

# Problem A: Grade School Multiplication

Source file: multiply.{c, cpp, java}

Input file: multiply.in

An educational software company, All Computer Math (ACM), has a section on multiplication of integers. They want to display the calculations in the traditional grade school format, like the following computation of  $432 \times 5678$ :

```

  432
 5678
-----
 3456
 3024
 2592
 2160
-----
2452896

```

Note well that the final product is printed without any leading spaces, but that leading spaces are necessary on some of the other lines to maintain proper alignment. However, as per our regional rules, there should *never* be any lines with *trailing* white space. Note that the lines of dashes have length matching the final product.

As a special case, when one of the digits of the second operand is a zero, it generates a single 0 in the partial answers, and the next partial result should be on the *same* line rather than the next line down. For example, consider the following product of  $200001 \times 90040$ :

```

 200001
 90040
-----
 8000040
180000900
-----
18008090040

```

The rightmost digit of the second operand is a 0, causing a 0 to be placed in the rightmost column of the first partial product. However, rather than continue to a new line, the partial product of  $4 \times 200001$  is placed on the same line as that 0. The third and fourth least-significant digits of the second operand are zeros, each resulting in a 0 in the second partial product on the same line as the result of  $9 \times 200001$ .

As a final special case, if there is only one line in the partial answer, it constitutes a full answer, and so there is no need for computing a sum. For example, a computation of  $246 \times 70$  would be formatted as

```

 246
 70
-----
17220

```

Your job is to generate the solution displays.

**Input:** The input contains one or more data sets. Each data set consists of two positive integers on a line, designating the operands in the desired order. Neither number will have more than 6 digits, and neither will have leading zeros. After the last data set is a line containing only 0 0.

**Output:** For each data set, output a label line containing "Problem " with the number of the problem, followed by the complete multiplication problem in accordance with the format rules described above.

Warning: A standard int type cannot properly handle 12-digit numbers. You should use a 64-bit type (i.e., a long in Java, or a long long in C++).

Example Input:	Example Output:
<pre>432 5678 200001 90040 246 70 0 0</pre>	<pre>Problem 1   432   5678 -----  3456  3024  2592 2160 ----- 2452896 Problem 2  200001   90040 -----  8000040 180000900 ----- 18008090040 Problem 3   246    70 ----- 17220</pre>