

Sabanci University

Faculty of Engineering and Natural Sciences
CS204 Advanced Programming
Spring 2019

Homework 2 – Formula 1 Qualifier Ranking with Linked Lists

Due: 6/3/2019, Wednesday, 21:00

PLEASE NOTE:

Your program should be a robust one such that you have to consider all relevant programmer mistakes and extreme cases; you are expected to take actions accordingly!

You can NOT collaborate with your friends and discuss solutions. You have to write down the code on your own. Plagiarism and homework trading will not be tolerated!

Introduction

In this homework, you are asked to implement a program that calculates the starting positions of Formula 1 drivers. This program must use a *Linked List* structure to store names of the drivers and their best lap times in a sorted fashion. Driver names and lap times are going to be read from a text file. The program details will be explained in the subsequent sections.

The Data Structure to be Used

In this homework, you **must** represent drivers and their best lap times as a *linked list* (regular one-way linked list). As a result, you are going to implement one node type that stores a list item with the **driver's name** and his **best lap time** recorded. **You are not allowed to use arrays, vectors and similar containers (including extra files) in this homework; all data must be stored and processed within the linked list.**

The Program Flow

Your program is going to start with getting an input from the user regarding the name of the file that contains qualification information. After getting the name of the file, the program must check whether the file has opened correctly. If not, another file name will be required from the user until a correct file name is entered.

After successfully opening the file, your program is going to start storing the best lap times of each driver by reading the file line by line. A sample file is shown below:

Fernando_Alonso	75402
Kimi_Raikkonen	75173
Ayrton_Senna	70560
Ayrton_Senna	77484
James_Hunt	75308
Fernando_Alonso	88005
Fernando_Alonso	70655
James_Hunt	88770
Ayrton_Senna	75622
James_Hunt	84596
James_Hunt	82232
Sebastian_Vettel	84514
Fernando_Alonso	93280
James_Hunt	92556
Ayrton_Senna	87909
Ayrton_Senna	97530
Fernando_Alonso	82716
Ayrton_Senna	97755
James_Hunt	97733
Ayrton_Senna	88975
Ayrton_Senna	86534
Fernando_Alonso	95724
James_Hunt	93512
Kimi_Raikkonen	78518
Fernando_Alonso	84235
Kimi_Raikkonen	77317
Sebastian_Vettel	88160
Ayrton_Senna	76489
Fernando_Alonso	92164
Sebastian_Vettel	78635

Each line in the file contains two pieces of information regarding a lap. The first word in the file is going to be the name of the driver. And the second one is going to be the lap time in milliseconds (to be read as positive integer). As seen in the example above, there can be any number of spaces between the name and the lap time of the driver. You can assume the file contains correct inputs so no input checks are required for the content of the file. As you see from the sample file above, a particular driver can be listed several times with various lap times.

The driver with the smallest lap time is going to start the race in the first place and driver with the largest lap time will start the race in the last place, and other drivers will be placed in the in the ascending order of their best lap times. If two or more drivers record the same time, first one examined by the program is going to start the race ahead.

These rules require your program to maintain the linked list in an ascending sorted fashion according to best (smallest) lap times of the drivers. In the list, a driver is going to be represented by a single node. After reading a line from the file, you have basically two options; either that driver exists in the linked list or not. If currently there is no node with a driver's name in the list, then a new node needs to be created and added to the linked list in a proper position to keep it sorted. If a node with that driver's name exists in the list, and the existing best lap time stored in that node is larger than the currently read one, your program is going to update that driver's lap time of the existing node. After that, the position of the driver's node may change in the list since you have to keep the list always sorted.

It is strictly forbidden the process the input file more than once. That means, do NOT even think of finding the smallest lap time of a driver by processing the file for him, rather than updating the linked list after each lap of that driver. And do not forget that you are not allowed to use extra helper containers. Thus, the only way is to do all of the data processing within the linked list. Any attempts to bypass this for the sake of simplicity will be penalized.

For each lap completed, your program is going to output who completed the lap in how many milliseconds. Moreover, after each lap, that driver's current personal best lap time and his current position should also be displayed. After the program is finished reading and processing the file, the starting positions of all of the drivers will be displayed on the screen in sorted manner. This latter output is actually the content of the linked list at the end of the program.

Sample Runs

Sample runs are given below, but these are not comprehensive, therefore you have to consider **all possible cases** to get full mark.

The sample input files are provided in the .zip package of this homework.

