Explanation:

The code starts by opening the file "file.csv" in read mode using the fopen() function and assigns the file pointer to the variable "fp". Next, the code uses the fscanf() function to read the first line of the file, which is stored in the variable "remove", and discards it. This is done because the first line of the file is the header and not part of the data that needs to be processed.

The code then enters a while loop that runs until the end of the file is reached. Inside the loop, the fscanf() function is used to read the data from the file, with the format specifiers "%[^,],%[^,],%[^,],%[^,],%[^,],%[^,],%[^,],%[^,],%[^\n]\n" indicating that the data is separated by commas and a newline character at the end of each row. The data read from the file is then stored in the struct array "data" at the current index "i". The index "i" is incremented after each iteration of the loop so that the next data read is stored in the next element of the array.

After the while loop, the code enters another nested loop that sorts the data in the struct array "data" by the "loc1" field using the strcmpi() function and a bubble sort algorithm. The outer loop runs for 3939-1 iterations, and the inner loop runs for 3939-j-1 iterations. If the strcmpi() function returns a value greater than 0, the elements at the current index "k" and "k+1" are swapped.

Finally, the code enters another loop that runs for 3939 iterations, and it prints the data stored in the struct array "data" for each iteration. Then the file is closed using the fclose() function and the program exits with a return value of 0.

Flowchart:

Diagram

Description automatically generated