CS 241L – Data Organization
Spring 2022

Project 7

Total points: 60

Due on Monday, 5/9/22 (not accepted after 5/11/22)

In this project you will write a C program to determine the sequence at which patients are seen at an Urgent Care clinic.

Files are available by copying from my directory. See paths in Learn.

(1) The input file: `arrivals.log`

Patients can arrive at the clinic as early as 7:30 am and as late as 5:00 pm (17h00). There are no appointments, so the arrivals happen at random times. A list of all patients arriving at the clinic on a particular day is shown in the file `arrivals.log`.

Patients are given a unique ID according to the time of arrival. When they arrive, they are triaged and their ID, arrival time, age, and pain level (on a scale from 1-10) are entered in a log in this order (the `arrivals.log` file), with a patient per line. Times are given in a 24h format.

(2) The clinic

Patients are seen every 15 min, starting at 7:45 am. The clinic closes after the last patient is seen. The closing time varies depending on the number of patients that come in a single day.

Patients are not seen in a first-come first-served basis, but are prioritized depending on a few factors:

1) Patients who have a pain level of 10 are seen first
2) For patients who have pain levels of 9 or lower:
    a. Patients with higher pain level are seen first
    b. If patients have the same pain level, the oldest patient is seen first
    c. If two or more patients have the same pain level and age, the patient who arrived earlier is seen first
    d. To prevent long wait times, if patients have been waiting for longer than 2h, they become first priority regardless of their pain level and age.

- *Optional (no points): If you like statistics and would like an extra challenge: what happens to the distribution of waiting times when the priority in (d) changes from 2h to 1h? And to 3h? What happens to the average wait time per patient?*

(3) The C Program: `createList`

You will create an executable called `createList` from two source files: `main.c`, which contains the main function, and `list.c`, which contains functions written by you. All declarations should be in a separate header file `header.h`.

In this project, you will **not** use stdin and stdout redirection. Instead, all files must be read and written from inside the program. The file names are entered as command line arguments in the form:

```
./createList arrivals.log mylist.txt
```

Where `arrivals.log` is the input file, and `mylist.txt` is the output file.

You are given an output file, `patientList.txt`. There should be no differences in a diff test between `mylist.txt` and `patientList.txt`.

What to submit:

The files:

```
list.c
main.c
header.h
mylist.txt
```

Note: Submit a single zip file named LastNameFirstName_project7.

Grading Rubric:

- 1 pt: Your C program does not start with a comment on top of the file with your name and description of the program.
- 1 pt: Your C program compiles with the `–ansi –pedantic –Wall` options with errors or warnings.
- 1 pt: Your C program does not follow the class coding standards, including function comments.
- 2 pts: You do not follow all directions in this PDF.
+ 20 pts: Your program runs without errors or memory leaks in Valgrind.
+ 20 pts: Your program produces the correct output for `patientList.txt`
+ 20 pts: Your program produces the correct output for a mystery grader file that contains the arrival logs from a different day.