

Sorting Array of Strings

To sort a given array of strings into lexicographically increasing order or into an order in which the string with the lowest length appears first, a sorting function with a flag indicating the type of comparison strategy can be written. The disadvantage with doing so is having to rewrite the function for every new comparison strategy.

A better implementation would be to write a sorting function that accepts a pointer to the function that compares each pair of strings. Doing this will mean only passing a pointer to the sorting function with every new comparison strategy.

Given an array of strings, you need to implement a **string_sort** function which sorts the strings according to a comparison function, i.e, you need to implement the function :

```
void string_sort(const char **arr,const int cnt, int (*cmp_func)(const char* a, const char* b)){  
    }  
}
```

The arguments passed to this function are:

- an array of strings : *arr*
- length of string array: *count*
- pointer to the string comparison function: *cmp_func*

You also need to implement the following four string comparison functions:

1. **int lexicographic_sort(char*, char*)** to sort the strings in lexicographically non-decreasing order.
2. **int lexicographic_sort_reverse(char*, char*)** to sort the strings in lexicographically non-increasing order.
3. **int sort_by_number_of_distinct_characters(char*, char*)** to sort the strings in non-decreasing order of the number of distinct characters present in them. If two strings have the same number of distinct characters present in them, then the lexicographically smaller string should appear first.
4. **int sort_by_length(char*, char*)** to sort the strings in non-decreasing order of their lengths. If two strings have the same length, then the lexicographically smaller string should appear first.

Input Format

You just need to complete the function `string_sort` and implement the four string comparison functions.

Constraints

- $1 \leq \text{No. of Strings} \leq 50$
- $1 \leq \text{Total Length of all the strings} \leq 2500$
- You have to write your own sorting function and you cannot use the inbuilt *qsort* function
- The strings consists of lower-case English Alphabets only.

Output Format

The locked code-stub will check the logic of your code. The output consists of the strings sorted according to the four comparsion functions in the order mentioned in the problem statement.

Sample Input 0

```
4
wkue
goi
sbv
fekls
```

Sample Output 0

```
fekls
goi
sbv
wkue

wkue
sbv
goi
fekls

goi
sbv
wkue
fekls

goi
sbv
wkue
fekls
```