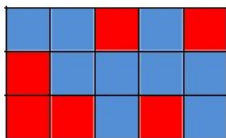


Table Coloring - 桌面著色

山姆和他的妹妹莎拉有一張由 $n \times n$ 個方格構成的桌面。他們希望所有的方格均為色紅或藍色。基於個人信仰, 他們希望每組 2×2 的方格內的紅色方格數目均為奇數(即 1 或 3)。例如, 下圖為一張 3×5 的桌面的有效的著色方法。



不幸的是昨晚已有人將部分的方格著上紅色及另外一部分著上了藍色! 山姆和莎拉想知道他們是否還可以有方法將其餘的方格著色而使整個桌面最終的顏色組合合符他們的要求的規則。如果是有可能, 他們希望知道共有多少種著色方法可以使得整個桌面的顏色在任何的 2×2 方格內都不含有偶數個紅色方格。

輸入

第一行輸入包含三個整數 n, m 及 k , 它們分別表示著桌面方格橫行及直列的數目以及最初已上了色的格子數目。而隨後的 k 行則描述了桌面方格的最初顏色狀態。詳細地說, 在第 i 行上將會有三個整數 x_i, y_i 及 c_i , 這三個數表示第 i 個最初已被人著了色方格, 其中 x_i 表示格子的行座標而 y_i 則表示格子的列座標, c_i 則為方格內最初已有的顏色。若 c_i 為 1, 則代表該方格為紅色, 而 0 則代表該方格為藍色。已知這 k 方格均為桌面上不同的方格。

輸出

在單一行輸出上輸出總共可能有的著色方法(假設為 W) 除以 10^9 的餘數(換言之, 若 W 大於或等於 10^9 時, 你的輸出應為 W 除以 10^9 後的餘數)。

條件限制

- 對於輸入資料而言, 可以肯定所有最初已著色的方格的座標均在桌面範圍內, 即可以肯定 $1 \leq x_i \leq n$ 及 $1 \leq y_i \leq m$ 。
- 在所有測試數據中, $2 \leq n, m \leq 10^6$ 且 $0 \leq k \leq 10^6$ 及 $k \leq 1000$ 。
- 在 20% 的數據中, $n, m \leq 5$ 且 $k \leq 5$ 。
- 在 50% 的數據中, $n, m \leq 5000$ 且 $k \leq 25$ 。

輸入例子	對應輸出例子
3 4 3 2 2 1 1 2 0 2 3 1	8

Find the Path

TooDee 是一個二維網格狀的土地的名稱，它就像我們的直角座標系統，其中可愛的“迪伊”就生活在那兒！迪伊是一種象蜜蜂的小動物，但他們生活在一個二維的世界內，而且也很文明。迪伊的巢是長方形的，而且它們的邊是平行於 TooDee 的座標系統的，即所有的邊均是東西方或南北方的。

由於迪伊們是非常先進的生物，在它們的世界中，有著一些固定的飛行路徑，這些飛行路徑可以被視為由一些連接有整數經度及緯度值的點的直線所組成。所有迪伊都十分遵守下列 TooDee 的飛行規例：(記著所有在 TooDee 的點都有整數的經緯座標)

- 如果你在 (X_s, Y_s) 這點上時，你只可以飛到與這點相鄰的四點去(即 $(X_s + 1, Y_s), (X_s - 1, Y_s), (X_s, Y_s + 1), (X_s, Y_s - 1)$)。
- 你不能進入任何一個迪伊巢內。
- 只有當你在一個迪伊巢的一角或一邊上時才能改變你的飛行方向。
- 你最初起飛時可選擇任何一個飛行方向。

今晚是 Deeficer (她是 TooDee 的部長官員) 的女兒的生日，因此她想盡快回家。假設她可以以每秒一個單位長度的速度來飛行。請幫她計出以最快而且合符當地飛行規例的飛行回家時所需要的時間。

輸入

第一行輸入只含有一個整數 T ，它代表檔案中包含的測試情景的數目。已知 $1 \leq T \leq 20$ 。檔案其餘的行中則包含著這 T 個測試情景。每一個情景開始前都有一行空行。

每個情景的第一行含有 Deeficer 的辦公室座標及她家的座標。每個座標均以兩個整數 X 及 Y 來表示。情景的第二行含有單一個整數 N ，它代表迪伊巢的數目。接下來的 N 行，每一行均描述一個迪伊巢的位置，而這些位置資料是以巢的兩個對角的座標來表示。你可以假設沒有任何兩個迪伊巢是相互重疊或接觸，甚至連巢角也不會互相接觸。你同時亦可以假設她的辦公室及家是兩個不同的點。每個迪伊巢的最小面積為一個單位。

輸出

對應於每個測試情景，請將其相應答案輸出到單獨一行上，這答案應為一個整數代表她最好的路徑飛回家時所需要的秒數。若她按照飛行規例是沒法飛回家的話，則請在該行上輸出“No Path”。

條件限制

- 在所有的測試情景中，所有的座標均在 $[-10^9, 10^9]$ 的範圍內，而且 $1 \leq N \leq 1000$ 。
- 在 20% 的測試數據中， $N \leq 10$ 且所有的情景中所有的座標均為非負整數且少於 100。

