

A. 好哈快捷鍵

Problem ID: Shortcut

「我要成為快捷鍵大師！」

由於你偶然發現世界上有好多人都不會使用 `Ctrl+Backspace` 這個快捷鍵，所以決定要來宣揚 `Ctrl+Backspace` 的偉大！

要傳教當然就要讓大家感受到這個快捷鍵的威力，於是你決定寫一個程式來展示 `Ctrl+Backspace` 的功能，準備讓大家嘆為觀止。

在這之前，讓我們定義一些規則。

1. 首先，一個英文句子除包含大小寫英文字母外，只會有 `,` (逗號), `.` (句號), `;` (分號)。
2. 標點符號的前一個字元沒有空白，而逗號和分號的後一個字元一定是空白。句號則為句子結尾，所以後面沒有空白。保證句子沒有多餘的空白，句尾一定會有句點。
3. 從句子的最後面開始使用 `Ctrl+Backspace`，每次會往前刪除，直到刪除了一整個單字 (word)，也就是至單字的開頭，範例如下

原句子長的樣子 “Hi, nice to meet you.”

按第一次快捷鍵 “Hi, nice to meet”

按第二次快捷鍵 “Hi, nice to”

按第三次快捷鍵 “Hi, nice”

按第四次快捷鍵 “Hi,”

按第五次快捷鍵 ""

注意行尾可能有空白

現在會給你一個英文句子，想問你經過 K 次的 `Ctrl+Backspace` 後的結果長什麼樣子。

輸入

輸入為單測資，共有兩列。

第一列有兩個整數 N, K ，代表英文句子的長度和等等要使用 `Ctrl+Backspace` K 次。

輸出

輸出只有一列，為所求的結果。

輸入限制

- $2 \leq N \leq 10^3$
- $0 \leq K \leq N$
- 令 Q 為該英文句子可使用 Ctrl+Backspace 的次數，保證 $K \leq Q$

子任務

編號	分數	限制
1	25	句子內只有一個單字 (word)
2	75	題目範圍

範例輸入

21 3

Hi, nice to meet you.

範例輸出

Hi, nice

提示

Ctrl: Control 鍵，在鍵盤的左下、右下角各有一顆。

B. 梨善富麗夜・變幻

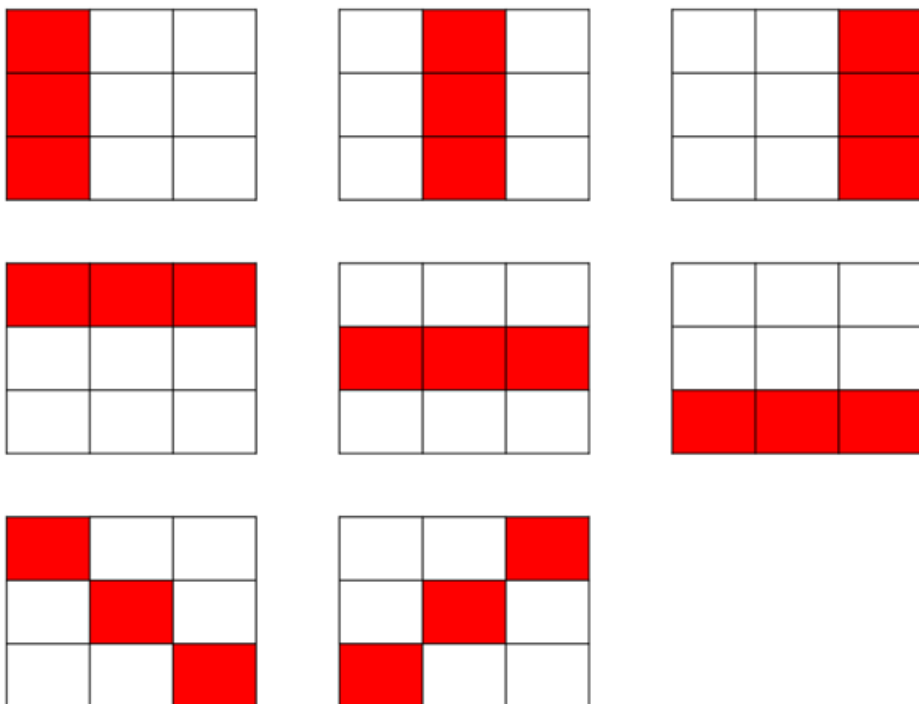
Problem ID: Yoshiriko



這是梨善。如果你沒看過，現在給你看。然而這和本題一點關係都沒有

給定 $n * m$ 的網格圖 (Grid)，問有多少種選 k 個相異格子，且剛好形成一條連線的選法。形成一條連線，表示可以用一條直線劃過所有格子，直線可以是縱、橫或斜的，且格子兩兩一定要都共用一條邊或都共用一個角 (點)。

