Author

[Stevey](https://codefights.com/profile/Stevey)

https://codefights.com/img/coins_new.png2000

A Group of people is about to enter a building. Unfortunately they just can't decide in which order to do it, so they called you and asked for help.

You have a very special way of sorting a list of names. For each name you calculate the sum of its characters (where'a' = 0, 'b' = 1, ..., 'z' = 25, case insensitive), and sort the names by that sum in ascending order. Luckily for you there are no names with equal sum in the group, so the sorting is relatively easy.

Implement a function that performs this kind of sorting, so that the next time the Group of people is indecisive, they won't call you again.

**Example**

For group = ["Denise", "Tim", "Jennifer"], the output should be  
SortingFun(group) = ["Tim", "Denise", "Jennifer"],  
because:

* Tim has a sum of 19 + 8 + 12 = 39;
* Denise has a sum of 3 + 4 + 13 + 8 + 18 + 4 = 50;
* Jennifer has a sum of 9 + 4 + 13 + 13 + 8 + 5 + 4 + 17 = 73.
* **[input] array.string group**

The names of the people in the group, 1 ≤ group.length ≤ 20.  
It is guaranteed that no names have the same sum of characters.

* **[output] array.string**

The order in which people in the group should enter the building.

<https://codefights.com/challenge/r3RqtqBxQHBDFRWsw>

static string[] SortingFun(string[] group)

{

List<int> sumas = new List<int>();

for (int i = 0; i < group.Length; i++)

{

int sum = 0;

for (int j = 0; j < group[i].Length; j++)

{

sum += (int)(group[i][j]) - 97;

}

sumas.Add(sum);

}

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

{

for (int j = i + 1; j < sumas.Count; j++)

{

if (sumas[i] > sumas[j])

{

int temp = sumas[i];

sumas[i] = sumas[j];

sumas[j] = temp;

string tempNombre = group[i];

group[i] = group[j];

group[j] = tempNombre;

}

}

}

return group;

}