You are given a string, your goal is to rearrange its letters in alphabetical order. If the same letter is present both in lowercase and uppercase forms in the initial string, the uppercase occurrence should go first in the resulting string.

**Example**

For str = "Vaibhav", the output should be  
sortIt(str) = "aabhiVv".

* **[time limit] 3000ms (cs)**
* **[input] string str**

A string consisting only of letters.

*Constraints:*  
1 ≤ str.length ≤ 20

* **[output] string**

<https://codefights.com/challenge/KBmHkh6b7q68QBiHd?utm_source=featuredChallenge&utm_medium=email&utm_campaign=email_notification>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static string sortIt(string str)

{

char[] ch = str.ToCharArray();

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

{

for (int j = i + 1; j < ch.Length; j++)

{

if (char.ToLower( ch[i] ) > char.ToLower( ch[j]))

{

char temp = ch[i];

ch[i] = ch[j];

ch[j] = temp;

}

else if (char.ToLower(ch[i]) == char.ToLower(ch[j])

&& char.IsLower(ch[i]) && char.IsUpper(ch[j]))

{

char temp = ch[j];

ch[j] = ch[i];

ch[i] = temp;

}

}

}

return new string(ch);

}

static void Main(string[] args)

{

//string s = "Vaibhav";

//char[] ch = s.ToCharArray();

//Array.Sort(ch);

//Console.WriteLine(new string(ch));

string s = "Vaibhav";

//string s = "vV";

Console.WriteLine(sortIt(s));

Console.ReadLine();

}

}

}