You may solve the following 3 exercises in the coding language of your choice. Computationally efficient solutions are desired, assume inputs might be arbitrarily large.

EX1: ASCII codes

Write a function that, given a string of digits, translates it into a human sentence, assuming the digits are a chain of ascii codes. Translate the following 6 sentences:

1221019811497

1145010050

701051141151164432115111108118101321161041013211211411198108101109463284104101110443211911410511610132116104101329911110010146

821179812132105115321141179898105115104333280728032105115321121041129711011697115116105933

661011021111141013211511110211611997114101329997110329810132114101117115979810810132105116321021051141151163210497115321161113298101321171159798108101321051163210210511411511632104971153211611132981013211711597981081014610

671111001013210511532108105107101321041171091111144632871041011103212111111732104971181013211611132101120112108971051103210511644321051163911532989710 0

(14) that cannot be added up with other two numbers without exceeding 15. All the other numbers can be added up. For instance, 2 can be added up with 5 and 1, giving as a result 8, that is less or equal than 15.

Input

The input consists of a line with an integer v, followed by a line with a number n between 3 and 10000, followed by a line with n different numbers.

Output

Your program must print a line with the quantity of the given numbers that can be added up with other two of these numbers so that the result is not greater than v.

Input 1	**Output 1**
15 5 12 2 5 14 1	**4**
Input 2	**Output 2**
6 3 1 2 3	**3**
Input 3	**Output 3**
-20 6 25 15 5 -5 -15 -25	**5**
Input 4	**Output 4**
100 3 33 34 35	**0**