3
8	000100	I	24	000110	Y	40	000101	o	56	000111	4
9	100100	J	25	100110	Z	41	100101	p	57	100111	5
10	010100	K	26	010110	a	42	010101	q	58	010111	6
11	110100	L	27	110110	b	43	110101	r	59	110111	7
12	001100	M	28	001110	c	44	001101	s	60	001111	8
13	101100	N	29	101110	d	45	101101	t	61	101111	9
14	011100	O	30	011110	e	46	011101	u	62	011111	+
15	111100	P	31	111110	f	47	111101	v	63	111111	/

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	11	$n \leq 11$ 。
2	45	$n \leq 18$ 。
3	44	無額外限制。



第三題：機器人的路徑規劃 (Planning)

問題敘述

踢歐埃共和國的首都是由 n 條水平街道與 m 條垂直街道構成。這些街道的寬度都不盡相同，此外，街道與街道之間會形成大樓林立的街區。如果我們把這些街道放在一個二維平面上，你會發現第 i 條水平街道可以描述成兩條水平線 $y = y_{2i-1}$ 與 $y = y_{2i}$ 之間的條狀區域；而第 j 條垂直街道可以描述成兩條垂直線 $x = x_{2j-1}$ 與 $x = x_{2j}$ 之間的條狀區域。夾在相鄰的兩條水平街道與相鄰兩條垂直之間，是為一個街區，充滿了建築物。

在踢歐埃共和國政府宵夜採購部門擔任超級外送員的你，現在想要從某條街道上的某一點 (S_x, S_y) 移動到某一條街道上的另一點 (T_x, T_y) 。你可以在街道範圍中暢行無阻、也可以沿著街區的邊緣移動，但是你不能進入街區的內部。請計算起點到目標點的最短路徑長度。

輸入格式

第一列有兩個正整數 n, m 。

第二列有 $2n$ 個整數 y_1, y_2, \dots, y_{2n} 。

第三列有 $2m$ 個整數 x_1, x_2, \dots, x_{2m} 。

第四列有一個正整數 q ，代表欲查詢的路徑數量。接下來的 q 列，每一列有四個整數 S_x, S_y, T_x, T_y 。

輸出格式

請輸出 q 列。對於每一個詢問，請輸出一個實數 d ，代表該詢問所對應之最短路徑長度。

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 $d_{\text{輸出}}$ ，實際的答案為 $d_{\text{答案}}$ ，那麼只要滿足

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

就算回答正確。

測資限制

- $2 \leq n, m \leq 40$ 。
- $-10^5 \leq y_1 < y_2 < \dots < y_{2n} \leq 10^5$ 。
- $-10^5 \leq x_1 < x_2 < \dots < x_{2m} \leq 10^5$ 。
- $1 \leq q \leq 20$ 。



- $x_1 \leq S_x, T_x \leq x_{2m}$ 。
- $y_1 \leq S_y, T_y \leq y_{2n}$ 。
- 輸入之 (S_x, T_x) 與 (S_y, T_y) 保證都在某條街道上或是街道邊緣。

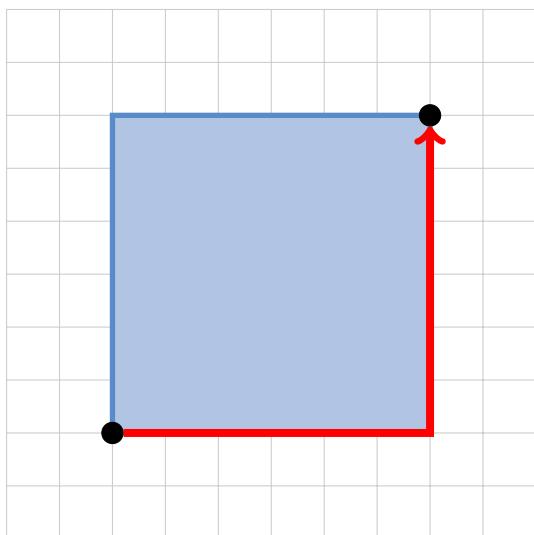
輸入範例 1

```
2 2
-5 -3 3 5
-5 -3 3 5
1
-3 -3 3 3
```

