



Prime Cryptarithm

(This poorly named task has nothing to do with prime numbers or even, really, prime digits. Sorry 'bout that.)

A cryptarithm is usually presented as a pencil-and-paper task in which the solver is required to substitute a digit for each of the asterisks (or, often, letters) in the manual evaluation of an arithmetic term or expression so that the consistent application of the digits results in a proper expression. A classic example is this cryptarithm, shown with its unique solution:

| | | | | | |
|--------|--------|------|------|------|------|
| SEND | 9567 | S->9 | E->5 | N->6 | D->7 |
| + MORE | + 1085 | M->1 | O->0 | R->8 | |
| ----- | ----- | | | | |
| MONEY | 10652 | Y->2 | | | |

The following cryptarithm is a multiplication problem that can be solved by substituting digits from a specified set of N digits into the positions marked with *. Since the asterisks are generic, any digit from the input set can be used for any of the asterisks; any digit may be duplicated as many times as desired.

Consider using the set {2,3,5,7} for the cryptarithm below:

| | |
|---------|--|
| * * * | |
| x * * | |
| ----- | |
| * * * | <-- partial product 1 -- MUST BE 3 DIGITS LONG |
| * * * | <-- partial product 2 -- MUST BE 3 DIGITS LONG |
| ----- | |
| * * * * | |

Digits can appear only in places marked by '*'. Of course, leading zeroes are not allowed.

The partial products must be three digits long, even though the general case (see below) might have four digit partial products.

***** Note About Cryptarithm's Multiplication *****

In USA, children are taught to perform multidigit multiplication as described here. Consider multiplying a three digit number whose digits are 'a', 'b', and 'c' by a two digit number whose digits are 'd' and 'e':

[Note that this diagram shows far more digits in its results than the required diagram above which has three digit partial products!]

| | | |
|----|-----------|--|
| | a b c | <-- number 'abc' |
| | x d e | <-- number 'de'; the 'x' means 'multiply' |
| | ----- | |
| p1 | * * * * | <-- product of e * abc; first star might be 0 (absent) |
| p2 | * * * * | <-- product of d * abc; first star might be 0 (absent) |
| | ----- | |
| | * * * * * | <-- sum of p1 and p2 (e*abc + 10*d*abc) == de*abc |

Note that the 'partial products' are as taught in USA schools. The first partial product is the product of the final digit of the second number and the top number. The second partial product is the product of the first digit of the

second number and the top number.

Write a program that will find all solutions to the cryptarithm above for any subset of supplied non-zero single-digits. Note that the multiplicands, partial products, and answers must all conform to the cryptarithm's framework.

PROGRAM NAME: crypt1

INPUT FORMAT

| | |
|---------|---|
| Line 1: | N, the number of digits that will be used |
| Line 2: | N space separated non-zero digits with which to solve the cryptarithm |

SAMPLE INPUT (file crypt1.in)

```
5
2 3 4 6 8
```

OUTPUT FORMAT

A single line with the total number of solutions. Here is the one and only solution for the sample input:

```
  2 2 2
x   2 2
-----
  4 4 4
4 4 4
-----
4 8 8 4
```

SAMPLE OUTPUT (file crypt1.out)

```
1
```

OUTPUT DETAILS

Here's why 222x22 works: 3 digits times 2 digits yields two (equal!) partial products, each of three digits (as required). The answer has four digits, as required. Each digit used {2, 4, 8} is in the supplied set {2, 3, 4, 6, 8}.

Why 222x23 doesn't work:

```
  2 2 2  <-- OK:  three digits, all members of {2, 3, 4, 6, 8}
  2 3   <-- OK:  two digits, all members of {2, 3, 4, 6, 8}
-----
  6 6 6  <-- OK:  three digits, all members of {2, 3, 4, 6, 8}
  4 4 4  <-- OK:  three digits, all members of {2, 3, 4, 6, 8}
-----
  5 1 0 6 <-- NOT OK: four digits (good), but 5, 1, and 0 are not in
                        {2, 3, 4, 6, 8}
```

Submit a solution:

No file chosen

[Training Gateway](#) | [Comment or Question](#)