```
//Name: Mehmet Fatih Çelik
//ID: 2385268

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

struct record{
	char name[15];
	int deaths, recovered, activeCases, totalCases;
	double population;
};

int Load_CoronavirusTable(struct record*, char[]);
void Display_CoronavirusTable(struct record*, int);
int Search(struct record*, int, char*);
void Sort(struct record*, int);

int main(int argc, char *argv[]){
	struct record *CoronavirusTable;
	int fileFlag = 1;
	char name[15];
	strcpy(name,argv[1]);
	while(fileFlag){
		if(!strcmp(name,"corona.txt"))
			fileFlag = 0;
		else{
			printf("This file does not exist, please enter again: ");
			scanf("%s",&name);
		}
	}
```

```
FILE *inFile;
inFile = fopen(name,"r");
if (inFile == NULL){
	printf("Error occured while reading the file!");
	exit(1);
}

int size = 0;
char line[1024];
while((fscanf(inFile,"%[^\n]\n",line))!=EOF) //for calculating the number of lines in the file
	size++;

fclose(inFile);

CoronavirusTable = (struct record*)malloc(size*sizeof(struct record));
if(CoronavirusTable == NULL){
	printf("Error occured while allocating the memory!\n");
	exit(1);
}

size = Load_CoronavirusTable(CoronavirusTable, name);

printf("Coronavirus records file (corona.txt) has been successully loaded!\n");

Display_CoronavirusTable(CoronavirusTable, size);

int option;
do{
	fflush(stdin);
	printf("\nPress 1 for search, 2 for sort and 3 for exit: ");
	scanf("%d",&option);
```

```
		if(option == 1){
			char name[15];
			printf("\nEnter the name of the country: ");
			scanf("%s",&name);

			int position, flag = 1;
			do{
				position = Search(CoronavirusTable, size, name);
				if(position != -1)
					flag = 0;
				else{
					printf("That country is unknown! Please try again!\n");
					printf("Enter the name of the country: ");
					scanf("%s",&name);
				}
			}while(flag);

			printf("\n%s with %.0lf population has %d total cases (%d deaths, %d recovered and %d active cases)\n",CoronavirusTable[position].name,CoronavirusTable[position].population,CoronavirusTable[position].totalCases,CoronavirusTable[position].deaths,CoronavirusTable[position].recovered,CoronavirusTable[position].activeCases);
		}

		else if(option == 2)
			Sort(CoronavirusTable, size);
		else{
			if(option == 3)
				printf("\nBye!");
			else
				printf("Please enter a valid choice!\n");
```

```
                }


        }while(option != 3);


        return 0;
}


int Load_CoronavirusTable(struct record *table, char name[]){
        int i=0;
        FILE *inFile;
        inFile = fopen(name,"r");
        if (inFile == NULL){
                printf("Error occured while reading the file!");
                exit(1);
        }


        while(fscanf(inFile,"%s %d %d %d
%lf\n",table[i].name,&table[i].deaths,&table[i].recovered,&table[i].activeCases,&table[i].population)
!= EOF)
                i++;


        fclose(inFile);


        return i;
}


void Display_CoronavirusTable(struct record *table, int size){
        double totalCases;
        int i;
        static int a = 0; /* We display this sentence only first we load the file, after the sorting
according to sample output,
```

```
	we do not display this, so as we learnt in the class, I used static int to retain the value, when we load, it will be displayed,

	and incremented by 1, and after the sorting, we wont be displaying the sentence.

	*/

	if(a == 0){

		printf("Following records have been loaded:\n");

		a++;

	}

	printf("\nCountry\t\tDeaths\t\tRecovered\tActive Cases\tPopulation\tTotal Cases\n");


	for(i=0;i<size;i++){

		table[i].totalCases = table[i].deaths+table[i].recovered+table[i].activeCases;

		printf("%-16s%-16d%-16d%-16d%-16.0lf%-16d\n",table[i].name,table[i].deaths,table[i].recovered,table[i].activeCases,table[i].population,table[i].totalCases);

	}

}


int Search(struct record *table, int size, char *name){

	int i, position = -1;


	for(i=0;i<size;i++){

		if(strcmp(table[i].name,name) == 0)

			position = i;

	}


	return position;

}


void Sort(struct record *table, int size){

	char opSort;

	fflush(stdin);
```

```
	printf("Sort by (T: total cases, A: active cases): ");
	scanf("%c",&opSort);



	int i,j;
	struct record temp;
	if(opSort == 'T'){
		for(i=1;i<size;i++){
			temp = table[i];

			for(j=i; j>0 && temp.totalCases > table[j-1].totalCases; j--)
				table[j] = table[j-1];

			table[j] = temp;
		}
	}
	else if(opSort == 'A'){
		for(i=1;i<size;i++){
			temp = table[i];

			for(j=i; j>0 && temp.activeCases > table[j-1].activeCases; j--)
				table[j] = table[j-1];

			table[j] = temp;
		}
	}

	Display_CoronavirusTable(table, size);
}
```