輸出範例 1

12.000000000

範例 1 的說明

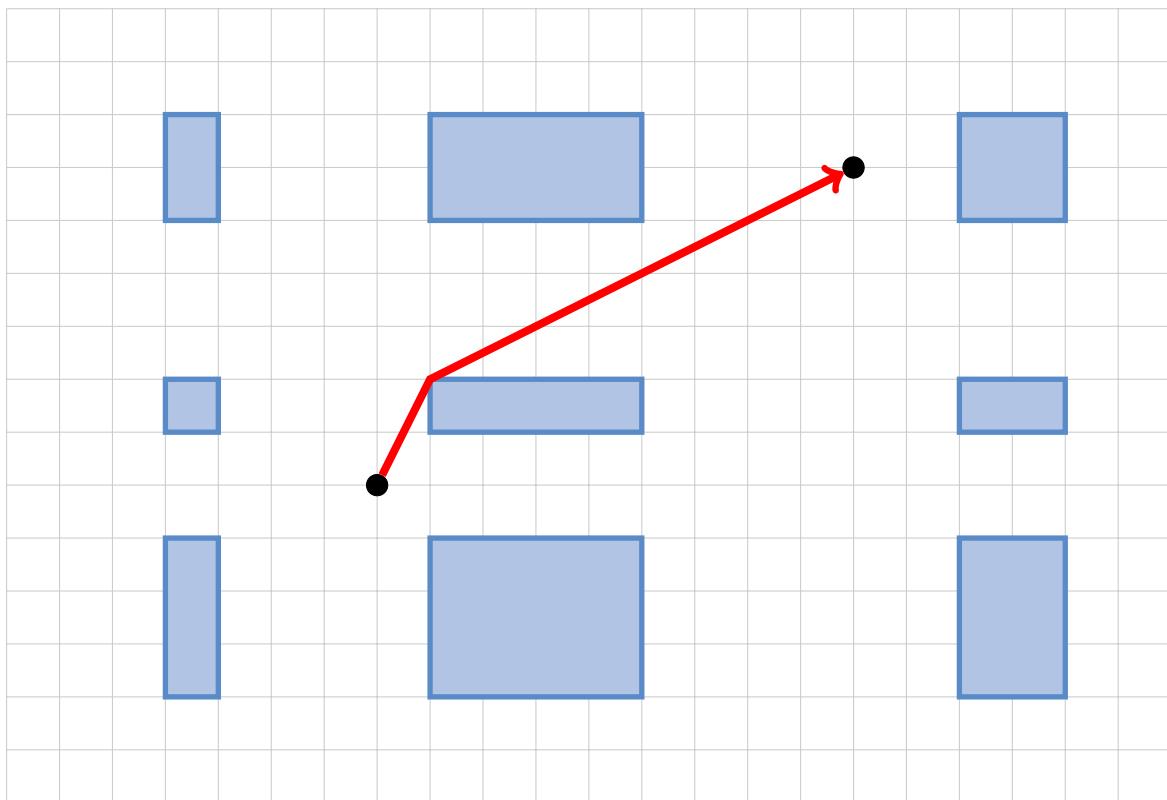


輸入範例 2

```
4 4
0 2 5 7 8 11 13 15
0 3 4 8 12 18 20 22
1
7 6 16 12
```

**輸出範例 2**

9.164271208

範例 2 的說明**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	10	$n, m \leq 8$ 且 $q \leq 5$ 。起點和終點都在街區角落。
2	16	$n, m \leq 23$ 且 $q \leq 5$ 。起點和終點都在街區角落。
3	20	$n, m \leq 23$ 且 $q \leq 5$ 。
4	26	$n, m \leq 23$ 。
5	28	無額外限制。



第四題：提款機問題 (ATM) [此題為互動題 Interactive]

問題敘述

踢歐埃共和國長久以來只有一種貨幣面額： M 元。所有的交易額度都以 M 的整數倍處理，相當地不方便。於是踢歐埃共和國的中央銀行決定要進行一些全國性的實驗，除了面值為 M 的貨幣以外，嘗試發行各種面額較小的貨幣。在實驗的過程中，央行可能會隨時發行面額為 x 的貨幣、或是回收所有面額為 x 的貨幣。

