L. Table Format

Program:	table.(cpp|java|py)
Input:	table.in
Balloon Color:	Silver

Description

We use tables with a number of rows and columns that are filled with text in several documents. Given a table with a fixed number of columns and number of rows, a fixed width, and a single sentence in every cell, your task is to find the minimum number of lines required so that the text fits the table.

For this problem, the number of rows is fixed to 2 and number of columns to 3.

Assume that all characters need the same space, only alphabetical characters are used, no punctuation marks, a single space is used to separate every two words, and no space needed at the beginning or the end of the statement. A word cannot be broken into two lines, if there is no room for the last word in one line, it must be moved into next line. On the other hand, if the first word in a line can fit into the end of previous line, then it must be moved to the previous line.

Example

Given the following six strings:

```
Today is Friday
Good afternoon
Java
GPC
I win this
Good morning Oman
```

If we would like to fit them in a 3x2 table with a overall width of 46 characters, we would need two lines as shown below in Table 1:

Table 1. Text arranged with width = 46, lines = 2

Today is Friday	Good afternoon	Java
GPC	I win this	Good morning Oman

In Table 1 the first column has a width of 15 (Today = 5, is = 2, Friday = 6, and 2 spaces), the second is 14 (Good = 4, afternoon = 9, and 1 space) characters wide, and the third is 17 (Good = 4, morning =7, Oman = 4, and 2 spaces). Therefore, the table is 46 characters wide. If we are given a width of 46, then the minimum number of lines needed is 2, with one line per row.

In Table 2, the width is limited to 45, therefore, one row must be stretched, allowing a sentence to span over two lines:

Table 2. Text arranged with width = 45, lines = 3

Today is Friday	Good afternoon	Java
GPC	I win this	Good morning Oman

In Table 3, with a width of 34, it is impossible to fit the text in 3 lines, therefore minimum number of lines is 4:

Table 3. Text arranged with width = 34, lines = 4

Today is Friday	Good afternoon	Java
GPC	I win this	Good morning Oman

Note that for the same table a width of 29 is sufficient to fit the text in 4 lines (8 + 9 + 12). However, if we would like to bring the word "Good morning Oman" into a single line, we must increase the width of the right column to 17. Similarly, bringing "I win this" into a single line, requires the middle column to have 10 characters, for a total width of 35, as shown in Table 4:

Table 4. Text arranged with width = 35, lines = 3

Today is Friday	Good afternoon	Java
GPC	I win this	Good morning Oman

Input

The input consists of several cases. Every case starts with a positive integer, *k* that represents the table width. Six lines follow, each line contains the statement to fit into one cell. Statements should fill the table in a row-major order (i.e. first row is filled, then the second row, going from left to right). Each statement contains at least one word, and at most 10 words. Input is terminated by having $k = 0$, which should not be processed. $min \leq k \leq 100$, where *min = longest word in column 1 + longest word in column 2 + longest word in column 3. Maximum length of any word = 20.*

Output

For each sequence, you are to output one line, containing the minimum number of lines.

Sample Input / Output

table.in

```
46
Today is Friday
Good afternoon
Java
GPC
I win this
Good morning Oman
45
Today is Friday
Good afternoon
Java
GPC
I win this
Good morning Oman
34
Today is Friday
Good afternoon
Java
GPC
I win this
Good morning Oman
15
Today is Friday
Good afternoon
Java
GPC
I win this
Good morning Oman
50
Today is Friday
The quick brown fox jumps over the lazy dog
International day
The quick brown fox jumps over the lazy dog
Today is a nice day
Good morning Oman
0
```

OUTPUT

```
2
3
4
6
6
```