**6 kyu**

**TV Remote (shift and space)**

14995% of 6142 of153[dinglemouse](https://www.codewars.com/users/dinglemouse)

Java

* [TRAIN](https://www.codewars.com/kata/tv-remote-shift-and-space/train/java)
* [NEXT KATA](https://www.codewars.com/trainer/java)

Details

[Solutions](https://www.codewars.com/kata/tv-remote-shift-and-space/solutions/java)

[Forks (1)](https://www.codewars.com/kata/tv-remote-shift-and-space/forks/java)

[Discourse (24)](https://www.codewars.com/kata/tv-remote-shift-and-space/discuss/java)

* |
* Share this kata:

**Background**

My TV remote control has arrow buttons and an OK button.

I can use these to move a "cursor" on a logical screen keyboard to type words...

**Keyboard**

The screen "keyboard" layout looks like this

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| a | b | c | d | e | 1 | 2 | 3 |
| f | g | h | i | j | 4 | 5 | 6 |
| k | l | m | n | o | 7 | 8 | 9 |
| p | q | r | s | t | . | @ | 0 |
| u | v | w | x | y | z | \_ | / |
| aA | SP |  |  |  |  |  |  |

* aA is the SHIFT key. Pressing this key toggles alpha characters between UPPERCASE and lowercase
* SP is the space character
* The other blank keys in the bottom row have no function

**Kata task**

How many button presses on my remote are required to type the given words?

Notes

* The cursor always starts on the letter a (top left)
* The alpha characters are initially lowercase (as shown above)
* Remember to also press OK to "accept" each letter
* Take a direct route from one letter to the next
* The cursor does not wrap (e.g. you cannot leave one edge and reappear on the opposite edge)
* Although the blank keys have no function, you may navigate through them if you want to
* Spaces may occur anywhere in the words string.
* Do not press the SHIFT key until you need to. For example, with the word e.Z, the SHIFT change happens **after** the . is pressed (not before)

**Example**

words = Code Wars

* C => a-f-k-p-u-aA-OK-U-P-K-F-A-B-C-OK = 14
* o => C-H-M-R-W-V-U-aA-OK-SP-v-q-l-m-n-o-OK = 16
* d => o-j-e-d-OK = 4
* e => d-e-OK = 2
* space => e-d-c-b-g-l-q-v-SP-OK = 9
* W => SP-aA-OK-SP-V-W-OK = 6
* a => W-V-U-aA-OK-u-p-k-f-a-OK = 10
* r => a-f-k-p-q-r-OK = 6
* s => r-s-OK = 2

Answer = 14 + 16 + 4 + 2 + 9 + 6 + 10 + 6 + 2 = 69

*Good Luck!  
DM.*

Series

* [TV Remote](https://www.codewars.com/kata/tv-remote)
* TV Remote (shift and space)
* [TV Remote (wrap)](https://www.codewars.com/kata/tv-remote-wrap)
* [TV Remote (symbols)](https://www.codewars.com/kata/tv-remote-symbols)

<https://www.codewars.com/kata/tv-remote-shift-and-space/java>

*/\**

*\* To change this license header, choose License Headers in Project Properties.*

*\* To change this template file, choose Tools | Templates*

*\* and open the template in the editor.*

*\*/*

**package** javaapplication41;

**import** java.util.HashMap;

***/\*\****

***\****

***\* @author Usuario***

***\*/***

**public** **class** JavaApplication41 {

**static** **int** ObtenerDistancia(HashMap<String, **int**[]> diccio, String a, String b )

        {

**return** Math.abs(diccio.get(a)[0] - diccio.get(b)[0]) + Math.abs(diccio.get(a)[1] -diccio.get(b)[1]) + 1;

        }

*//static Dictionary<string, Dictionary<string, double>> pTable = Preloaded.CreateTable();*

**public** **static** **int** tvRemote(String word)

        {

            String[][] control = {

             {"a","b","c","d","e","1","2","3" },

             {"f","g","h","i","j","4","5","6" },

             {"k","l" ,"m","n","o","7","8","9" },

             {"p","q","r","s","t",".","@","0"},

             {"u","v","w","x","y","z","\_","/" },

             {"aA","SP"}

            };

            HashMap<String, **int**[]> diccio =

**new** HashMap();

**for** (**int** i = 0; i < control.length; i++) {

**for** (**int** j = 0; j < control[i].length; j++){

*//diccio[control[i][j]] = new int[] { i + 1, j + 1 };*

**int**[] arr = {i+1, j+1};

                     diccio.put( control[i][j] , arr );

                  }

            }

**int** sum = 0;

            String anterior = "a";

**boolean** mayus = **false**;

**for** (**int** i = 0; i < word.length(); i++)

            {

*//string c = char.ToLower(word[i]).ToString();*

                String c = String.valueOf(Character.toLowerCase(word.charAt(i)));

**if**(word.charAt(i) != ' ' && ((Character.isUpperCase(word.charAt(i)) && !mayus) ||(Character.isLowerCase(word.charAt(i)) && mayus)))

                {

                    sum += ObtenerDistancia(diccio, "aA", anterior);

                    sum += ObtenerDistancia(diccio, c, "aA");

                    mayus = !mayus;

                }

**else** **if** (word.charAt(i) == ' ') sum += ObtenerDistancia(diccio, "SP", anterior);

**else** sum += ObtenerDistancia(diccio, c, anterior);

                anterior = c;

**if** (word.charAt(i) == ' ')

                {

                    anterior = "SP";

                }

            }

*// Your code here*

**return** sum;

        }

**public** **static** **void** main(String[] args) {

*// TODO code application logic here*

        String s = "Does";

        System.out.println(tvRemote(s));

    }

}

[k14](https://www.codewars.com/users/k14)

**public class Dinglemouse {**

**public static int tvRemote(final String words) {**

**char old\_char = 'a';**

**int keys=0;**

**boolean shift = false;**

**for(char c:words.toCharArray()) {**

**if(!shift&&Character.isUpperCase(c)||shift&&Character.isLowerCase(c)){**

**keys+=keypadDistance('A', old\_char)+1;**

**old\_char='A';**

**shift=!shift;**

**}**

**c=Character.toLowerCase(c);**

**keys+=keypadDistance(Character.toLowerCase(c),old\_char)+1;**

**old\_char = c;**

**}**

**return keys;**

**}**

**private static int keypadDistance(char a, char b){**

**String s = "abcde123fghij456klmno789pqrst.@0uvwxyz\_/A ";**

**return Math.abs(s.indexOf(a) / 8 - s.indexOf(b) / 8)**

**+Math.abs(s.indexOf(a) % 8 - s.indexOf(b) % 8);**

**}**

**}**