ADPF Assignment – 2

1. Write a C function to find the k^{th} occurrence of an even integer in a sequence of non-negative integers, and then call your function from main. Your function should be according to the following declaration:

   int find_even(int k);

   Input

   You are given the input in two lines:

   -The first line contains a positive integer k.
   -In the second line, you will be given a sequence of numbers.
   You have to find the k^{th} occurrence of n in the sequence below. The second line consists of a sequence of non-negative integers, terminated with a -1. The -1 is not part of the sequence.

   Output

   If there are k even numbers in the sequence, then output the k^{th} occurrence of even in the sequence. Otherwise, output a -1.

   Sample Input

   2
   1 1 3 2 3 4 1 -1
   Sample Output

   4

Sample Test Cases	Input	Output
Test Case 1	2 1 1 2 3 3 -1	-1
Test Case 2	2 2 1 1 2 3 3 -1	2
Test Case 3	1 1 1 2 3 3 -1	2
Test Case 4	10 1 1 2 3 3 -1	-1
Test Case 5	3 1 2 3 4 5 6 7 -1	6
Test Case 6	3 1 -1	-1

2. When you keep track of stock prices, or your weight during a diet programme, the daily prices or weights fluctuate a lot. One way to identify the general trend is to keep track of the average over the last 3 days, for example. This technique often smooths out the fluctuations, and makes the trend clearer. This technique is called "moving average".

In this question, you have to output the "moving average" of a sequence of non-negative numbers. The moving average is the sequence of averages of the last 3 entries. For the first 2 numbers, no average is output.

For example, if the sequence of numbers is

a1, a2, a3, a4, a5

and k=3, then the 3-moving average is

(a3+a2+a1)/3, (a4+a3+a2)/3, (a5+a4+a3)/3.

Input
The input is a sequence of non-negative floating point numbers, terminated by a -1. The -1 is not part of the sequence. There will be at least 3 numbers in the sequence.

Output
You have to output the moving average of the sequence. The output should be printed correct to one digit after the decimal.

Sample Input 1
70.8 70.9 71.2 70.7 70.2 -1

Sample Output 1
71.0 70.9 70.7

Sample Test Cases

	Input	Output
Test Case 1	1 2 0 1 -1	1.0 1.0
Test Case 2	1 1 2 3 5 8 13 21 -1	1.3 2.0 3.3 5.3 8.7 14.0

3. A line of English text will be given, where words are separated by one of the following symbols:
' ', '\t', '.', ',' and ';'

Each word may be separated from the next and the previous by one or more of the following symbols. You have to count the number of words in the sentence.
Note that to read the input, you have to read until the EOF symbol is read, as in the following example.

```
int main(){
        int c;
        c = getchar();
        while ( c != EOF ) {
                c = getchar();
        }
        return 0;
}
```

Note: it is possible to solve this question without arrays, so the maximum length of the line is not important.

Input

A line of English text with words separated from one another by one or more occurrences of the symbols
' ' , '\t' , '.' , ',' and ';'

Output

The number of words in the line.

Sample Input

This is a sentence, it has words separated by spaces and fullstops.

Sample Output

12

Sample Test Cases

	Input	Output
Test Case 1	this is a line.	4
Test Case 2	This is a third line, with a lot of punctuation... but it should be alright.	15
Test Case 3	in fac;t the punctuation need not be proper.	9
Test Case 4	one more sentence,	3
Test Case 5	science is awesome.	3
Test Case 6	This is another line.	4

4. Given two arrays of integers output the smallest number in the first array not present in the second one.

Input Specification:
The first line contains the size N1 of the first array.
Next line give the contents of the first array.
Next line contains the size N2 of the second array.
Next line give the contents of the second array.

Output Format:
Output must be a single number which is the smallest number occurring in the first array that does not occur in the second. In case there is no such number, output NO.

Variable Constraints:
The sizes of the arrays are smaller than 20. Each array entry is an integer which fits an int data type.

Example:
Input:
3
2 3 4
4
1 3 5 7

Output: 2

Input
1
1
2
1 2
Output: NO

Sample Test Cases	Input	Output
Test Case 1	5 1 2 3 4 5 3 2 7 9	1
Test Case 2	4 3 5 7 9 4	3

	7 9 10 55	
Test Case 3	2 4 6 2 4 7	6
Test Case 4	3 2 7 1 3 1 7 2	NO
Test Case 5	3 1 3 4 4 1 3 5 7	4
Test Case 6	1 1 2 1 2	NO

5. Find the number of distinct numbers in a given sequence. The sequence need not be sorted.

Input

The input consists of two lines.

The first line consists of a positive number N. N is at most 1000.

The second line consists of N numbers separated by spaces.

Output

The output should be the number of distinct elements in the sequence.

Sample Input

4

1 2 3 1

Sample Output

3

Sample Test Cases

	Input	Output
Test Case 1	4 33 66 77 66	3
Test Case 2	7 0 0 0 0 0 0 0	1
Test Case 3	5 1 2 3 4 -5	5
Test Case 4	3 1 2 1	2
Test Case 5	4 4 6 8 0	4

6. Write Program to generate following pattern for input size N.
For N = 5 output is: