

Problem 9: Hex Carries

Adding hexadecimal numbers is not too difficult, but you must remember the carries! Like adding decimal numbers, carries of 1 are generated when the sum of two digits and the carry from the right exceeds the maximum value that can be represented in a single digit (15, of course, for hexadecimal digits). In this problem you will be given a pair of hexadecimal numbers and asked to determine the number of non-zero carries that are involved in adding these numbers. We don't care about the sum.

For example, suppose the numbers to be added are 1A2B and D6E4. The following steps would be used to determine the number of carries.

- B (decimal 11) and 4 are added, yielding a sum of 15 (there is no carry into the low-order position). Since this is less than 16, no carry is generated.
- 2 and E (decimal 14) are added (with no carry from the preceding step), yielding a sum of 16. Since this is too large for a single hexadecimal digit (it's larger than 15), a carry is generated to the next step.
- A (decimal 10) and 6 are added, along with the carry from the previous step. The sum is $10 + 6 + 1 = 17$, and a carry is generated.
- Finally, 1 and D (decimal 13) and the carry from the previous step are added yielding $1 + 13 + 1 = 15$, and no carry is generated.

So a total of 2 carries were generated, and this is the answer for this case.

Input

There will be multiple cases to consider, each numbered sequentially starting with 1. For each case the input consists of a single line containing two non-zero hexadecimal numbers, each containing no more than 30 hexadecimal digits. The allowable hexadecimal digits are 0 through 9 and 'A' through 'F' and 'a' through 'f'. These numbers are separated, and possibly preceded and followed by spaces. The input for the last case is followed by a line containing two hexadecimal numbers, each consisting of a single '0'.

Output

For each case display the case number and the number of carries generated. Use a format identical to that shown in the samples below, and display a blank line after the output for each case.

Sample Input

```
1A2B D6E4
1 f
  1 222222222
  0 0
```

Output for the Sample Input

```
Case 1: 2 carries

Case 2: 1 carry

Case 3: No carries
```