



Saturday, 8 May, 2010

任務名稱	突擊隊	巡邏隊	信號
時間限制	1 sec	1 sec	2 sec
記憶體限制	64 MB	64MB	64 MB
分數	100	100	100
輸入	標準輸入(鍵盤)		
輸出	標準輸出(螢幕)		

語言	編譯器版本	編譯器選項
C	gcc version 4.1.2	-m32 -lm
C++	g++ version 4.1.2	-m32 -lm
Pascal	fpc 2.0.4 for i386	-Sd -Sh

突擊隊

你是一位擁有由1到 n 個士兵的部隊司令官。為了即將到來的戰役，你計畫將這 n 個士兵分成數個突擊單位。為了凝聚士氣，每個單位將由連續序列(如 $(i, i+1, \dots, i+k)$)的士兵組成。

每個士兵 i 都有其戰鬥力評分 x_i 。原先，每個突擊單位的戰鬥力總和為 x ，其計算方式是將每位士兵的戰鬥力相加，也就是說， $x = x_i + x_{i+1} + \dots + x_{i+k}$ 。

然而，過去輝煌勝利的經驗告訴你，一個突擊單位的戰鬥力總和應該修正為：修正戰鬥力 x' ，其計算公式為 $x' = ax^2 + bx + c$ ，而 a, b, c 是已知係數($a < 0$)， x 是這個單位的原本戰鬥力。

身為一個司令官，你的任務就是將士兵們分配成數個突擊單位，確保所有單位的修正戰鬥力總和為最大值。

假設你有4個士兵， $x_1 = 2, x_2 = 2, x_3 = 3, x_4 = 4$ 。接著，藉由公式的係數改變該單位的戰鬥力，其中係數 $a = -1, b = 10, c = -20$ 。既然這樣，最好的方式是將所有士兵分成三個戰鬥單位：第一個單位包含士兵1和2、第二個單位包含士兵3、第三個單位包含士兵4。這三個單位的戰鬥力總和將分別為4、3、4，而修正後的戰鬥力總和為4、1、4。這種分配方式的總修正戰鬥力將為9，而且這是最好的分法。

輸入格式

輸入共有三列，第一列包含一個正整數 n ，也就是所有士兵總數。第二列包含3個整數 $a, b,$ 和 c ，為公式中為了調整突擊部隊戰鬥力的係數。最後一列包含 n 個整數 x_1, x_2, \dots, x_n ，以空格隔開，分別代表著第一位士兵、第二位士兵、到第 n 位士兵。

輸出格式

以單列整數形式代表可達到之最大修正後的戰鬥力。

輸入樣本

4

-1 10 -20

2 2 3 4

輸出樣本

9

限制

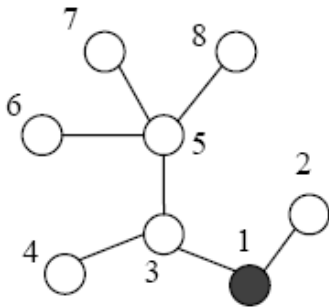
- 在 20% 的測試案例中, $n \leq 1000$;
- 在 50% 的測試案例中, $n \leq 10,000$;
- 在 100% 的測試案例中, $n \leq 1,000,000$, $-5 \leq a \leq -1$, $|b| \leq 10,000,000$, $|c| \leq 10,000,000$ and $1 \leq x_i \leq 100$.

巡邏隊

在一個城市裡，有 N 個編號從1、2、至 N 的村莊，而有 $N-1$ 條路連接著這些村莊。每條路都確實連接2個村莊，人們都可以利用這些路從任何一個村莊到達其他村莊。每條路的長度為1單位。

為了確保這個城市居民的安全，城市警察巡邏隊必須每天巡查每條路。警察局在1號村莊，所以巡邏隊必須從1號村莊出發，到最後也必須回到1號村莊。

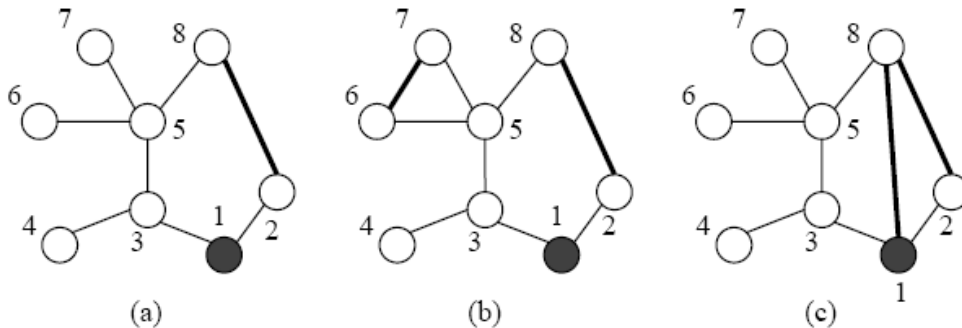
以下列擁有8個村莊的城市為例，可以看到村莊排列如同圖形，而那個黑色圓形即是1號村莊。連接著村莊的路就是那些線。要巡查所有道路，巡邏隊每天必須走過14條路。注意！每條道路巡邏隊必須走過兩遍才能完成每天的任務。



為了減少每天必須走過的總距離，這個城市計畫在村莊間興建一條新捷徑 K 。每條捷徑可以連結任兩個村莊。兩條捷徑可以在同一個村莊會合（如同下圖的c）。一條捷徑也可以是一個環狀，也就是說，連結該村莊本身。