Sample Run 1: *t1.txt*

Fernando_Alonso	75402
Kimi_Raikkonen	70655
Ayrton_Senna	70560
Ayrton_Senna	77484
James_Hunt	75308
Fernando_Alonso	88005
Fernando_Alonso	70655
James_Hunt	88770
Ayrton_Senna	75622
James_Hunt	84596
James_Hunt	82232
Sebastian_Vettel	84514
Fernando_Alonso	93280
James_Hunt	92556
Ayrton_Senna	87909
Ayrton_Senna	97530
Fernando_Alonso	82716
Ayrton_Senna	97755
James_Hunt	97733
Ayrton_Senna	88975
Ayrton_Senna	86534
Fernando_Alonso	95724
James_Hunt	93512
Kimi_Raikkonen	78518
Fernando_Alonso	84235
Kimi_Raikkonen	77317
Sebastian_Vettel	88160
Ayrton_Senna	76489
Fernando_Alonso	92164
Sebastian_Vettel	78635

Please enter a file name.
t.txt
Unable to open file t.txt
Please enter a different file name.
t1.txt
Successfully opened file t1.txt

#####

Qualifying Laps:

#####

Fernando_Alonso completed the lap in 75402 milliseconds
Fernando_Alonso: current personal best is 75402; current position is 1
Kimi_Raikkonen completed the lap in 70655 milliseconds
Kimi_Raikkonen: current personal best is 70655; current position is 1
Ayrton_Senna completed the lap in 70560 milliseconds
Ayrton_Senna: current personal best is 70560; current position is 1
Ayrton_Senna completed the lap in 77484 milliseconds
Ayrton_Senna: current personal best is 70560; current position is 1
James_Hunt completed the lap in 75308 milliseconds
James_Hunt: current personal best is 75308; current position is 3
Fernando_Alonso completed the lap in 88005 milliseconds
Fernando_Alonso: current personal best is 75402; current position is 4
Fernando_Alonso completed the lap in 70655 milliseconds
Fernando_Alonso: current personal best is 70655; current position is 3
James_Hunt completed the lap in 88770 milliseconds
James_Hunt: current personal best is 75308; current position is 4
Ayrton_Senna completed the lap in 75622 milliseconds
Ayrton_Senna: current personal best is 70560; current position is 1
James_Hunt completed the lap in 84596 milliseconds
James_Hunt: current personal best is 75308; current position is 4
James_Hunt completed the lap in 82232 milliseconds
James_Hunt: current personal best is 75308; current position is 4
Sebastian_Vettel completed the lap in 84514 milliseconds
Sebastian_Vettel: current personal best is 84514; current position is 5
Fernando_Alonso completed the lap in 93280 milliseconds
Fernando_Alonso: current personal best is 70655; current position is 3
James_Hunt completed the lap in 92556 milliseconds
James_Hunt: current personal best is 75308; current position is 4
Ayrton_Senna completed the lap in 87909 milliseconds
Ayrton_Senna: current personal best is 70560; current position is 1
Ayrton_Senna completed the lap in 97530 milliseconds
Ayrton_Senna: current personal best is 70560; current position is 1
Fernando_Alonso completed the lap in 82716 milliseconds
Fernando_Alonso: current personal best is 70655; current position is 3
Ayrton_Senna completed the lap in 97755 milliseconds
Ayrton_Senna: current personal best is 70560; current position is 1
James_Hunt completed the lap in 97733 milliseconds
James_Hunt: current personal best is 75308; current position is 4
Ayrton_Senna completed the lap in 88975 milliseconds
Ayrton_Senna: current personal best is 70560; current position is 1
Ayrton_Senna completed the lap in 86534 milliseconds
Ayrton_Senna: current personal best is 70560; current position is 1
Fernando_Alonso completed the lap in 95724 milliseconds
Fernando_Alonso: current personal best is 70655; current position is 3
James_Hunt completed the lap in 93512 milliseconds
James_Hunt: current personal best is 75308; current position is 4
Kimi_Raikkonen completed the lap in 78518 milliseconds
Kimi_Raikkonen: current personal best is 70655; current position is 2
Fernando_Alonso completed the lap in 84235 milliseconds
Fernando_Alonso: current personal best is 70655; current position is 3
Kimi_Raikkonen completed the lap in 77317 milliseconds
Kimi_Raikkonen: current personal best is 70655; current position is 2
Sebastian_Vettel completed the lap in 88160 milliseconds
Sebastian_Vettel: current personal best is 84514; current position is 5
Ayrton_Senna completed the lap in 76489 milliseconds
Ayrton_Senna: current personal best is 70560; current position is 1
Fernando_Alonso completed the lap in 92164 milliseconds
Fernando_Alonso: current personal best is 70655; current position is 3

Sebastian_Vettel completed the lap in 78635 milliseconds
Sebastian_Vettel: current personal best is 78635; current position is 5

```
#####  
Results:  
#####  
1. Ayrton_Senna 70560  
2. Kimi_Raikkonen 70655  
3. Fernando_Alonso 70655  
4. James_Hunt 75308  
5. Sebastian_Vettel 78635
```

Sample Run 2: *t2.txt*

Michael_Schumacher	78718
Alain_Prost	85729
Ayrton_Senna	82385
Juan_Manuel_Fangio	82045
Juan_Manuel_Fangio	88185
James_Hunt	73638
Juan_Manuel_Fangio	93603
Alain_Prost	89437
Ayrton_Senna	77191
James_Hunt	80564
Juan_Manuel_Fangio	81698
James_Hunt	73062
Ayrton_Senna	74972
Niki_Lauda	72928
Alain_Prost	91068
James_Hunt	87926
Alain_Prost	93782
Michael_Schumacher	80827
Ayrton_Senna	89408
Ayrton_Senna	89486

