My method

--------------------------------------

#include<bits/stdc++.h>

using namespace std;

int solve(string s){

unordered\_map<char,int> mp; // use index of char of string map<char,index>

int i=0;

int res = 0, n = s.size();

while(i<n){

char ch = s[i];

if(mp.find(ch)!=mp.end()){

int ans = mp.size();

res = max(res,ans);

int idx = mp[ch];

mp.clear();

i = idx+1;

}

else{

mp[ch] = i;

if(i+1==n){

int x = mp.size();

res = max(x,res);

}

i++;

}

}

return res;

}

int main(){

string s;

cin>>s;

cout<<solve(s);

}

---------------------------------------------------------------------------------------------

Sir method

#include<bits/stdc++.h>

using namespace std;

int solve(string s){

unordered\_map<char,int> mp; // use index of char of string map<char,index>

int i=-1,j=-1;

int res = 0;

while(true){

bool f1 = false, f2 = false;

int n = s.size();

while(i<n-1){

f1 = true;

i++;

char ch = s[i];

mp[ch]++;

if(mp[ch]>1){

break;

}

else{

int len = i-j;

res = max(res,len);

}

}

while(j<i){

f2 = true;

j++;

char ch = s[j];

mp[ch]--;

if(mp[ch]==1){

break;

}

}

if(f1==false && f2==false){

break;

}

}

return res;

}

int main(){

string s;

cin>>s;

cout<<solve(s);

}