26th jan 2023

FIRST:-

Case-specific Sorting of Strings

MediumAccuracy: **69.88%**Submissions: **33K+**Points: **4**

You can earn more Geekbits by participating in GFG Weekly Coding Contest

Given a string **S** consisting of **only uppercase** and **lowercase** characters. The task is to **sort** uppercase and lowercase letters **separately** such that if the i_{th} place in the original string had an Uppercase character then it should not have a lowercase character after being sorted and vice versa.

Example 1:

```
Input:
N = 12
S = defRTSersUXI
Output: deeIRSfrsTUX
Explanation: Sorted form of given string
with the same case of character as that
in original string is deeIRSfrsTUX
```

Example 2:

```
Input:
N = 6
S = srbDKi
Output: birDKs
Explanation: Sorted form of given string
with the same case of character will
result in output as birDKs.
```

Your Task:

You only need to complete the function **caseSort** that takes a string str and length of the string n and **returns** sorted **string**.

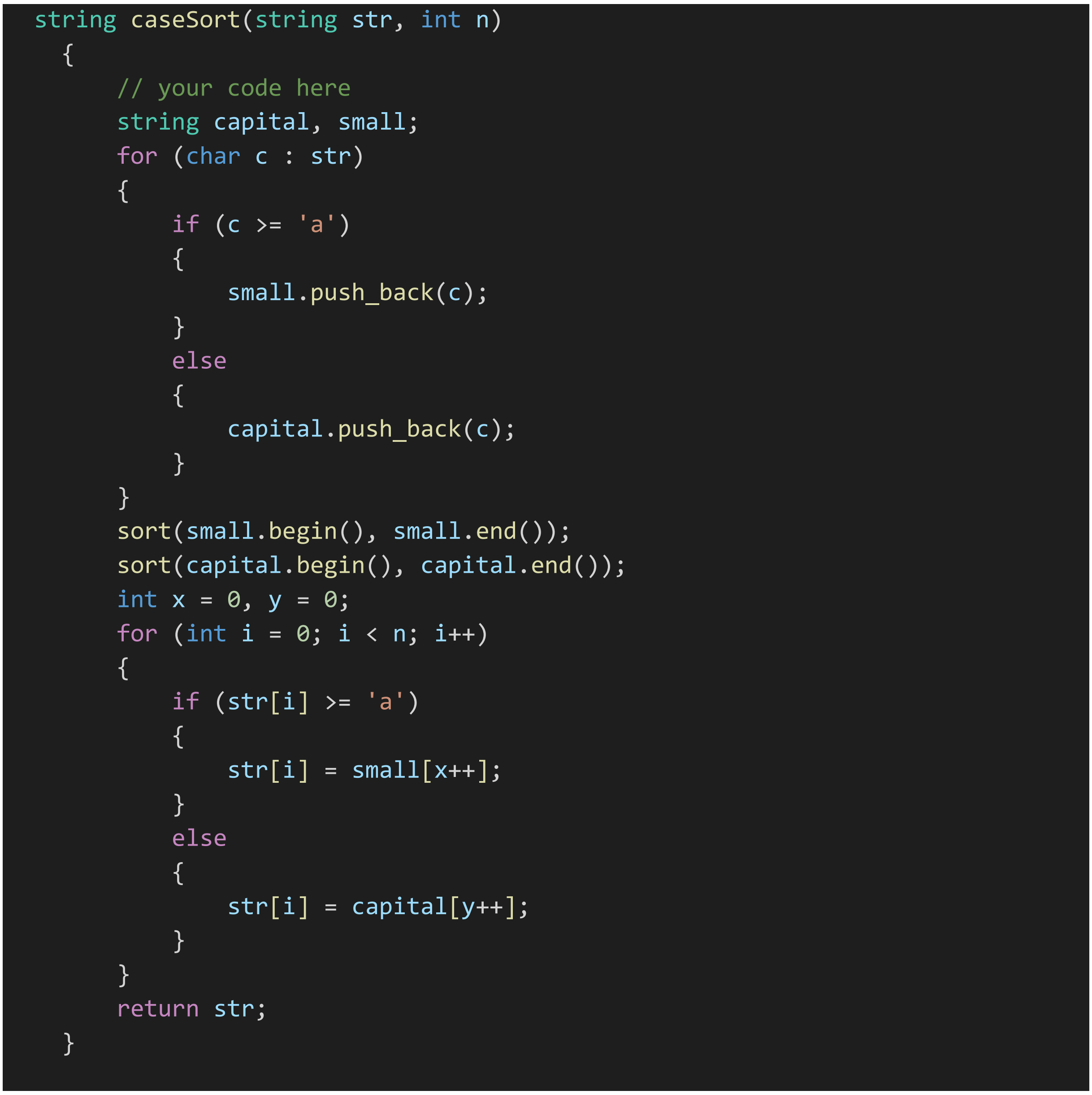
Expected Time Complexity: O(Nlog(N)).

Expected Auxiliary Space: O(N).

Constraints:

$1 \le N \le 10^5$

CODE SECTION:-



```
 string caseSort(string str, int n)
   {
       // your code here
       string capital, small;
       for (char c : str)
       {
           if (c >= 'a')
           {
               small.push_back(c);
           }
           else
           {
               capital.push_back(c);
           }
       }
       sort(small.begin(), small.end());
       sort(capital.begin(), capital.end());
       int x = 0, y = 0;
       for (int i = 0; i < n; i++)
       {
           if (str[i] >= 'a')
           {
               str[i] = small[x++];
           }
           else
           {
               str[i] = capital[y++];
           }
       }
       return str;
   }
```

-: DONE FOR TODAY :-