

第二題：土地徵收 EX (Land_EX) [此題為互動題 Interactive]

問題敘述

歪批國正在徵收土地，經由最佳化計算出的土地規劃，歪批國勢必得徵收到小 P 的土地。小 P 私底下很不爽這件事，他某天晚上偷偷逃離了歪批國，使得沒有人找得到他在哪裡。

歪批國對此非常懊惱，因為小 P 的土地有 10^{18} 公頃這麼大，但其實大多數的地區都被小 P 下毒過了，真正能用的區域只剩下一個矩形而已。如果把小 P 的土地用一格一公頃的座標表示，左下角是 $(0, 0)$ ，右上角是 $(10^9, 10^9)$ ，則這個矩形的範圍就是由左下角 (a, b) 一直到右上角 (c, d) 所劃出的範圍 $(0 \leq a < c \leq 10^9, 0 \leq b < d \leq 10^9)$ 。

而問題便是，小 P 並沒有留下任何關於可用土地範圍的資訊，所以歪批國根本無法輕易得知哪些地方能用。為了解決這個問題，科學家們研發出一個可以自由伸縮大小的矩形探測器，它可以幫助歪批國計算出一些資訊，經由探測器自動轉換數據過後，探測器會回傳「這個矩形中，可用區域的面積」。

不過這個探測器非常耗電，基於電量考量，歪批國希望探測的次數越少越好，你能夠幫忙歪批國找到矩形實際的位置（找出 a, b, c, d ）嗎？

實作細節

C/C++ 的使用者請在程式碼的首行加上「`#include "Land.h"`」，其餘詳細的範例請見該題 CMS 內的附件檔案「Land.c/Land.cpp」。

評分端會提供一個物件，在 C/C++ 中如下：

```
typedef struct rectangle {
    int a, b, c, d;
} rectangle;
```

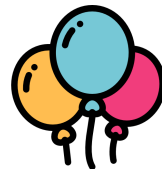
在 Java 中如下：

```
public class rectangle {
    int a, b, c, d;
}
```

你需要完成以下函式：

```
rectangle find_rectangle(int subtask);
```

- 該函式可能會在同一筆測資內被呼叫至多 10^4 次，請務必在每次呼叫時確保函式有經過重設。



- *subtask* 變數表示該筆測資隸屬的子任務編號（見評分說明）。
- 請在找到矩形的位置後，回傳一個內含矩形正確位置座標的 **rectangle** 物件。

你的程式可以呼叫以下函式：

```
long long area(int x1, int y1, int x2, int y2);
```

- 這代表你要偵測「左下角 $(x1, y1)$ 右上角 $(x2, y2)$ 矩形中可用區域的面積」，該函式會回傳你一個非負整數，代表面積。
- 基於電量考量，該函式至多只能被呼叫 128 次。
- 你傳入的整數需滿足 $0 \leq x1 < x2 \leq 10^9$ ， $0 \leq y1 < y2 \leq 10^9$ 。
- 請注意函式的回傳型態為 **long long**（在 Java 內為 **long**）。

如果不滿足上述條件、或是回傳值不符合題目要求，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： $(a, b, c, d) = (2, 2, 4, 5)$ 。

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>find_rectangle(3)</code> 。	呼叫 <code>area(0, 0, 100, 100)</code> 。
回傳 6。	呼叫 <code>area(0, 0, 5, 5)</code> 。
回傳 6。	呼叫 <code>area(0, 0, 3, 3)</code> 。
回傳 1。	呼叫 <code>area(2, 2, 4, 4)</code> 。
回傳 4。	呼叫 <code>area(2, 2, 5, 4)</code> 。
回傳 4。	呼叫 <code>area(2, 2, 4, 5)</code> 。
回傳 6。	回傳 <code>rectangle{2, 2, 4, 5}</code> 。



評分說明

對於一筆測試資料，假設在每次呼叫 `find_rectangle` 的過程中呼叫 `area` 的次數 `count` 的最大值為 x ，則得到的分數比重 S 值如下：

$$S = 2 - \frac{x}{60}$$

本題共有 1 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	100	$subtask \in \{1, 2, 3\}$ 。

範例評分程式

範例評分程式以下列格式讀取輸入：

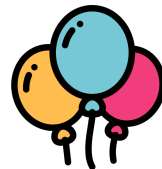
- 第 1 列： $T \ subtask$
- 第 2 ~ $T + 1$ 列： $a \ b \ c \ d$

其中 T 為測資筆數， $subtask$ 為子任務編號， a, b, c, d 如題目所述。

當然， a, b, c, d 須滿足 $0 \leq a < c \leq 10^9$ ， $0 \leq b < d \leq 10^9$ 。

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: MaxQ`，其中 $MaxQ$ 表示你詢問的最大次數。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：

- `invalid query`: 存在任何一次不合法的 `area()` 呼叫。
- `too many queries`: 存在任何一筆 `find_rectangle()` 的函式呼叫內，呼叫的 `area()` 總次數超過 128 次。
- `incorrect place`: 每當 `find_rectangle()` 函數執行結束後， a, b, c, d 內的值與當前答案不相符。



備註

在 CMS 內的附件檔案中，有一個名為「Land」的 .c/.cpp 檔案，內部已經有寫好一些東西的檔案，直接上傳該檔案並不會導致 CE，並且往後你要寫的程式都可以參考這份檔案，並在這份檔案內進行修改、編譯及執行。

請注意，在 C/C++ 的檔案內有兩行註解分別為「do not modify above」和「do not modify below」，這兩行意味著希望你盡量只更動被這兩行夾住的區域，若你對互動題並沒有很熟悉，請不要更動外面的區域，**更動外面的區域不會也不可能讓你能直接拿到更高的分數**。我們有提供 C/C++ 的使用者「`compile_c.sh/compile_cpp.sh/compile_c.bat/compile_cpp.sh`」做使用，請選手請根據自己的需求多加利用。

在本題中，你能拿到的分數可不僅限於 $[0, 100]$ 喔！請嘗試使用各種各樣的方法，盡可能的拿到最高分吧！