

A. 進位或不進位

問題描述:

$6+8=14$ 看起來似乎是沒問題的。可是為什麼 $4+6=2$ 呢？

如你所見，小明在數位邏輯課中始終非常努力，然而在某次考試中小明執行一個範圍為 32bit 的加法時，他在設計的部份犯了一些錯誤。經過了半小時後，他終於發現問題所在了！當他在執行二進位加法的過程中，原本進位時該補 1 到下一位卻沒有補，以致於進位動作沒有完全。就像下例一樣：

```
4 = 00000000 00000000 00000000 00000100
+6 = 00000000 00000000 00000000 00000110
-----
2 = 00000000 00000000 00000000 00000010
```

對小明來說，能夠發現問題所在是件好事，可惜似乎有點太晚了。老師考慮到小明上課的用心，決定再給小明一次機會。這次小明必須設計出一個有效率的程式，這個程式能夠將 2 個無正負號 10 進位的整數相加，並輸出兩者的和（以 10 進位表示），而加法的運算方式必須像小明剛剛的做法一樣，也就是進位時不補 1 到下一位。

輸入說明:

每筆測試資料一行，有 2 個 10 進位的無正負號整數。整數的範圍 $\leq 2^{32}$ 。
為多筆測資。

輸出說明:

對於每筆測試資料，輸出 1 個 10 進位的無正負號整數，這個整數為使用小明上課時使用的方法作兩數相加的和。

Sample input
2
11
47
1234567892
0
Sample output
2
2
2
2

B. 修正字串

內容:

打字時一個常見的錯誤就是沒有把手放在正確位置，而是偏右邊一個位置。所以會發生 Q 被打成 W，J 被打成 K 等等的情況。你的任務就是要把打錯的字修正回來。



輸入說明:

輸入包含許多列，每列可能包含有數字，空白字元，大寫英文字母（Q、A、Z 除外），標點符號（`除外）。

輸出說明:

對每一列中的每個字元，請輸出在鍵盤(如上圖)上其左邊一個位置的字元。但是輸入中的空白字元，輸出時亦請輸出空白字元。

Sample input
O S, GOMR YPFSU/ URD. ,U [JPMR MI,NRT OD 8346333
Sample output
I AM FINE TODAY. YES, MY PHONE NUMBER IS 7235222