以 3×3 的網格圖，選 3 個相異格子為例 ($n = 3m = 3k = 3$)，共有 8 種選法：



本題有 T 筆詢問。如果你不清楚連線的定義，可以參考備註的代數定義。

輸入

第一行有一個整數 T ，表示有 T 筆詢問。接下來有 T 行，每行有三個整數 n, m, k ，表示該筆詢問的網格圖高、寬及選擇的格子數量。

輸出

對於每筆詢問，輸出一個整數，表示選法數量，接著換行。

輸入限制

$1 \leq T \leq 50$ 所有詢問的 $2 \leq n, m, k \leq 10^5$

子任務

編號	分數	限制
1	9	所有詢問的 $n = m = k = 2$ 且 $T = 1$
2	11	所有詢問的 $n = 2$
3	20	所有詢問的 $2 \leq n, m \leq 5$
4	29	所有詢問的 $2 \leq n, m \leq 40$
5	31	無額外限制

範例輸入

5
3 3 3
3 3 2
4 4 3
4 2 6
6 9 5

範例輸出

8
20
24
0
68

備註

要求在 $n * m$ 的網格圖上選 k 個相異格子並形成連線，相當於詢問有多少種相異二元組序列 $(x_1, y_1), (x_2, y_2), (x_3, y_3) \dots (x_k, y_k)$ ，滿足以下八個條件：

- $1 \leq x_i \leq n$ ，對於所有的 $1 \leq i \leq k$
- $1 \leq y_i \leq m$ ，對於所有的 $1 \leq i \leq k$
- $x_i \leq x_{i+1}$ ，對於所有的 $1 \leq i < k$
- $x_{i+1} - x_i = x_{i+2} - x_{i+1}$ ，對於所有的 $1 \leq i < k - 1$
- $y_{i+1} - y_i = y_{i+2} - y_{i+1}$ ，對於所有的 $1 \leq i < k - 1$
- $x_2 - x_1 = 0$ 或 $x_2 - x_1 = k$
- $|y_2 - y_1| = 0$ 或 $|y_2 - y_1| = k$
- $\max\{x_2 - x_1, |y_2 - y_1|\} = k$

C. 蘿莉控的雲端硬碟

Problem ID: LoliDriver

ub33 是個蘿莉控，他非常喜歡蘿莉，所以他趁著去日本旅行的時候拍了很多蘿莉回家欣賞。而他的好朋友 ub44 也是個蘿莉控，俗話說「好東西要跟好朋友分享」，所以一起分享蘿莉的照片也是很重要的，像是 ub33 去日本拍蘿莉，而 ub44 去韓國拍蘿莉，兩個人再互相分享，這樣就能輕輕鬆鬆看到很多蘿莉，大飽眼福了。

不過這麼多照片如果用隨身碟傳輸實在太麻煩了，更何況 ub33 住在台南，而 ub44 住在台北，所以他們使用了 geegle 雲端硬碟來互相分享檔案。

有些蘿莉實在太珍貴了，ub33 想要獨享，他只想分享一些給 ub44。而且 ub33 很懶惰，他覺得要一張一張照片設訂共用實在太麻煩了，因此他只想要選擇一個資料夾設定共用，來讓他想分享的所有照片都被共用。

geegle 雲端硬碟的設計是，只要一個資料夾被共用，他底下的所有資料夾跟檔案全部都會跟著被共用，不管是底下幾層都一樣。

可是 ub33 也不想要設定了一個資料夾共用之後，有太多其他的檔案被看見，所以他希望設定的這個資料夾底下包含的檔案跟資料夾越少越好，可是還是要包含所有他要共用的檔案。