貨幣的設計本身是相當特別的，把它拿在手上其實不具有任何意義。只有在金流移動的時候（比方說付款、或是提款等等）該貨幣才會在兩方的戶頭產生轉帳的功效。而事實上，面額為 x 的一張貨幣，根據給出這張貨幣的人的選擇，它可以發揮出 $+x$ 或 $-x$ 其中之一的效果。

踢歐埃共和國有一款非常神奇的提款機。這些提款機跟自動販賣機長得很像，只不過，它的每一個品項對應到的是一綑鈔票。更仔細地說，提款機上有一些兩個一組的按鈕，每一組按鈕對應到了兩個參數 (v, q) ，當你選擇了這組按鈕，並按下其中一個按鈕時，提款機會吐出恰好 q 張面額為 v 元並且發揮了 $+v$ 功效的貨幣。相反地，如果你按下了同組的另一個按鈕，提款機會吐出恰好 q 張面額為 v 元並且發揮了 $-v$ 功效的貨幣。此外，這個提款機有個安全限制：對於任何一次交易，每一組按鈕只能擇一按鈕按下至多一次。同一個按鈕按兩次以上、或是同一組的兩個按鈕都按，都會引起保全人員的注意。隨便亂按參數是 $(0, 0)$ 的按鈕或按到不存在的面額也是不被允許的。

目前踢歐埃共和國裡面的每一台提款機，都支援同時存在 65536 組不同的按鈕參數，每一組按鈕參數編號為 $1, 2, \dots, 65536$ 。目前只有前 60 組有在使用：對於 $1 \leq i \leq 60$ ，第 i 組的按鈕參數是 $(v_i = M, q_i = 2^{i-1})$ 。對於其他的 $i > 60$ ，皆有 $(v_i = 0, q_i = 0)$ 。

身為維護該神奇提款機的工程師，央行要求你動態修改提款機提供的參數，並且協助進行一些實驗：央行可能會隨時發行面額為 x 的貨幣、回收面額為 x 的貨幣、或是詢問你某些金額要怎麼湊才能用**最少張數**湊出來。當一種新的貨幣開始發行、或一種現有貨幣被回收的時候，你可以更動提款機的按鈕參數至多 60 次。**目標是讓使用到的最大按鈕參數編號越小越好**。當然，更動後必須確保按下這個按鈕的時候，會有鈔票跑出來，也就是說回收一種貨幣時，需要把該面額貨幣使用到的參數歸零。

實作細節

你需要完成以下四個函式：

```
void initialize(int M);
void make_available(int v);
void make_unavailable(int v);
vector<set<int>> calculate_optimal_bundle_set(vector<long long> values);
```



- 評分程式一開始會呼叫 `initialize` 並傳入 M 值。
- 傳入的 M 值滿足 $1 \leq M \leq 500$ 。
- `make_available` 函式被呼叫的時候，代表央行決定增加一種貨幣面額 v 。
輸入保證 $1 \leq v < M$ 而且是目前沒有發行的面額。
- `make_unavailable` 函式被呼叫的時候，代表央行決定回收並取消面額為 v 的貨幣。
輸入保證 $1 \leq v < M$ 而且是目前已發行的面額。
- `calculate_optimal_bundle_set` 函式被呼叫的時候，代表央行想要知道你調整後的提款機，要如何湊出指定數值。
- 對於每一個 `values[i]`，`calculate_optimal_bundle_set` 回傳的第 i 個集合 S 包含一些整數。若 $j \in S$ 則代表按鈕 j 被按下，且提款機吐出恰好 q_j 張面額為 v_j 、發揮 $+v_j$ 功效的貨幣。若 $-j \in S$ 則代表按鈕 j 被按下，且提款機吐出恰好為 q_j 張面額為 v_j 、發揮 $-v_j$ 功效的貨幣。這些紙鈔的總張數必須是所有能夠湊出 `values[i]` 元最少的湊法。如果有多种最小張數湊法，回傳其中的任何一種都可以。
- 對於每一個 `values[i]`，你可以假設必定存在一種使用當前央行發行的貨幣，湊出恰好 `values[i]` 元的方法。
- 輸入保證 $0 \leq \text{values}[i] \leq 10^{18}$ 。
- 對於每一筆測試資料，央行會呼叫 `make_(un)available` 函式合計至多 10^5 次。
- 對於每一筆測試資料，央行至多會呼叫 30 次 `calculate_optimal_bundle_set` 函式，且所有 `values` 的長度加起來不超過 10^5 。

