#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define mxx 1024

int myFunc(const void \*a, const void \*b)

{

return strcmp(\*(const char \*\*)a, \*(const char \*\*)b);

}

int main()

{

FILE \*ifile = fopen("listings.csv", "r");

if (ifile == NULL)

{

perror("Error");

return 1;

}

char line[mxx];

char \*\*f1 = NULL;

char \*\*f2 = NULL;

int mxf = 100;

int f1count = 0;

int f2count = 0;

f1 = (char \*\*)malloc(mxf \* sizeof(char \*));

f2 = (char \*\*)malloc(mxf \* sizeof(char \*));

if (f1 == NULL || f2 == NULL) {

perror("Memory cannot be allocated");

fclose(ifile);

return 1;

}

while (fgets(line, sizeof(line), ifile)) {

if (f1count >= mxf || f2count >= mxf) {

mxf \*= 2;

f1 = (char \*\*)realloc(f1, mxf \* sizeof(char \*));

f2 = (char \*\*)realloc(f2, mxf \* sizeof(char \*));

if (f1 == NULL || f2 == NULL) {

perror("Memory cannot be allocated");

fclose(ifile);

return 1;

}

}

char \*token = strtok(line, ",");

int f\_idx = 0;

while (token != NULL) {

if (f\_idx == 2) {

f1[f1count] = strdup(token);

f1count++;

} else if (f\_idx == 8) {

f2[f2count] = strdup(token);

f2count++;

}

token = strtok(NULL, ",");

f\_idx++;

}

}

fclose(ifile);

qsort(f1, f1count, sizeof(char \*), myFunc);

FILE \*opf1 = fopen("c1.csv", "w");

if (opf1 == NULL) {

perror("Error in f1");

return 1;

}

for (int i = 0; i < f1count; i++) {

fputs(f1[i], opf1);

fputs("\n", opf1);

free(f1[i]);

}

fclose(opf1);

free(f1);

qsort(f2, f2count, sizeof(char \*), myFunc);

FILE \*opf2 = fopen("c2.csv", "w");

if (opf2 == NULL) {

perror("Error in f2");

return 1;

}

for (int i = 0; i < f2count; i++) {

fputs(f2[i], opf2);

fputs("\n", opf2);

free(f2[i]);

}

fclose(opf2);

free(f2);

return 0;

}