你能幫他找到他該設定哪個資料夾嗎？

如果你幫他找出來，除了給你分數之外，他搞不好還能給你一張他珍藏的蘿莉照片喔！

輸入

第一行兩個數字 n 、 q ，代表總共有 n 個檔案或資料夾，以及有 q 筆詢問。

第二行有 $n - 1$ 個數字，代表每個檔案或資料夾的上一層是哪個資料夾，從編號 1 的開始，因為編號 0 是根資料夾。

再來有 q 行，每行代表一筆詢問，每行剛開始有一個數字 x ，再來有 x 個數字 f_i ，每個 f_i 之間以空白隔開，代表有哪些編號的檔案或資料夾要共用。

編號 0 的資料夾是根資料夾。

輸出

輸出 q 行，代表對於每筆詢問要將哪個資料夾設成共用。

輸入限制

- $1 \leq n \leq 10^6$
- $1 \leq q \leq 100$
- $1 \leq x \leq 100$

子任務

編號	分數	限制
1	19	$x = 2$
2	19	$q = 1$
3	11	所有資料夾底下最多只有一個資料夾或檔案
4	7	對於每筆詢問，所有要共用的檔案的上一層資料夾都相同
5	17	$n \leq 100$
6	27	無額外限制

範例輸入

```

6 7
0 0 1 3 1
3 4 5 2
2 3 2
2 4 3
2 4 5
3 4 3 5
1 1
1 4

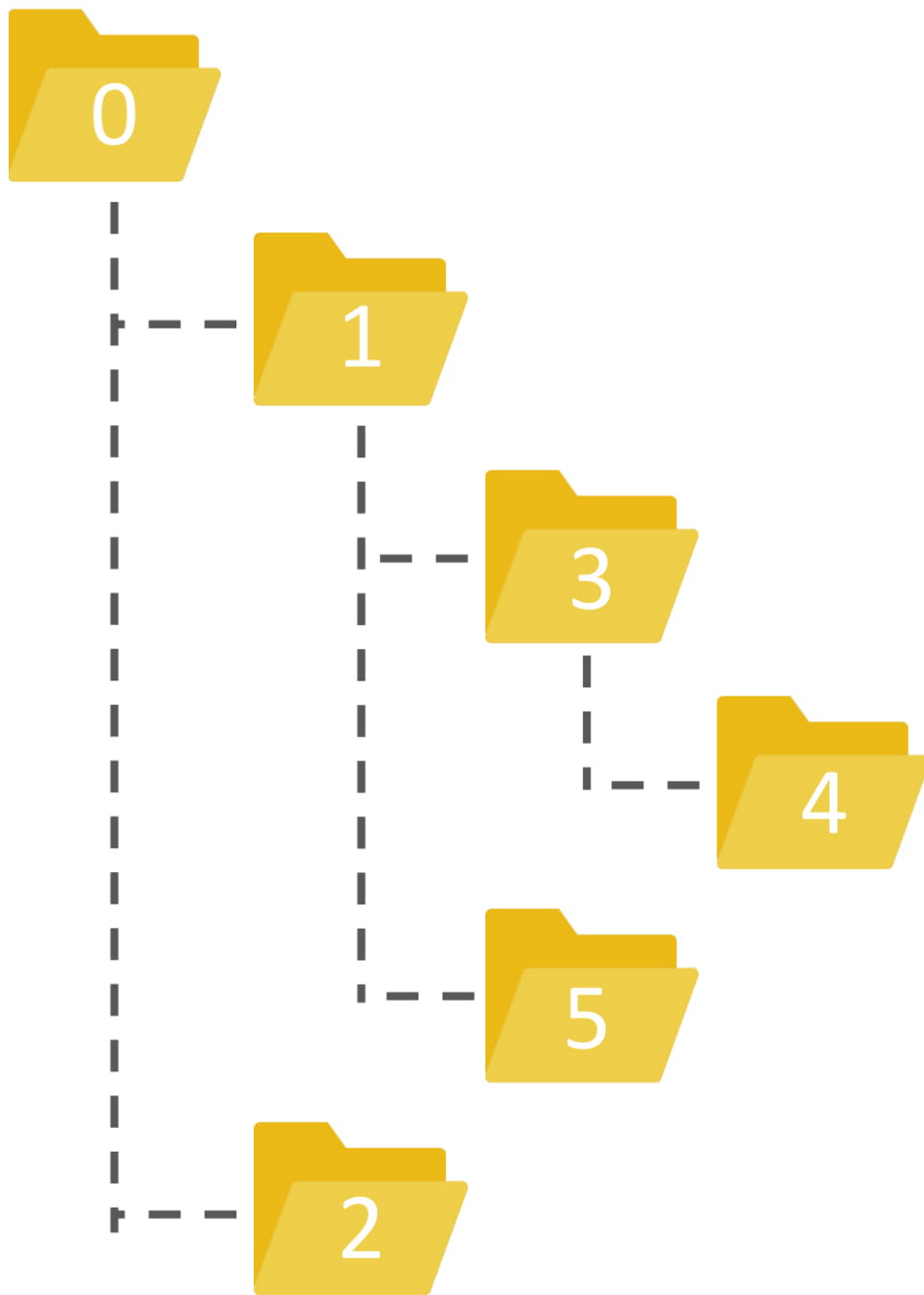
```

