**Problem name:** Odd Substring

**Topic:** BitMasking

**Tags:** Bit Masking

**Level:** Hard

**Language used:** C++

**Problem Statement:**In the town of Dholakpur, there's a mysterious message encoded in a string. To decipher it, Chota Bheem needs to find all non-empty substrings where at least one character occurs an odd number of times.

Help Chota Bheem in finding the all non-empty substrings where at least one character occurs an odd number of times.

**Input Format:**

The first line of input will contain a single integer T, denoting the number of test cases.

Each test case consists of multiple lines of input.

The first line of each test case contains a string of lowercase english alphabets only.

**Output Format:**

For each test case, output on a new line the number of non-empty substrings where at least one character occurs an odd number of times.

**Constraints:**

**1<=T<=10**

**1<=N<=10^5**

**S contains lowercase english alphabets only**

**Sample Input 1:**

1

ababa

**Sample Output 1:**

13

**Explanation for Sample case 1**:

Total Valid Strings: "a","ab","aba","ababa","b","ba","bab","a","ab","aba","b","ba","a".

Hence total valid string are 13

**Sample Input 2:**1

ab

**Sample Output 2:**

**3**

**Explanation for Sample case 2**:

Total Valid Strings: "a","ab","b"

Hence total valid string are 3

**Code:**

#include <bits/stdc++.h>

using namespace std;

int main() {

  int t=1;

  cin>>t;

  while(t--)

  {

    string s;

   cin>>s;

    int n=s.size();

     int mask=0,ans=0;

     int cnt=0;

     map<int,int>m;

     m[0]=1;

      for(int i=0;i<n;i++)

      {

         mask ^= 1 << (s[i] - 'a');

         cnt+=m[mask];

         m[mask]++;

      }

      ans=(n\*(n+1))/2-cnt;

      cout<<ans<<endl;

  }

    return 0;

}

**One Compiler Link:**

[**https://onecompiler.com/cpp/422syujmc**](https://onecompiler.com/cpp/422syujmc)