

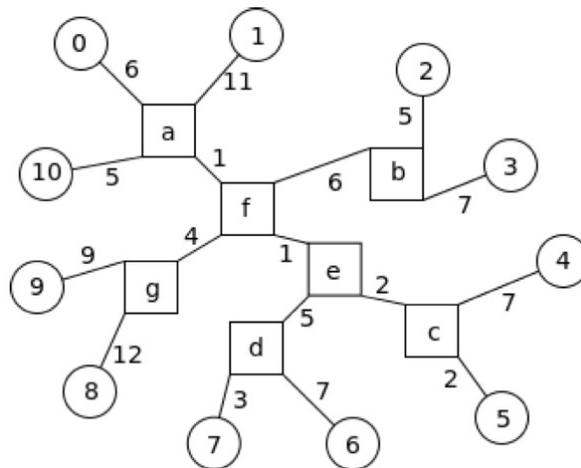
Towns 小鎮

在哈撒克斯坦內有 N 個小鎮, 其編號為 0 至 $N - 1$ 。在那裡同時亦有些未明數目的大城市。在哈撒克斯坦的這些小鎮及大城市統稱為聚居點。

在哈撒克斯坦內的所有聚居點之間都由一條雙向的公路連接著。每條公路均連接著兩個不同的聚居點。每兩個聚居點之間最多由一條公路直接連接著。對於每兩個聚居點 a 和 b 而言, 在公路不重複使用的前題下, 這裡是有唯一的一條路徑 (通過這些公路) 可以由 a 通往 b 的。

在這裡, 小鎮是一個只直接連接到別的一個聚居點的聚居點。而大城市則均直接連接到三個或三個以上的聚居點。

下圖展示了一個含有 11 個小鎮及 7 個大城市的公路網絡。小鎮以圖形表示之, 並加有整數數字的標籤。大城市則以正方形表示之, 並加有英文字母的標籤。



每條公路都有正整數長度。兩個聚居點之間的距離是由一個聚居點到另一個聚居點所經的所有公路的長度的最少總和。

對於每個大城市 C , 我們用 $r(C)$ 來表示由大城市 C 到最遠一個小鎮的距離。一個大城市是一個樞紐城市若它的 $r(C)$ 是所有大城市之中最小的。樞紐城市到最距它最遠的一個小鎮的距離以 R 表示之。換言之, R 是所有 $r(C)$ 中最小的一個。

在以上的例子中, 距離大城市 a 最遠的小鎮是 8 號小鎮, 它們之間的距離是 $r(a) = 1 + 4 + 12 = 17$ 。對於大城市 g 而言, 我們亦有 $r(g) = 17$ (距離大城市 g 最遠的其中一個小鎮是 6 號小鎮)。上例中唯一的樞紐城市是大城市 f , 其中 $r(f) = 16$ 。因此上例中的 R 是 16。

若我們將樞紐城市刪除, 則整個網絡會被分割成為若干個獨立的連接部份。一個樞紐城市是平衡的若這些分割出來的部份含有 $\lfloor N/2 \rfloor$ 個小鎮。(這裡要強調一下的是我們只計算小

鎮的數目而不需要理會大城市的數目)。注意 $\lfloor X \rfloor$ 表示不大於 x 的最大整數。

在我們的樣例中, f 是個樞紐城市。若我們把 f 剛除, 則會分出 4 個部份。這些部份分別含有小鎮: $\{0, 1, 10\}$, $\{2, 3\}$, $\{4, 5, 6, 7\}$ 及 $\{9, 8\}$ 。這裡沒有一個部份含有多於 $\lfloor 11/2 \rfloor = 5$ 個小鎮, 因此 f 是平衡的樞紐城市。

任務

最初時, 對於整個網絡系統你唯一知道的只是小鎮的數目 N 。你不知大城市的數目, 你亦不知道那些公路網絡的結構情況。你唯一可以獲取新資訊的方法是通過查詢兩個小鎮之間的距離。

你的任務是要決定:

- 在所有的子任務中, 距離 R
- 在子任務 3 至 6 中, 決定是否有平衡的樞紐城市存在

你需要編寫函數 `hubDistance`。評分系統將會在一次運行中評測多個測試點。每次評測運行時最多會有 40 個測試點。對每個測試點而言, 評測系統都會呼叫你的函數 `hubDistance` 謹一次。請確保你的函數在每次呼叫時都將所需的變量初始化。

- `hubDistance(N, sub)`
 - N : 小鎮的數目
 - sub : 子任務編號 (詳見關於子任務的說明部份)
 - 若 sub 為 1 或 2, 則你的函數可以返回 R 或 $-R$
 - 若 sub 為大於 2, 則若平衡的樞紐城市存在的話, 則函數的返回值應為 R , 否則應返回 $-R$

你的 `hubDistance` 函數可以通過呼叫 `grader` 函數 `getDistance(i, j)` 而獲得關於公路網絡的資料。這個函數將返回小鎮 i 與小 j 之間的距離。要注意的是若 i 和 j 相同的話, 則函數的返回值將是 0。而當參數不合法時, 其返回值亦會為 0。

子任務

對每個測試點而言:

- N 在 6 和 110 之間 (含)
- 兩個小鎮之間的距離在 1 和 1,000,000 之間 (含)

你可作的查詢數目是有一定限制的。這限制將因子任務不同而不同如下表所示。若你的程式嘗試使用超過制定的查詢次數, 你的程式將會數終止同時會被視為錯誤答案處理。

子任務	得分	查詢數目	找到平衡的樞紐城市	額外限制條件
1	13	$\frac{N(N-1)}{2}$	NO	沒有
2	12	$\lceil \frac{7N}{2} \rceil$	NO	沒有
3	13	$\frac{N(N-1)}{2}$	YES	沒有
4	10	$\lceil \frac{7N}{2} \rceil$	YES	每一個大城市均連接到恰恰三條公路
5	13	$5N$	YES	沒有
6	39	$\lceil \frac{7N}{2} \rceil$	YES	沒有

注: $\lceil x \rceil$ 表示大於或等於 x 的最小整數。

樣例 grader

請注意子任務的編號是輸入的一部份。樣例 grader 將跟據子任務的編號而改變它評分的方法。

樣例將輸入下面格式的文字檔 towns.in:

- 第 1 行: 子任務編號及測試點的數目
- 第 2 行: N_1 , 第一個測試點中小鎮的數目
- 隨後的 N_1 行: 在這部份的第 i 行上的第 j 個數字表示小鎮 $i - 1$ 和 $j - 1$ 之間的距離
- 隨後是下一個測試點的數據, 其格式與第一個測試點相同

對於每個測試點而言, 樣例 grader 將會在不同行上輸出函數 hubDistance 的返回數及其呼叫函數的次數。

對應於上例的輸入檔案如下:

```
1 1
11
0 17 18 20 17 12 20 16 23 20 11
17 0 23 25 22 17 25 21 28 25 16
18 23 0 12 21 16 24 20 27 24 17
20 25 12 0 23 18 26 22 29 26 19
17 22 21 23 0 9 21 17 26 23 16
12 17 16 18 9 0 16 12 21 18 11
20 25 24 26 21 16 0 10 29 26 19
16 21 20 22 17 12 10 0 25 22 15
23 28 27 29 26 21 29 25 0 21 22
20 25 24 26 23 18 26 22 21 0 19
11 16 17 19 16 11 19 15 22 19 0
```

這個格式是有別於其他一般描寫公路網絡的格式。請注意, 你是可以修改樣例 grader 的內容, 使它可以接受其他不同的輸入格式。