#include<bits/stdc++.h>

using namespace std;

int pall(string &s, int start, int end){

int n = s.size();

vector<vector<int> > dp(n+1,vector<int> (n+1,-1));

string r = s;

reverse(s.begin(), s.end());

for(int i=0;i<=n;i++){

dp[i][0] = 0;

}

for(int i=0;i<=n;i++){

dp[0][i] = 0;

}

for(int i=1;i<=n;i++){

for(int j=1;j<=n;j++){

if(s[i-1]==r[j-1]){

dp[i][j] = 1 + dp[i-1][j-1];

}

else{

dp[i][j] = max(dp[i][j-1],dp[i-1][j]);

}

}

}

return dp[n][n];

}

int pall(string &s, int start, int end, vector<vector<int> > &dp){

if(start>end){

return 0;

}

if(start==end){

return dp[start][end] = 1;

}

if(dp[start][end]==-1){

if(s[start]==s[end]){

int sub = pall(s,start+1,end-1,dp);

dp[start][end] = 2+sub;

}

else{

int l = pall(s,start+1,end,dp);

int r = pall(s,start,end-1,dp);

dp[start][end] = max(l,r);

}

}

return dp[start][end];

}

int main(){

string s1;

cin>>s1;

int n = s1.size();

vector<vector<int> > dp(n+1,vector<int> (n+1,-1));

cout<<pall(s1,0,n-1);

}