

CMPE 252 C Programming, Spring 2023**HOMEWORK 3****20.05.2023, Saturday****Due Date: 04.06.2023, Sunday, 23:59****This is an individual work. No team work is allowed.****Similarity check will be applied to submitted codes.**

In this assignment, you are going to write a C program that reads, processes, and displays attendance records in a Zoom meeting report provided as a txt file.

Getting to know input file:

A sample Zoom meeting report (**input1.txt**) is given as follows:

```
Join Time,Leave Time,Duration(Minutes),Name,Email
02/28/2020 10:54:38,02/28/2020 12:56:38,122.0,Feyza Nur Duman,fnur.duman@tedu.edu.tr
02/28/2020 10:57:09,02/28/2020 10:58:28,2.0,Eda Nur YILMAZ,enur.yilmaz@tedu.edu.tr
02/28/2020 10:59:19,02/28/2020 12:56:28,118.0,Serhat Gokhan YAMAC,sgokhan.yamac@tedu.edu.tr
02/28/2020 10:59:27,02/28/2020 12:56:11,117.0,Duru Yamac,
02/28/2020 10:59:28,02/28/2020 12:56:38,118.0,Asya Hale Gencel,ahale.gencel@tedu.edu.tr
02/28/2020 10:59:30,02/28/2020 12:33:32,95.0,Eda Nur YILMAZ,enur.yilmaz@tedu.edu.tr
02/28/2020 11:00:12,02/28/2020 11:00:24,1.0,Taner Ceyhun,taner.ceyhun@tedu.edu.tr
02/28/2020 11:00:33,02/28/2020 12:21:07,81.0,Sevil TERZI,sevil.terzi@tedu.edu.tr
02/28/2020 11:00:30,02/28/2020 12:56:26,116.0,Taner Ceyhun,taner.ceyhun@tedu.edu.tr
02/28/2020 10:58:44,02/28/2020 12:56:29,118.0,Ilhan Tekin,
02/28/2020 10:56:49,02/28/2020 12:56:38,120.0,Aytekin Safak,
02/28/2020 12:27:50,02/28/2020 12:55:37,28.0,Sevil Terzi,
02/28/2020 12:58:30,02/28/2020 12:59:30,1.0,Sevil TERZI,sevil.terzi@tedu.edu.tr
```

The first line of the input file is a header line which includes title of each column of the data file:

- Join Time: Date and time when participants join the meeting
- Leave Time: Date and time when participants leave the meeting
- Duration: Duration (in terms of minutes) passed between join and leave time
- Name: Name & surname
- Email: Email address

Each of the remaining lines includes a record for a connection of an attendee to the meeting where attendees are uniquely recognized by their name & surname. Name of an attendee might include more than one word but surname is always a single word at the end of name & surname.

Name & surname refer to the same attendee if they are the same without considering uppercase/lowercase distinction of the letters. For example:

Eda Nur Yilmaz and EDA NUR YILMAZ refer to the same attendee.

Eda Nur Yilmaz and Eda Nur YILMAZ refer to the same attendee.

Of course, order and number of the words in name & surname matters. For example:

Eda Nur Yilmaz and Nur Eda Yilmaz refer to different attendees.

EDA NUR YILMAZ and EDA YILMAZ refer to different attendees.

Assume that there is a single space character between words in name & surname column of input file. For example:

EDA NUR YILMAZ and EDANUR YILMAZ refer to different attendees.

Of course, if there is a difference between surnames or names, they refer to different attendees. For example:

EDANUR YILMAZ and EDA YILMAZ refer to different attendees.

EDA YILMAZ and EDA YILMAZCAN refer to different attendees.

Notice that it is possible that e-mail address is not provided in some records.

Also, notice that there might be more than one record for some of attendees as they leave and join the meeting multiple times. Assume that there is no time overlap between multiple connections made by each attendee. That means each attendee has at most one connection at each moment of the meeting.

Note: The number of test cases containing multiple connections situation will be less than the others, if you do not do this part, you will not get full points.

Assume that duration is always provided for each record and it is compatible with the difference between join and leave time so you will consider duration of each connection as the number at the middle of its corresponding line (you do not need to compute duration of each connection from join and leave time).

Assume that the number of attendees in a Zoom session is not greater than 500.

What the program is expected to do:

Your program is supposed to do the following tasks:

- Read input file so that, for each connection, store name & surname of the corresponding attendee along with duration of the connection.
- If there are multiple connections made by the same attendee:

- Update duration of the first connection by adding durations of the remaining connections.
 - Keep the first connection (whose duration is updated) and remove the remaining connections.
- Write surname, name of attendees along with duration of their total connection
 - If *sortedAnswer=no*, the order of attendees in the output file will be in their original order in the input file.
 - If *sortedAnswer=yes* is provided (as sorted), the order of attendees in the output file will be based on alphabetical order of their surname & name.

Sample Runs:

Sample run 1:

Enter the input file name: *input1.txt*

Sort the results? (yes/no): *yes*

Ceyhun Taner 117.00
Duman Feyza Nur 122.00
Gencel Asya Hale 118.00
Safak Aytekin 120.00
Tekin Ilhan 118.00
TERZI Sevil 110.00
Yamac Duru 117.00
YAMAC Serhat Gokhan 118.00
YILMAZ Eda Nur 97.00

Sample run 2:

Enter the input file name: *input1.txt*

Sort the results? (yes/no): *no*

Duman Feyza Nur 122.00
YILMAZ Eda Nur 97.00
YAMAC Serhat Gokhan 118.00

Yamac Duru 117.00

Gencel Asya Hale 118.00

Ceyhun Taner 117.00

TERZI Sevil 110.00

Tekin Ilhan 118.00

Safak Aytekin 120.00

Sample run 3:

Enter the input file name: input2.txt

Sort the results? (yes/no): yes

Ceyhun Taner 61.00

Gencel Asya Hale 118.00

TERZI Sevil 85.00

YAMAC Serhat Gokhan 118.00

Yilmaz Eda Nur 95.00

Sample run 4:

Enter the input file name: input2.txt

Sort the results? (yes/no): no

YAMAC Serhat Gokhan 118.00

Gencel Asya Hale 118.00

Yilmaz Eda Nur 95.00

Ceyhun Taner 61.00

TERZI Sevil 85.00

Hints

You can use any available string sorting algorithm in your source code in the literature, but your source code is expected to be comprised by only 1 source file (so all your functions including the sorting function should be in the same file).

In this homework, you are mainly expected to use your skills from Chapter 8 Strings of the book.

Check out string.h library functions.

Check also character functions available in ctype.h.

You may also consider using fgets, fputs functions under stdio.h, if applicable to your solution.

Grading

For grading, your executables will be executed using VPL from LMS.

The debug test cases are provided in LMS VPL and, also, in the following zip file in Moodle:
CMPE252_HW3_debug_test_cases.zip