**Justifier**

By TopCoder\* https://urionlinejudge.r.worldssl.net/gallery/images/flags/us.gif USA

**Timelimit: 1**

We have some words and we want to right justify them, that is, align them to the right. Create a program that reads a word and print it all right justified, in the same order as they appear in the input.

**Input**

The input contains several test cases. The first line of a test case will contain an integer **N** (1 ≤ **N** ≤ 50) indicating the number of following words. Each word is composed of up to 50 uppercase letters (‘A’-‘Z’) and will contain at least one letter. The end of input is indicated by **N** = 0.

**Output**

For each test case print the words padded on the left with space characters so that they are all the same length as the longest word found in that text. Print an empty line between all the test cases. There must be no trailing spaces printed out, and you should discard any unnecessary leading spaces, so that at least one line on every output word starts with a letter.

| **Sample Input** | **Sample Output** |
| --- | --- |
| 3 BOB TOMMY JIM 4 JOHN JAKE ALAN BLUE 4 LONGEST A LONGER SHORT 0 | BOB TOMMY   JIM  JOHN JAKE ALAN BLUE  LONGEST       A  LONGER   SHORT |

<https://www.urionlinejudge.com.br/judge/es/problems/view/1273>

static void Main(string[] args)

{

int N = int.Parse(Console.ReadLine());

while (N != 0)

{

List<string> words = new List<string>();

int max\_len = 0;

for (int i = 0; i < N; i++)

{

words.Add(Console.ReadLine());

max\_len = Math.Max(max\_len, words[i].Length);

}

for (int i = 0; i < words.Count; i++)

{

for (int j = 0; j < max\_len - words[i].Length; j++)

{

Console.Write(" ");

}

Console.WriteLine(words[i]);

}

N = int.Parse(Console.ReadLine());

if (N != 0)

{

Console.WriteLine();

}

}

//Console.ReadLine();

}