由於經費有限，因此，捷徑 K 必須是1條或2條。而且，為了確保該城市沒有浪費金錢，巡邏隊在一天中必須走過每條捷徑一次。

以下是可以考慮的可能性。



在可能性(a)中，只興建一條捷徑，總距離為11。在可能性(b)中，興建兩條捷徑，巡邏隊必須走過10單位路線。在可能性(c)中，興建兩條捷徑，可是因為有次數規定巡邏隊需走過捷徑一次，因此總距離變成15。

請寫一個關於連結村莊間的道路與欲興建的捷徑數量的程式，並且估算捷徑的所在地讓巡邏隊每天需巡查的總距離為最短。

輸入格式

輸入的第一列包含兩個整數 N 和 K ($1 \leq K \leq 2$)。下一個 $N-1$ 列包含道路資訊，每一列都包含兩個整數 A 和 B ($1 \leq A, B \leq N$)，亦即村莊 A 和村莊 B 都有一條路可以連接。

輸出格式

你的程式輸出結果為一個整數，讓捷徑 K 被興建後，巡邏隊可以以最短的距離巡查城市。

輸入樣本 1

8 1

1 2

3 1

3 4

5 3

7 5

8 5

5 6

輸出樣本 1

11

輸入樣本 2

8 2

1 2

3 1

3 4

5 3

7 5

8 5

5 6

輸出樣本 2

10

輸入樣本 3

5 2

1 2

2 3

3 4

4 5

輸出樣本 3

6

限制

- 在 10%的測試案例中， $N \leq 1,000$ 且 $K = 1$ 。
- 在 30%的測試案例中， $K = 1$ 。
- 在 80% 的測試案例中，相鄰村莊的最大值是25。
- 在 90%的測試案例中，相鄰村莊的最大值是150。
- 在100% 的測試案例中， $3 \leq N \leq 100,000$ 且 $1 \leq K \leq 2$ 。

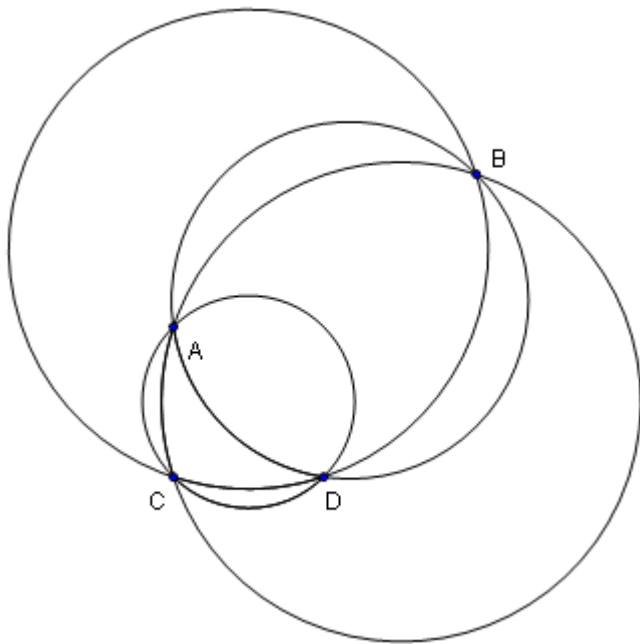
信號

一家電信公司正在北京市發展GSM網路。這個城市中有 n 個房子需要被網路所覆蓋。因為經費限制，該公司只能安裝一條天線。

為了簡化天線的設置，必須先挑選3間房子畫成一個圈圈，然後將天線設置在該圈圈的中心點。天線的範圍將會是所有房子都落在圈圈內，包含圈圈邊界上的房子都將被覆蓋。

這家公司計畫隨機挑選3間房子，所以他們想要計算在天線所設置的可能位置上，即將被覆蓋的房子間的平均數。

舉例來說，假設有4間房子A, B, C和D，所在位置如下圖所示。



如果我們選擇圈圈是ABC或BCD所串連，每個房子都將被覆蓋著。但如果我們選擇圈圈是ACD或ABD所串連，第四間房子將無法被天線的範圍所覆蓋。因此，將被覆蓋房子的平均數為 $(4 + 4 + 3 + 3) / 4 = 3.50$ 。

你的任務就是計算被信號覆蓋的房子平均數及假設房子位置。房子的位置將依據二維座標系統讓所有房子都有其座標。你要確保沒有三間房子會落在同一條線上，也沒有四間房子會落在同一個圈圈裡。

輸入格式

輸入的第一列包含一個正整數 n ，也就是房子的總數。接下來有 n 列，描述著房子的位置。當 $i \in \{1, 2, \dots, n\}$ ，房子 i 的座標是一對整數 x_i 和 y_i 在輸出的 $i+1$ 列上，由空格間隔開來。

輸出格式

輸出必須包含一個單一實數，讓房子的平均數都能被信號所覆蓋。結果的絕對誤差必須少於或等於0.01。

輸入樣本

4
0 2
4 4
0 0
2 0

輸出樣本

3.5000

限制

- 在 100% 的測試案例中，當 $i \in \{1, 2, \dots, n\}$ ，其房子 i 的座標 (x_i, y_i) 都是整數，也就是 $-1000000 \leq x_i, y_i \leq 1000000$ ，沒有三間房子會落在同一條線上，也沒有四間房子會落在同一個圈圈裡。
- 在 40% 的測試案例中， $n \leq 100$.
- 在 70% 的測試案例中， $n \leq 500$.
- 在 100% 的測試案例中， $3 \leq n \leq 1,500$