```
//palindrome  Checking || Input: Madam => Output: Yes
#include <stdio.h>
int strcmp1(char* s1,char* s2)
{
   int diff=0;
   while(*s1!='\0' && *s2!='\0')
   {
      if(*s1!=*s2)
        return *s1-*s2;
      s1++;
      s2++;
   }
   return diff;
}
int strlen1 (char *s)
{
   int l=0;
   while(*s++ != '\0')
   {
      l++;
   }
   printf("Length of the string is:%5d",l);
   return l;
}
char* strrev1(char* s)
{
   char *t;
   int i=0, len;
   len=strlen1(s);
   *(t+len)='\0';
   len--;
   while(*s != '\0')
   {
```

```
        *(t+len)=*s;
        len--;
        s++;
    }
    return t;
}
int main()
{
    char str[50], *str1, n;
    printf("\nEnter a string\n");
    scanf("%[^\n]s",str);

    puts("\nYou entered a string:\t"); puts(str);
    //printf("\nreverse of that string:\t%s",strrev1(str));

    if(!strcmp1(strrev1(str),str))
      puts("\n\nPALINDROME");
    else
      puts("\n\nNOT PALINDROME");
    return 0;
}
```

```
//signature
#include <stdio.h>
int count_space(char* s)
{
   int c=0;
   while(*s !='\0')
   {
      if(*s== ' ')
       c++;
      s++;
   }
   printf("\n%3d spaces are detected...\n",c);
   return c;
}
void mk_sign(char* s)
{
   int cs=count_space(s); //count no of spaces

   while(*s)
   {
      printf("%c. ",*s-32);
      while(*s!=32) //skip all chars untill a ' ' is found
        s++;
      cs--;    // reduce space counter
      if(cs==0)
      break;
      s++;  // skip the space
   }
   //printf("\n\n%s\n\n",*s);
   while(*s)
   {
      printf("%c",*s-32);
      s++;
```

```
    }
}
int main()
{
    char str[50]="ayan kumar dey";

    printf("\nEnter a string\n");
 //   scanf("%[^\n]s",str);


    puts("\nYou entered a string:\t"); puts(str);
    //printf("\nreverse of that string:\t%s",strrev1(str));

    mk_sign(str);
    return 0;
}
```

```
//word replacement
#include <stdio.h>
#include <string.h>
void replace (char* s, char* s1, char* s2)
{
   char temp[50];
   int i;
  while(*s)
  {
    //cut a word
    i=0;
    while(*s!=32)
    {
       temp[i++]=*s;
       s++;
    }
    temp[i]='\0'; //complete string
    //compare with s2 if found replace it else print it
    if(!strcmp(temp,s1))  // if good is found
      puts(s2);               // we'll print bad
    else
      puts(temp);
    s++;
  }
}
int main()
{
   char str[50]="Ayan is a good boy. Ratan is a good boy";
   char str1[50]="good", str2[50]="bad";
   replace(str, str1, str2);
   return 0;
}
```