- 在 60% 的測試數據中, 所有的座標的絕對值均少於 1000 而且 $0 \leq N \leq 100$ 。

輸入例子	對應輸出例子
2	9
1 7 7 8	No Path
2	
2 5 3 8	
4 10 6 7	
2 1 5 4	
1	
3 1 4 3	

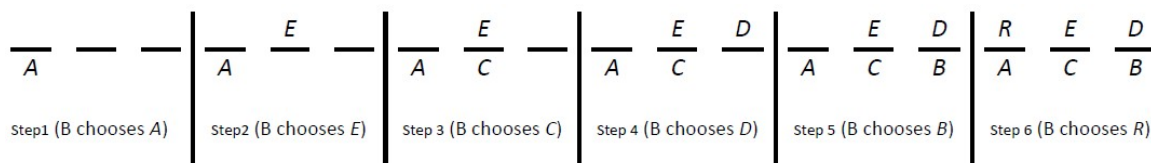
Guess My World!

“Guess My World” (簡稱 GMW) 是一個雙人遊戲, 它是一種廣泛地受青年伊朗學生們歡迎的遊戲。假設這遊戲的兩個玩家分別為 A 及 B, 開始時, A 先開始, 他將從一個大家均知道的字庫中選取一個字, 並把它記在腦中。然後他就會在一張紙上畫上與該字中的字母數目 (假設為 N) 相同的橫線段。

現在 B 開始一個字母一個字母地猜這個字是什麼。在每一輪中, B 將會選擇一個字母並把它告訴 A。而 A 則會作出以下的回應:

- 如果 B 所選的字母出現在 A 的字內, A 就會將這個字母寫在相應正確位置的橫線段上。如果整個字的所有正確字母都已寫了出來, B 就是勝利者。
- 否則若 B 所選的字母並沒有出現在 A 的字內, A 就會將該字母寫在橫線段下方最左而且未填有字母的位置上, 若橫線段下方的位置已經全部填滿而不能填入字母時 (即 B 已經選了 n 個錯誤的字母時), B 就輸了而 A 就為勝利者。這時 A 在勝出後就需要告訴 B 他所選的字是什麼。

例如假設 A 從字庫中所選定的字為 RED, 而 B 則在每輪中分別順次選擇 A, E, C, D, B 及 R 這幾個字母。在這過程中, 每步的變化見下圖。這個例中 B 是勝利者。不過, 若 B 在最後一次不選 R 而選 S 的話, 則他就輸了這場遊戲。



Aidin 是一個熱愛 GMW 遊戲的人! 他相信若果給定的字庫夠大而且包含一些相對好的字的話, A (先開始的那位玩家) 是有可用不公平的手段在遊戲中途改變他所選定的字。因為 A 所選定的字只存放在他的腦內而沒有寫在任何地方, 所以他是有可能在遊戲進行過程中, 將所選定的字更改為另一個合符遊戲至今的條件的不同的字。如上例中, 若 RED, BED, LED 及 TED 這四個字均在字庫內的話, 則 A 在第四步之後是肯定可以獲勝的。因為 A 可以將 B 接下來所選的所有字母都在寫在橫線的下面列為錯誤字母, 這是因為 B 接下來所選的字母一定會敗於字集 {RED, BED, LED, TED} 的其中一個字。而在遊戲結束時, A 可以對 B 說: “我所選的字是...”, 然後這時 A 就在以上字集內找一個餘下的字。

Aidin 甚至認為若有一個好的字庫, 玩家 A 有時一開始就可以保證自己一定會勝出! 例如若他們所玩的是含有兩個字母的字, 而在字庫內的兩個字母的字包括 {ME, MD, DE, ED, AS, IS, AI, SI} 這些字, 則 A 必定可以勝出。你可以自行試試!

給定一個字庫, Aidin 想知道 A 是否有方法在無論 B 使用什麼策略下 A 都可以保證勝出!

輸入

輸入資料中包含有多個字庫, 每個字庫均需要獨立對待及解決。

輸入的第一行含有一個整數 C , 它代表要解決的字庫的數目。其後的行中, 分別組成這 C 個字庫的輸入資料。你可以假設 $1 \leq C \leq 20$ 。

每個字庫的資料的第一行將含有一個整數 K ，它代表字庫內字的數目。其後的行中含有 K 個字，這些字以空格, tabs 或換行字符分隔開。所有字均以英文大階字母組成，且字母的數目少於 7 個。組成一個字的字母均為不同的字母，換言之在每個字內，同一個字母不會出現多於一次。

你可以假設輸入檔案的大小必定小於 500KB。

輸出

對應於每個字庫而言，輸出 “Yes” 如果 A 有一個必勝的策略 (即無論 B 如何進行這場遊戲, A 都可以勝出)。若不然的話則輸出 “No” 到一行輸出上。

記著在遊戲結束時，當 A 勝出後，A 需要在字庫選出一個並且合符 B 在過程中所選的每一個字母條件下的選定字。

條件限制

- 可以肯定在每個字庫中，字的數目不會多於 1000 個。
- 在 20% 測試數據中，每個字最多含有 3 個字母，且每個字庫最多有 100 個字。
- 在 50% 測試數據中，每個字最多含有 4 個字母，且每個字庫最多有 300 個字。

輸入例子	對應輸出例子
2 12 SI ME AND AI ARE MD AS WHEN ED IS DE HARPY	Yes No
5 A B AB AC AD	