範例輸出

```

0
0
3
1
1
1
4

```



D. 蘿姬雜湊大法

Problem ID: Logichash

眾所皆知，如果所有帳號共用同一組密碼，一旦被盜就是全滅。如果所有帳號都使用不同密碼，很快就會忘光；寫在紙上貼螢幕上只會加速滅亡。

學過演算法的 llc 很快想到一招：何不利用 hash？沒錯，只要把帳號『用在哪個地方』拿來 hash 就能得到不同密碼！而且規則很難破解，但永遠只需要記一組規則。llc 決定將其命名為《lolihash 大法》！

例如用在 TOJ 的密碼，就將 TOJ 扔進 hash；用在 UVa 的密碼，就將 U.. 啊，UVa 已經不叫 UVa 了。..... 沒、沒關係，密碼是可以修改的！！

很快地，llc 已經完成了《lolihash 大法》的 code，可以將任意字串轉成 mod 1000000007 後的整數，沒錯，llc 只接受整數！但更快地，有些囉嗦嘮叨的網站要求每 16 天要換一次密碼，真是麻煩。加上有些重要網站綁有魔法卡，llc 希望有一套追加規則來處理密碼的再加密。

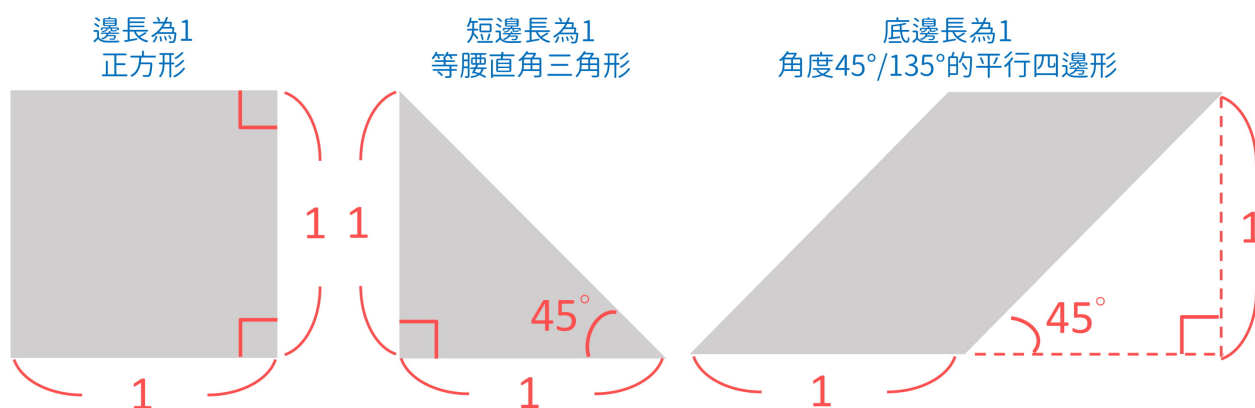
於是熱愛 math 的 llc 設定了一道 math 題，可以將 hash 後的數字作為輸入，把計算結果 mod 1000000007 作為輸出，這樣自己的輸出可以再作為自己的輸入再次加密！最後只要在密碼提示寫上再加密次數即可。llc 真他媽天才！！

llc 決定就將其命名為《logichash 大法》!! (lolihash + 積木 => 蘿莉積木雜湊 => 蘿姬雜湊 => logichash) 並且將題目藏到大海中，準備開啟大海賊時代。附帶一提，方法是漂流瓶。

