

Tutorial for Problem C - Anagram Test

All we need to do is to count the number of occurrences (or frequencies) of each letter in the two strings. If the number of occurrences of each letter is the same, then the two strings are anagrams of each other.

Why does this work? Well, if the number of occurrences of each letter in s and t is the same, then we can rearrange the letters of s to get t . For example, if $s = \text{abba}$ and $t = \text{baba}$, then we can rearrange the letters of s to get t .

This solution is $O(n)$.

Code

```
#include <stdio.h>
#include <string.h>

#define MAX_SIZE 100'000

char s[MAX_SIZE + 10];
char t[MAX_SIZE + 10];

int main() {

    scanf("%s %s", s, t);

    int s_cnt[26] = {};
    int s_len = strlen(s);
    for (int i = 0; i < s_len; ++i) {
        ++s_cnt[s[i] - 'a'];
    }

    int t_cnt[26] = {};
    int t_len = strlen(t);
    for (int i = 0; i < t_len; ++i) {
        ++t_cnt[t[i] - 'a'];
    }

    int is_anagram = 1;
    for (int i = 0; i < 26; ++i) {
        if (s_cnt[i] != t_cnt[i]) {
            is_anagram = 0;
            break;
        }
    }

    printf("%s\n", is_anagram ? "yes" : "no");
}
```

```
    return 0;  
}
```