你的程式可以呼叫以下函式：

```
void set_bundle(int id, int v, long long q);
```

- `set_bundle` 函式會設定提款機編號為 id 的按鈕參數組，將這組按鈕參數改為 (v, q) 。
- 傳入之參數 id 必須滿足 $1 \leq id \leq 65536$ 。
- 傳入之參數 v 必須是目前央行發行中的貨幣面額，或是 0。
- 傳入之參數 q 必須滿足 $0 \leq q < 2^{60}$ ，且 $v = 0$ 時 q 必須也等於 0、 $v > 0$ 時 q 必須也大於 0。
- 在 `calculate_optimal_bundle_set` 函式執行的時候，不得呼叫 `set_bundle` 函式；在每次 `make_(un)available` 函式執行時，允許呼叫 `set_bundle` 函式至多 60 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**。



互動範例

考慮以下的測試資料： $M = 10$ 。

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>initialize(10)</code> 。	呼叫 <code>set_bundle(61, 3, 1)</code> 。 呼叫 <code>set_bundle(62, 3, 2)</code> 。
呼叫 <code>make_available(3)</code> 。	回傳 <code>[{62}, {-61, 1}, {-62, -61, 2}]</code> 。
呼叫 <code>calculate_optimal_bundle_set([6, 7, 11])</code> 。	

評分說明

對於一筆測試資料，假設你呼叫 `set_bundle` 的過程中使用的最大參數 id 值為 x ，則得到的**分數比重** S 值如下：

- 若 $x > 60M$ ，則 $S = 0.0$ 。
- 若 $x \leq 60 + \lceil \frac{4}{3}M \rceil - \lfloor \log_2 M \rfloor$ ，則 $S = 1.0$ 。
- 若為其他情形，得分比重為：

$$S = 1 - 0.18 \left(-1 + \sqrt{\log_{\frac{4}{3}} \left(\frac{x - 60 + \lfloor \log_2 M \rfloor}{M} \right)} \right)$$

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，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	29	$M \leq 250$ ，且所有 <code>values</code> 內數值都不超過 255。
2	23	$M \leq 250$ ，且評分程式只會呼叫一次 <code>calculate_optimal_bundle_set</code> 。
3	14	$M \leq 10$ 。
4	31	$M \leq 250$ 。
5	3	無額外限制。



範例評分程式

範例評分程式以下列格式讀取輸入：

- 第 1 列： M, Q
- 第 $2 \sim Q + 1$ 列：每一列一開始有一個 op ，根據 op 決定評分程式的動作。
 - $op = 1$ ：後面緊接著 x ，此時評分程式呼叫 `make_available(x)`
 - $op = 2$ ：後面緊接著 x ，此時評分程式呼叫 `make_unavailable(x)`
 - $op = 3$ ：後面接著 $k, values_0, values_1, \dots, values_{k-1}$ 代表一組詢問。

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: $MaxId$ ，其中 $MaxId$ 表示你使用到的最大按鈕參數編號。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：