不過為了分散風險，llc 決定不自己寫這道 math 的 code，將這份重責大任交給足以託付背後的你，並且不打算將 code 放自己身邊。再加密的自產自食部份 llc 已經完成，你只需要幫忙完成 math 的部份。

你收到的題目如下：

你擁有三種類型的積木：- 邊長 1 正方形 - 短邊長 1 等腰直角三角形 - 底邊長 1 角度為 45/135 度的平行四邊形



求有多少種不同方式，拼成高 1 底 n 的長方形？

噢，別忘了幫 llc 保密。

輸入

輸入僅有一正整數 n

輸出

輸出方法數 mod 1000000007

範圍

$0 < n \leq 1000000007$

子任務

編號	分數	限制
1	10	$n \leq 4$
2	20	$n \leq 10$
3	30	$n \leq 10^6$
4	40	無額外限制

範例輸入 1

1

範例輸出 1

3

範例輸入 2

2

範例輸出 2

11

E. 地板置物櫃

Problem ID: FloorBox

社部的地板是個神奇的東西，一整片地板被分成了一格一格的，而且居然還可以打開，超酷的對吧！

由於地板具有被分成一格一格而且還可以打開的特性，於是不知道哪位腦洞很大的社員就把它拿來當置物櫃用，久而久之東西越來越多，一格地板已經放不下了，需要更多格的地板來放。

到後來整個社部的地板底下就都被拿來放東西了。

社部的管理員 Joseph 看到這個現象後，發現了一個賺錢的機會--收租金。

不過因為地板置物櫃也有等級之分，所以租金也就有高低之分，像是在桌子底下的等級就比較低，因為比較難打開，所以租金就相對便宜。

地板置物櫃還有一個特性是每格地板能放的物品數量是一樣多的。

聽到要收租金，所有社員們就決定來想個辦法盡量繳少一點的租金。

大家想到了一個很簡單但也很直接的方法，把物品盡量集中起來，這樣一來就不用用到這麼多置物櫃，租金自然也就比較低。

但是要移動這些物品得請住在地板底下的小精靈幫忙，這些小精靈不僅會傳遞訊息 (toj133)，也幫忙在地板置物櫃之間移動物品，整個地板置物櫃就像倉儲系統一樣。

而小精靈幫忙搬東西當然也就需要收費，不過他們收費方式很簡單，不管什麼物品要從哪裡移動到哪裡，收費都一樣，每個物品的移動固定收費 x 元。

社部的地板被分割成 n 格，每格的租金分別是 a_i ，而且每格地板都能放 k 樣物品。

給你目前每格置物櫃的物品數量，請你幫忙計算出社員們的最少花費，也就是租金 + 小精靈的搬運費最少是多少。

輸入

第一行有 3 個整數， n, k, x ，有 n 個置物櫃，每個置物櫃都能放 k 個物品，每個物品移動都要付給小精靈 x 元

第二行有 n 個整數 a_i ，代表第 i 個置物櫃租金是 a_i ($1 \leq i \leq n$)。

第三行有 n 個整數 b_i ，第 i 個置物櫃放了 b_i 個物品 ($1 \leq i \leq n$)。

保證一開始每個置物櫃都有物品

輸出

請輸出一個整數代表最少的總租金 + 小精靈的搬運費

範圍

$$0 < n \leq 10^6$$

$$0 \leq a_i, x \leq 1000$$

$$1 \leq k \leq 10$$

$$0 < b_i \leq k$$

子任務

編號	分數	限制
1	3	$a_i = 0$
2	7	$x = 0$
3	5	$k = 1$
4	21	$a_i = a_{i+1} \ (1 \leq i < n)$
5	25	$n \leq 18$
6	39	無額外限制

範例輸入 1

3 2 4

1 2 3

1 2 1

範例輸出 1

6

範例輸入 2

3 2 1

1 2 3

1 2 1

範例輸出 2

4

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 " , " x3 " , " /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 " , " /2 " , " x2 " , " +5 "，那麼密碼可以是 " +5+5 " 也可以是 " +5x2 " 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 也想的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。

而你也很有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

" xX " 把原本的數字乘 x

" /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，操作完變成 1235，原本是 -234，x 是 1，操作完變成 -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