CRACK THE HACK

Maximum Palindromes

Name:Swati S Shiriyannavar
USN:01FE15BEC237

Introduction:

Prefix sums, combinatorics is used to find the maximum palindromes of the given string in range of [l,r] where l is starting index and r is ending index of given string.Lets ignore the range of the string ,

Following algorithm is used to find number of maximum palindrome of the string.

Algorithm:

1) Approach A palindrome can be represented as **"str + t + reverse(str)"**. **Note**: **"t"** is empty for even length palindromic strings.
2) Calculate in how many ways "str" can be made and then multiply with "t" (number of single characters left out).
3) Let ci be the number of occurrences of character in the string. Consider the following cases:
   1. If ci is even. Then a half of every maximum palindrome will contain exactly letters fi = ci / 2.
   2.If ci is odd. Then a half of every maximum palindrome will contain exactly letters fi = ci – 1/ 2.
4) Let k be the number of odd ci. If k=0, the maximum palindromes length will be even; otherwise it will be odd and there will be exactly k possible middle letters i.e., we can set this letter to the middle of palindrome.
5) The number of permutations of n objects with n1 identical objects of type 1, n2 identical objects of type 2,, and n3 identical objects of type 3

is **n! / (n1! * n2! * n3!)** .
So here we have total number of characters as fa+fb+fa+…….+fy+fz .
So number of permutation is **(fa+fb+fa+…….+fy+fz+)! / fa! fb!…fy!fz!.**

6) Now If K is not 0, it's obvious that the answer is k * (fa+fb+fa+…….+fy+fz+)! / fa! fb!…fy!fz!

Example:

```
Input : str = "ababa"
Output: 2
Explanation :
palindromes of maximum of lenghts are  :
 "ababa", "baaab"

Input : str = "abababb"
Output: 4
Explanation :
palindromes of maximum of lenghts are  :
 "ababa", "baaab", "abbba", "babab"
```

Code:

C++ code

```
#include <iostream>
#include <memory.h>
using namespace std;
typedef long long ll;
const int N = 100001;
```

```
const int A = 'z' - 'a' + 1;
const ll MOD = (ll) 1e9 + 7;
ll power(ll x, ll y) {
    if (y == 0) {
        return 1;
    }
    if (y & 1) {
        return power(x, y - 1) * x % MOD;
     }
      else {
        ll tmp = power(x, y / 2);
        return tmp * tmp % MOD;
    }}
ll fact[N], rfact[N];
int n, q;char s[N];int cnt[N][A];
ll calc(int l, int r) {
    int sum = 0;
    int odd = 0;
    ll res = 1;
    for (int i = 0; i < A; i++) {
        int cur = cnt[r][i] - cnt[l - 1][i];
        sum += cur / 2;
        res = res * rfact[cur / 2] % MOD;
        if (cur % 2 == 1) {
            odd++;
        }
    }
    res = res * max(1, odd) % MOD;
```

```
    res = res * fact[sum] % MOD;
    return res;}
int main() {
    fact[0] = 1;
    rfact[0] = 1;
    for (int i = 1; i < N; i++) {
        fact[i] = fact[i - 1] * i % MOD;
        rfact[i] = power(fact[i], MOD - 2);
    }

    memset(cnt, 0, sizeof cnt);

    scanf("%s %d", s, &q);
    n = strlen(s);
    for (int i = 0; i < n; i++) {
        cnt[i + 1][s[i] - 'a']++;
    }
    for (int i = 0; i <= n; i++) {
        for (int j = 0; j < A; j++) {
            cnt[i][j] += cnt[i - 1][j];
        }
    }

    for (int i = 0; i < q; i++) {
        int l, r;
        scanf("%d%d", &l, &r);
        printf("%d\n", (int) calc(l, r));
    }}
```

References:

1) https://www.geeksforgeeks.org/count-maximum-length-palindromes-string/

2) https://www.hackerrank.com/challenges/maximum-palindromes/editorial

THANK YOU.