- no quota to edit: 呼叫 `set_bundle` 次數超過限制。
- button id out of range: 呼叫 `set_bundle` 的時候按鈕參數 id 超過範圍。
- invalid v: 呼叫 `set_bundle` 的時候按鈕參數 v 不滿足題目條件。
- invalid q: 呼叫 `set_bundle` 的時候按鈕參數 q 不滿足題目條件。
- (v, q) must be both zero or both non-zero: 呼叫 `set_bundle` 的時候按鈕參數不滿足題目要求。
- face value is unavailable: 呼叫 `set_bundle` 的時候按鈕參數 v 不是央行正在發行的面額。
- invalid length of returned vector: 呫叫 `calculate_optimal_bundle_set` 回傳的陣列長度不符合規定。
- button id out of range: 呫叫 `calculate_optimal_bundle_set` 回傳的某個集合使用了不合法的按鈕編號。
- id and -id cannot coexist: 呫叫 `calculate_optimal_bundle_set` 回傳的某個集合同時按下了屬於同一組的兩個按鈕。
- answer contains invalid button: 呫叫 `calculate_optimal_bundle_set` 回傳的某個集合內包含不能按的按鈕編號。
- returned set does not result in correct value: 呫叫 `calculate_optimal_bundle_set` 回傳的某個集合湊不出指定金額。
- exists unavailable face value on ATM: 呫叫 `make_unavailable` 結束以後，有一些該面額的按鈕參數沒有清除。



第一題：ID 數量計算器 (Counter)

問題敘述

有一線上遊戲的設計者，在為他的遊戲設計玩家用的 ID。這個遊戲當中的 ID 都是以 k 進位制整數來表示的。他在設計 ID 的時候有一個規則：就是將這個 ID 表示成不含前導零的 k 進位制整數時，所有的位數中至少要有一位重複的數字。而每個伺服器的 ID 都必須介於一下界 L 及一上界 H 之間(即 $L \leq \text{ID} \leq H$)，於是想先經由計算知道一個伺服器最多可以有幾個 ID 可用，請你幫他算出這個數量，即幫助他尋找伺服器允許的範圍中，有多少數字的 k 進位制表示法包含 2 個以上相同的位數。

以 $k = 10$ 進位制來說，如果 $L = (23)_{10}$ 、 $H = (46)_{10}$ ，則因為 $(23)_{10}$ 到 $(46)_{10}$ 之間有 $(33)_{10}$ 和 $(44)_{10}$ 這兩個正整數符合條件，故輸出為 2。又如以 $k = 8$ 進位制來說，如果 $L = (77)_8$ 、 $H = (101)_8$ ，則因為 $(77)_8$ 到 $(101)_8$ 之間有 $(77)_8$ 、 $(100)_8$ 、以及 $(101)_8$ 這三個正整數符合條件，故輸出為 3。

噢對了，由於符合條件的 ID 很多，因此你只要以 10 進位制輸出滿足條件的 ID 數量除以 $10^{15} + 37$ 的餘數即可。

輸入格式

輸入的第一列有三個正整數 k, n_L, n_H ，兩數字間以一個空白隔開。

第二列有 n_L 個以空白隔開的整數 $a_{n_L-1}, \dots, a_1, a_0$ ，代表給定 L 值之 k 進位制表達式 $L = (a_{n_L-1} \cdots a_1 a_0)_k$ 。

第三列有 n_H 個以空白隔開的整數 $b_{n_H-1}, \dots, b_1, b_0$ ，代表給定 H 值之 k 進位制表達式 $H = (b_{n_H-1} \cdots b_1 b_0)_k$ 。

輸出格式

請輸出滿足條件的 ID 數量除以 $10^{15} + 37$ 的餘數（以 10 進位表示之）即可。

測資限制

- $2 \leq k \leq 5000$ 。
- $1 \leq n_L \leq n_H \leq 100000$ 。
- 對於所有的 $0 \leq i < n_L - 1$ ，皆有 $0 \leq a_i < k$ 。
- $1 \leq a_{n_L-1} < k$ 。
- 對於所有的 $0 \leq i < n_H - 1$ ，皆有 $0 \leq b_i < k$ 。
- $1 \leq b_{n_H-1} < k$ 。



輸入範例 1

```
10 2 2  
2 3  
4 6
```

輸出範例 1

```
2
```

輸入範例 2

```
8 2 3  
7 7  
1 0 1
```

輸出範例 2

```
3
```

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	11	$k = 10$ ， $n_L \leq n_H \leq 9$ 。
2	23	$2 \leq k \leq 16$ ， $n_L \leq n_H \leq 16$ 。
3	29	$2 \leq k \leq 1000$ ， $n_L \leq n_H \leq 1000$ 。
4	37	無額外限制。



