

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  const n = 1000;
5
6  int mot_palindrome(char*text)
7  {
8      int i = 0,j = 0,nb_palin = 0;
9
10     while(i < strlen(text))
11     {
12         while(i < strlen(text) && (text[i]==' ' || text[i]=='.'))
13             i++;
14         j = i + 1;
15         while(j < strlen(text) && (text[j]>='a' && text[j]<='z' ||
16             text[j]>='A' && text[j]<='Z' ))
17             j++;
18         while(i <= j && text[j]==text[i])
19         {
20             i++;j--;
21         }
22         if(i > j)
23             nb_palin++;
24         while(i < strlen(text) && (text[i]>='a' && text[i]<='z' ||
25             text[i]>='A' && text[i]<='Z' ))
26             i++;
27     }
28     return nb_palin;
29 }
30 void optimise_blancs(char *text)
31 {
32     int i,j,s=0;
33
34     for(i=0,j=0;i<strlen(text);i++){
35         text[j] = text[i];
36         if( text[j] != ' ')
37         {
38             j++;s=0;
39         }
40         else if(text[j] == ' ' && s==0){s++;j++;}
41     }
42     text[j]='\0';
43 }
44
45 int recherche_mot(char *text, char *mot)
46 {
47     int i=0,j,comp = 0;char t[15];
48     while(i<strlen(text))
49     {
50         j=0;
51         while(i < strlen(text) && text[i]!=' ')
52             t[j++] = text[i++];
53
54         t[j]='\0';
55         if(strcmp(t,mot)==0)
56             comp++;
57         i++;
58     }
59     return comp;
60 }
61 int main()
62 {
63     char t[100],m[15];
64     strcpy(t," elle now yes ");
65
66     printf("nombre de palindrome =%d \n",mot_palindrome(t));
67     optimise_blancs(t);
68     puts(t);
69     printf("veuillez introduire un mot\n");
70     gets(m);
71     printf("nombre d'occurrence =%d \n",recherche_mot(t,m));

```

```
72     return 0;  
73 }
```