```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>

//note- since in all these functions the array
name itself is passed , it acts as an pointer and
the changes are made on the array itself.Hence no
return value is given.
void sort_normal(char s1[][200],int tot_line)
    //normal sort taking the array with all the
lines from the files and total number of lines as
arguments
{
    int round,i,r;
                     //function variables
    char s[200];
    for(round=1; round<=tot_line; round++)
//repeating comparison of lines for "tot" number
of times
    {
        for(i=0; i<=(tot_line-1); i++)
        //comparing all the lines once
        {
            r=strcmp(s1[i],s1[i+1]);
  //comparing two lines and switching them if the
first characters of  first words of the two lines
are not in alphabetical order without ignoring
the cases
            if(r>0)
            {
                strcpy(s,s1[i]);
                strcpy(s1[i],s1[i+1]);
                strcpy(s1[i+1],s);
            }
        }
    }
}

void sort_icase(char s1[][200],int tot_line)
         //ignore the cases and sort taking the
array with all the lines from the files and total
number of lines as arguments
{
    int round,i,r;
    char s[200];
    for(round=1; round<=tot_line; round++)
    {
        for(i=0; i<=(tot_line-1); i++)
        {
            r=strcasecmp(s1[i],s1[i+1]);
       //comparing two lines and switching them
if the first characters of  first words of the
two lines are not in alphabetical order with
ignoring the cases
            if(r>0)
            {
                strcpy(s,s1[i]);
                strcpy(s1[i],s1[i+1]);
                strcpy(s1[i+1],s);
            }
        }
    }
}

void sort_reverse(char s1[][200],int tot_line)
      //reversed  sort taking the array with all
the lines from the files and total number of
lines as arguments
{
    int round,i,r;
    char s[200];
    for(round=1; round<=tot_line; round++)
    {
        for(i=0; i<=(tot_line-1); i++)
        {
            r=strcmp(s1[i],s1[i+1]);
    //comparing two lines and switching them if
the first characters of  first words of the two
lines are  in alphabetical order with ignoring
the cases
            if(r<0)
            {
                strcpy(s,s1[i]);
                strcpy(s1[i],s1[i+1]);
                strcpy(s1[i+1],s);
            }
        }
    }
}

void sort_icase_reverse(char s1[][200],int
tot_line)                   // ignore the cases and
reverse sort taking the array with all the lines
from the files and total number of lines as
arguments
{
    int round,i,r;
    char s[200];
    for(round=1; round<=tot_line; round++)
    {
        for(i=0; i<=(tot_line-1); i++)
        {
            r=strcasecmp(s1[i],s1[i+1]);
         //comparing two lines and switching them
if the first characters of  first words of the
two lines are in alphabetical order with ignoring
the cases
            if(r<0)
            {
                strcpy(s,s1[i]);
                strcpy(s1[i],s1[i+1]);
                strcpy(s1[i+1],s);
            }
        }
    }
```

```
}

void output_file(char s1[][200],FILE *fp1,int
tot_line)
{
        int i;
        for(i=1; i<=tot_line; i++)
        {
                fputs(s1[i],fp1);
//printing the sorted lines onto the output files
        }
}
```