第二題：期望長度 (Expectation)

問題敘述

S 和 R 為兩個長度為 n 的字串，每一個字元不是 0 就是 1 。若字串 C 同時滿足下面兩個條件，那麼我們稱 C 為 S 和 R 的**共同子序列**。

1. 把 S 中零個或多個字元刪除後，將剩下的字元按原本在 S 中的順序拼接起來，可以得到字串 C 。
2. 把 R 中零個或多個字元刪除後，將剩下的字元按原本在 R 中的順序拼接起來，可以得到字串 C 。

所有可能的共同子字串中，長度最長的一個稱為**最長共同子序列**，簡寫為 $\text{LCS}(S, R)$ 。

給定 S ，已知 R 是長度為 n 的**隨機字串**（即 R 等於每一個長度為 n 的任何二元字串的機率皆為 $1/2^n$ ）。我們想要找出 S 和 R 的最長共同子序列的期望長度。舉例來說，當 $n = 3$ 而且字串 $S = 010$ ，若 $R = 010$ 則 $\text{LCS}(S, R) = 3$ ；若 $R = 111$ ，則 $\text{LCS}(S, R) = 1$ ；若 R 為其他字串則 $\text{LCS}(S, R) = 2$ 。因此期望長度為

$$\frac{1}{8} \cdot 3 + \frac{1}{8} \cdot 1 + \frac{6}{8} \cdot 2 = 2$$

輸入格式

輸入僅有一列，包含一個正整數 n ($1 \leq n \leq 32$)，以及一個由 0 、 1 字元所組成且長度為 n 個字串 S 。 n 與 S 以一空白分隔。

輸出格式

請輸出一個最簡分數 a/b ，表示 $\text{LCS}(S, R)$ 的期望長度。當分母 $b = 1$ 的時候，請以整數方式輸出。

測資限制

- $1 \leq n \leq 32$ 。



輸入範例 1

3 010

輸出範例 1

2

輸入範例 2

4 0101

輸出範例 2

11/4

輸入範例 3

5 01000

輸出範例 3

51/16

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	$n \leq 10$ 。
2	23	$n \leq 16$ 。
3	37	$n \leq 26$ 。
4	31	無額外限制。



第三題：鑿井問題 (Plurality)

問題敘述

在一個離水源地很遠的村莊中，村民們打算鑿井取水。然而經費有限，集眾人的財力也僅能集資開鑿一座井。

整個村莊可以視為二維平面上的一個區域，而每一位村民的家可以看成平面上的一個點、欲鑿之井的位置也可以看成平面上的一個點。任兩點之間的距離，是以歐幾里德距離計算之。即點 $A = (x_a, y_a)$ 與點 $B = (x_b, y_b)$ 之間的距離定義為

$$\sqrt{(x_a - x_b)^2 + (y_a - y_b)^2}.$$

每個人都希望開鑿的井離自己家愈近愈好，即對於任一村民 v ，若位置 A 離 v 的家比位置 B 來的近，則 v 會比較偏好 A ，反之(即 v 離 B 較近) 則偏好 B ，當然在兩位置離 v 的家距離相同時， v 對於這兩個位置偏好程度相同。於是村長提供了以下的決策方案：對於兩個位置 A 與 B ，若偏好 A 位置的人數**嚴格大於**偏好 B 位置的人數，則定義 A 優於 B 。希望找出任何一個可能的位置 P ，使得任何其他位置都「不優於」 P 。

輸入格式

輸入的第一列有一個正整數 T ，代表此輸入檔案中的測試資料筆數。

對於同一個輸入檔案中的每一筆測試資料，第一列有一個正整數 n ，代表村民的數量。接下來的 n 列，第 i 列有兩個整數 x_i, y_i 以一個空白隔開，代表第 i 位村民的家的位置。

