

旋轉的直線們 (rotate)

Asadullo 是一位任職於 APIO（全名為「能源與產業最佳化聯盟」：Alliance for Power and Industrial Optimization）的傑出科學家。近期，他正潛心研究著一種使用未知材料來產生能源的新方法。

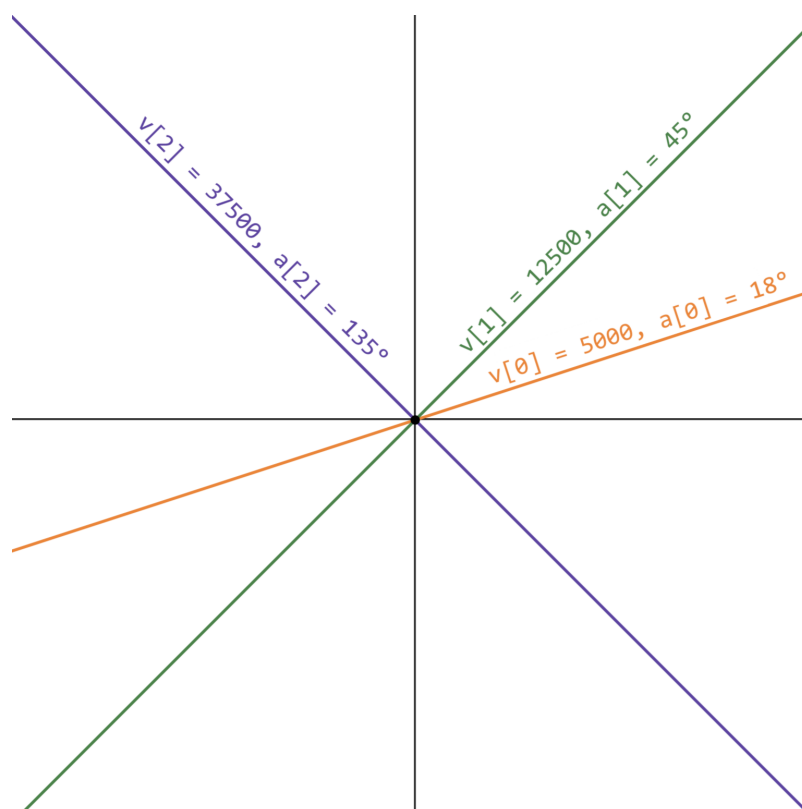
這種未知材料本身無法自行產生能量，但若使用該材料製作出數根極長的竿子，這些竿子透過彼此之間的交互作用便能產生能源。

具體來說，我們會給你 n 根竿子的資訊 $v[0], v[1], \dots, v[n-1]$ 。第 i 根竿子將以逆時針方向、相對於 x 軸的角度擺放，其角度為 $a[i] = 360 \cdot \frac{v[i]}{100000}^\circ$ 。這 n 根竿子所產生的能源效率將被定義為：

$$\sum_{i < j} \text{acute}(i, j)$$

其中 $\text{acute}(i, j)$ 代表第 i 根竿子與第 j 根竿子之間的銳角夾角。在這個題目中，我們將 90° 視為銳角。更正式地，我們定義 $\text{acute}(i, j) = \min(|v[i] - v[j]|, 50000 - |v[i] - v[j]|)$ 。換言之，能源效率等於所有兩兩竿子之間銳角大小的總和。

舉例來說，若 $v = [5000, 12500, 37500]$ ，則對應的角度為 $a = [18, 45, 135]$ ，我們將得到如下圖所示的配置：



此時， $\text{acute}(0,1) = 7500$ （即 27° ）， $\text{acute}(0,2) = 17500$ （即 63° ）， $\text{acute}(1,2) = 25000$ （即 90° ）。因此，這三根竿子的總能源效率為 $7500 + 17500 + 25000 = 50000$ 。

現在，Asadullo 想要透過調整這 n 根竿子的配置，使能源效率達到最大。然而，在調整過程中必須遵守以下幾項限制：

- 首先，由於這種材料對生物極為危險，竿子只能透過一種特殊的機械裝置進行控制式旋轉。該裝置允許一次選取多根竿子，並讓它們同步旋轉相同的角度。
- Asadullo 不希望能源效率下降。因此，每次使用裝置進行操作後，能源效率必須不低於原本的數值。
- 此外，操作裝置本身也會消耗大量能量，因此在所有操作過程中，被選取的竿子總數不得超過 2 000 000 根。

在這些限制之下，Asadullo 希望能夠以最佳方式進行調整操作，讓能源效率盡可能提高。請你撰寫一個程式，幫助 Asadullo 達成能源效率的最大化。

實作細節 (Implementation details)

你應該實作以下的函式：

```
void energy(int n, std::vector<int> v)
```

- n ：竿子的總數。
- v ：一個長度為 n 的陣列，代表竿子的資訊。
- 這個函式會被呼叫恰好一次。

在該函式內，你可以呼叫以下的函式：

```
void rotate(std::vector<int> t, int x)
```

- t ：一個有相異註標的陣列，即 $0 \leq t[i] < n$ 對所有的 i ，且 $t[i] \neq t[j]$ 對所有的 $i < j$ 。陣列 t 不一定是排好序的。
- 這個函式將陣列 t 裡面的所有竿子同時進行參數 x 的旋轉。也就是說，對於所有出現在陣列 t 中的註標 i ， $v[i]$ 會變成 $(v[i] + x) \bmod 50000$ 。
- 該函式可以被呼叫很多次。所有呼叫的陣列 t 長度總和不得超過 2 000 000。

範例 (Examples)

範例 1 (Example 1)

考慮以下呼叫：

```
energy(2, [20000, 10000])
```

其中， $v = [20000, 10000]$ 且初始能源效率等於 $20000 - 10000 = 10000$ 。其中一種可能的執行場景如下：

- 呼叫 `rotate([0, 1], 8000)`。此時 v 變為 $[28000, 18000]$ ，而能源效率保持不變。
- 呼叫 `rotate([0], 15000)`。此時 v 變為 $[43000, 18000]$ ，能源效率變為 $43000 - 18000 = 25000$ 。

可以證明對於這組輸入，25000 是可能達到的能源效率最大值。因此，Asadullo 可以在此時停止操作。

範例 2 (Example 2)

考慮以下呼叫：

```
energy(3, [5000, 12500, 37500])
```

該範例對應的圖案已在上方呈現。可以證明初始的能源效率已經是最大可能值，因此不需要進行任何操作。

限制 (Constraints)

- $2 \leq n \leq 100\,000$
- $0 \leq v[i] \leq 49\,999$ 對每個 $0 \leq i < n$
- 所有 v 中的元素不一定相異

子任務 (Subtasks)

1. (5 points) $n = 2$
2. (11 points) $v[i] < 25\,000$ 對每個 $0 \leq i < n$
3. (8 points) $n \leq 10$
4. (15 points) $n \leq 100$
5. (15 points) $n \leq 300$
6. (20 points) $n \leq 2000$
7. (26 points) 無額外限制。

範例評分程式 (Sample Grader)

範例評分程式以下列格式讀取輸入：

- line 1: n
- line 2: $v[0] \ v[1] \ \dots \ v[n-1]$

範例評分程式以下列格式輸出：

- line 1: 所有竿子最終的能源效率

此外，評分程式也會將你每一次旋轉操作細節記錄在 `log.txt` 這個檔案裡。