

F. 社部的密碼鎖

Problem ID: Locker

社部是個好地方，可以吹冷氣、睡覺、打電動。但是長期下來，Joseph 覺得這樣太糜爛了，於是他決定把社部上鎖。他選用了當今最最高科技的密碼鎖。這個密碼鎖長得很像一部計算機，上面有一些按鍵，每個按鍵上都有不同的文字像是「+1」，「-2」，「 $\times 3$ 」，「 $\div 5$ 」之類的當把這些按鍵按照一定順序按下去之後，就能解鎖。至於按鍵上那些文字的功用是當你忘記密碼的時候可以給你提示用的。如果你忘記密碼，密碼鎖的螢幕上會顯示兩個數字， a 跟 b ，你的密碼就是透過那些按鍵把 a 變成 b 。所以你會發現，其實密碼並不一定只有一組。例如， a 跟 b 分別是 0 跟 10，而按鍵有「+1」，「 $\div 2$ 」，「 $\times 2$ 」，「+5」，那麼密碼可以是「+5+5」也可以是「+5 $\times 2$ 」也可以是「+1+1+1+1+1+1+1+1+1+1」等等很多種密碼都可以成功開鎖。這個門鎖的所有運算都是整數，如果有運算後不是整數的情形，那個運算就無法被操作。

而社部裡面有很涼的冷氣跟舒服的地板，你超想進去的，所以你決定要解開密碼。

然而 Joseph 也想得到一定會有人去破解密碼，於是他事先準備了 100 個門鎖，只要一個被破解就換下一個。而你也很幸運的拿到了一份那 100 個門鎖的資訊。於是你決定事先破解完 100 個門鎖，這樣不管換哪個門鎖你都不怕了。

門鎖上的按鈕操作可能有以下幾種：

- 「+X」：把原本的數字加 x 。
- 「-X」：把原本的數字減 x 。
- 「 $\times X$ 」：把原本的數字乘 x 。
- 「 $\div X$ 」：把原本的數字除 x ，答案必須是整數，否則無法操作。
- 「>>」：把原本的數字右移一位，原本是 123，操作完變成 12，原本是 -234，操作完變成 -23。
- 「X」：把原本的數字左移一位補上 x ，例如原本是 123， x 是 5，操作完變 1235，原本是 -234， x 是 1，操作完變 -2341。
- 「+/-」：把原本的數字負的轉成正的，正的轉成負的，原本是 -123 變成 123，原本是 123 變成 -123，原本是 0 變成 0。

輸入說明

測資全部在同一個輸入檔每筆測資間以一個空行隔開，每筆測資都有 3 行每筆測資第一行都是「Case #x:」不含引號， x 是該筆測資編號，從 1 開始每筆測資第二行有兩個整數 a ， b 以空白隔開，代表要把 a 變成 b 每筆測資第三行剛開始會有一個正整數，代表有幾個按鈕接下來每

個按鈕會以上面題目敘述的格式表示，每個按鈕間都有一個空格根據按鈕的輸入順序，也會是按鈕的編號，編號從 1 開始詳見範例

輸出說明

全部輸出請放在同一個檔案每個測資輸出一行每一行剛開始是一個整數，代表測資編號，再來有一個空格，接下來是一個整數 n ，代表作了幾次操作，再來有一個空格再來就是操作方法，請輸出每個操作，每個操作間以一個空白隔開輸出不一定要按照測資順序，也就是先輸出第 2 筆測資再輸出第 1 筆測資是可以的你也可以選擇只破解一些密碼鎖，那你只要輸出你要破解的那些就行詳見範例

實作說明

本題為 Output Only，只需要上傳輸出檔即可，不需要上傳程式碼。輸入檔在本題附件，請自行下載，只有一個檔案。附件裡還有一份程式碼是檔案操作的模板，在那份程式碼中，所有的 `stdin` (`cin`, `scanf`) 皆會從 `input.txt` 檔案讀取，所有的 `stdout` (`cout`, `printf`) 皆會寫入到 `output.txt`。

範例輸入

Case #1:

0 10

4 +1 x2 /2 +5

Case #2:

0 2

3 +4 x9 >>

範例輸出

2 5 +4 x9 >> x9 >>

1 2 +5 x2

限制

- 每筆測資至少最多 24 次操作就能完成
- 操作最多 9 種
- 運算過程數字都會在 `int` 範圍
- 每個操作的 x 都只有 1 位數
- Case 1~70 都保證按鈕數量 + 操作數量 ≤ 16

- Case 71~80 都保證按鈕數量 + 操作數量 ≤ 19
- Case 81~100 都保證按鈕數量 + 操作數量 ≤ 28 且按鈕數量都是 4

分數計算

總共 100 個密碼鎖，每成功破解一個就得 1 分