輸出格式

若存在滿足題目條件的位置 P ，請輸出兩個實數 x_P, y_P ，以一個空白隔開，表示 P 的座標。若有多個可能的位置，輸出任何一個都視為正確。若該位置 P 不存在，請輸出 \times 代表無解。

輸出的座標值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 $d_{\text{輸出}}$ ，實際的答案為 $d_{\text{答案}}$ ，那麼只要滿足

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

就算回答正確。



測資限制

- $1 \leq T \leq 5$ 。
- $1 \leq n \leq 10^5$ 。
- 對於所有的 $1 \leq i \leq n$ ，皆有 $-10^7 \leq x_i, y_i \leq 10^7$ 。
- 同一筆測試資料內，對於所有的 $i \neq j$ 皆有 $(x_i, y_i) \neq (x_j, y_j)$ 。

輸入範例

```

2
3
5 1
7 1
9 1
3
0 0
20 20
40 0
  
```

輸出範例

```

7.000000 1.000000
x
  
```

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	10	所有村民的家都擁有相同的 y 座標。
2	24	輸入保證存在鑿井的位置 P 、且保證有答案在至少一位村民家。
3	21	n 是奇數。
4	6	n 是偶數，且 $n \leq 10$ 。
5	39	無額外限制。



第四題：咳嗽問題 (Coughing)

問題敘述

當今有一種很令人感冒的感冒病毒，當病毒量在人體內累積到達一定程度的時候，就會引起咳嗽。但根據每一個人體質的不同，引起咳嗽時所需要的病毒量，也就是**咳嗽閾值**都不盡相同。一但某個人體內病毒量大於或等於他的咳嗽閾值，這個人就會開始咳嗽，並且他會將病毒散播給鄰居。

值得注意的是，假若此人有 d_i 個鄰居，則他的咳嗽閾值恰好也為 d_i 。這個人在咳完嗽以後，體內的病毒量會減少恰好 d_i 單位，而且每一位鄰居的體內病毒量都會因此而 +1。這個人會重複咳嗽，直到他體內的病毒減少至小於 d_i 為止。

踢歐埃村莊是一個祥和平靜的村莊，裡頭住著 n 個人，編號為 $1, 2, \dots, n$ 。人與人之間的鄰居關係說密切沒那麼密切，說疏遠倒也都還相互聯絡得到。更具體地說，人與人之間的關係形成了一個**樹狀結構**：對於任意的居民 i ，他若想要到居民 j 的家玩，不重複地經過鄰居家的拜訪路線總是唯一的。

現在給你這樣一個樹狀結構。已知現在整個村莊裡只有編號為 1 的人身上帶有病毒量 $k < n - 1$ 。在這種情況下，數學家幫我們證明了最後病毒一定會散播到某些人身上，而且大家都不會咳嗽。此外，無論大家咳嗽的順序如何，在大家停止咳嗽的時候，每個人身上的病毒量只有一種可能性。

請你寫一支程式，模擬病毒在村莊內傳播的情形，輸出最終每一位居民體內的病毒量。

輸入說明

輸入的第一列包含兩個整數 n, k ，以一個空白隔開。

接下來的 $n - 1$ 列描述居民與居民之間的鄰居關係。第 i 列有兩個正整數 a_i, b_i ，代表編號為 a_i 的居民與編號為 b_i 的居民互相為鄰居關係。

輸出說明

請輸出 n 個整數，依序代表每一個人在最終病毒停止傳播時，體內的病毒量。

測資限制

- $2 \leq n \leq 10^5$ 。
- $0 \leq k < n - 1$ 。
- 對於所有 $1 \leq i \leq n - 1$ 皆有 $1 \leq a_i, b_i \leq n$ 而且 $a_i \neq b_i$ 。



輸入範例

```
10 6  
2 1  
3 1  
4 1  
5 4  
6 2  
7 3  
8 4  
9 6  
10 7
```

輸出範例

```
2 0 0 2 0 1 1 0 0 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	8	$n \leq 500$ 。
2	12	人與人之間的鄰居關係形成一條鏈。
3	61	$n \leq 5000$ 。
4	19	無額外限制。

備註

請大家勤洗手、戴口罩，降低散播或感染病毒的風險唷！