АНАЛИЗ НА РЕШЕНИЕТО НА ЗАДАЧА ПАЛИНДРОМИ

Задачата решаваме с Динамично програмиране. Нека зададената дума е с дължина n – $a_0a_{1...}a_{n-1}$. Да означим с $T_{i,j}$ броя на палиндромите, които можем да получим със задраскване на някои от буквите в подниза с дължина j, започващ в a_i – $a_i\, a_{i+1\, ...}\, a_{\, i+j-1}$. Решението на задачата ще бъде точно $T_{0,n}$.

Очевидно е че всяко буква на низа е палиндром с дължина с дължина 1, затова $T_{i,1} = 1$, за i =0, 1, …, n – 1.

Ако $a_i \neq a_{i+1}$, тогава a_ia_{i+1} не е палиндром и в този низ палиндроми са само двете участващи букви. Затова, в този случай, $T_{i,2} = 2$. Ако $a_i = a_{i+1}$, тогава a_ia_{i+1} е палиндром, заедно с палиндромите на двете участващи букви. Затова, в този случай, $T_{i,2} = 3$.

Нека j >2. Ако $a_i \neq a_{i+j-1}$, тогава $T_{i,j} = T_{i,j-1} + T_{i+1,j-1} – T_{i+1,j-2}$, т.е. събираме броя на палиндромите с не повече от j – 1 букви в двата подинтервала с дължина j – 1: $a_i\, a_{i+1\, ...}\, a_{\, i+j}$ и $a_{i+1}a_{i+2\, ...}\, a_{\, i+j-1}$ и извадим от получената сума броя на палиндромите с не повече от j – 2 букви в подинтервала с дължина j – 2: $a_{i+1}a_{i+2\, ...}\, a_{\, i+j-2}$, които сме преброили два пъти при сумирането.

По различен е случаят когато $a_i = a_{i+j-1}$. Тогава не е необходимо да изваждаме $T_{i+1,j-2}$ от сумата $T_{i,j-1} + T_{i+1,j-1}$, защото всеки палиндром в подинтервала с дължина j – 2: $a_{i+1}a_{i+2\, ...}\, a_{\, i+j-2}$ ще направи нов палиндром започващ с a_i и завършващ с a_{i+j-1}. Освен това $a_i\, a_{i+j-1}$

Също е палиндром, получаващ се след задраскване на останалите букви. Затова в този случай $T_{i,j} = T_{i,j-1} + T_{i+1,j-1} + 1$.

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

using namespace std;
long long a[62][62];
int n; char s[65];
int main()
{
  int i,j;
  cin>>s;n=strlen(s);
  for(j=1;j<=n;j++)
   for(i=0;i<n+1-j;i++)
   { if(j==1) a[i][j]=1;
     else if(j==2)
     { if(s[i]==s[i+1]) a[i][j]=3;
       else a[i][j]=2;
     }
     else
     { a[i][j]=a[i][j-1]+a[i+1][j-1];
       if(s[i]!=s[i+j-1]) a[i][j]-=a[i+1][j-2];
       else a[i][j]+=1;
     }
   }
  cout<<a[0][n]<<endl;
  return 0;
}
```

Автор: Красимир Манев