Please enter a file name.
t2.txt
Successfully opened file t2.txt

```
#####  
Qualifying Laps:  
#####  
Michael_Schumacher completed the lap in 78718 milliseconds  
Michael_Schumacher: current personal best is 78718; current position is 1  
Alain_Prost completed the lap in 85729 milliseconds  
Alain_Prost: current personal best is 85729; current position is 2  
Ayrton_Senna completed the lap in 82385 milliseconds  
Ayrton_Senna: current personal best is 82385; current position is 2  
Juan_Manuel_Fangio completed the lap in 82045 milliseconds  
Juan_Manuel_Fangio: current personal best is 82045; current position is 2  
Juan_Manuel_Fangio completed the lap in 88185 milliseconds  
Juan_Manuel_Fangio: current personal best is 82045; current position is 2  
James_Hunt completed the lap in 73638 milliseconds  
James_Hunt: current personal best is 73638; current position is 1  
Juan_Manuel_Fangio completed the lap in 93603 milliseconds  
Juan_Manuel_Fangio: current personal best is 82045; current position is 3  
Alain_Prost completed the lap in 89437 milliseconds  
Alain_Prost: current personal best is 85729; current position is 5  
Ayrton_Senna completed the lap in 77191 milliseconds  
Ayrton_Senna: current personal best is 77191; current position is 2  
James_Hunt completed the lap in 80564 milliseconds  
James_Hunt: current personal best is 73638; current position is 1  
Juan_Manuel_Fangio completed the lap in 81698 milliseconds  
Juan_Manuel_Fangio: current personal best is 81698; current position is 4
```

```

James_Hunt completed the lap in 73062 milliseconds
James_Hunt: current personal best is 73062; current position is 1
Ayrton_Senna completed the lap in 74972 milliseconds
Ayrton_Senna: current personal best is 74972; current position is 2
Niki_Lauda completed the lap in 72928 milliseconds
Niki_Lauda: current personal best is 72928; current position is 1
Alain_Prost completed the lap in 91068 milliseconds
Alain_Prost: current personal best is 85729; current position is 6
James_Hunt completed the lap in 87926 milliseconds
James_Hunt: current personal best is 73062; current position is 2
Alain_Prost completed the lap in 93782 milliseconds
Alain_Prost: current personal best is 85729; current position is 6
Michael_Schumacher completed the lap in 80827 milliseconds
Michael_Schumacher: current personal best is 78718; current position is 4
Ayrton_Senna completed the lap in 89408 milliseconds
Ayrton_Senna: current personal best is 74972; current position is 3
Ayrton_Senna completed the lap in 89486 milliseconds
Ayrton_Senna: current personal best is 74972; current position is 3

```

```

#####
Results:
#####
1. Niki_Lauda 72928
2. James_Hunt 73062
3. Ayrton_Senna 74972
4. Michael_Schumacher 78718
5. Juan_Manuel_Fangio 81698
6. Alain_Prost 85729

```

Sample Run 3: *t3.txt*

```

Fernando_Alonso      84030
James_Hunt 88197
Kimi_Raikkonen 79282
Ayrton_Senna      92750
Kimi_Raikkonen 77201
Kimi_Raikkonen 82502
Sebastian_Vettel      81741
Sebastian_Vettel 73966
James_Hunt      80994
Ayrton_Senna 71047
James_Hunt      80649
James_Hunt      71841
Ayrton_Senna      99303
Fernando_Alonso      95106
Kimi_Raikkonen 97888
Sebastian_Vettel 90617
Ayrton_Senna 81740
Ayrton_Senna 82619
James_Hunt 75975
Sebastian_Vettel 99538
James_Hunt 70664
Kimi_Raikkonen 99347
Sebastian_Vettel 97309
Fernando_Alonso 96873
James_Hunt 86078
Kimi_Raikkonen 90040
Fernando_Alonso 76980

```

```

Please enter a file name.
t3.txt
Successfully opened file t3.txt

```

#####

Qualifying Laps:

#####

Fernando_Alonso completed the lap in 84030 milliseconds
Fernando_Alonso: current personal best is 84030; current position is 1
James_Hunt completed the lap in 88197 milliseconds
James_Hunt: current personal best is 88197; current position is 2
Kimi_Raikkonen completed the lap in 79282 milliseconds
Kimi_Raikkonen: current personal best is 79282; current position is 1
Ayrton_Senna completed the lap in 92750 milliseconds
Ayrton_Senna: current personal best is 92750; current position is 4
Kimi_Raikkonen completed the lap in 77201 milliseconds
Kimi_Raikkonen: current personal best is 77201; current position is 1
Kimi_Raikkonen completed the lap in 82502 milliseconds
Kimi_Raikkonen: current personal best is 77201; current position is 1
Sebastian_Vettel completed the lap in 81741 milliseconds
Sebastian_Vettel: current personal best is 81741; current position is 2
Sebastian_Vettel completed the lap in 73966 milliseconds
Sebastian_Vettel: current personal best is 73966; current position is 1
James_Hunt completed the lap in 80994 milliseconds
James_Hunt: current personal best is 80994; current position is 3
Ayrton_Senna completed the lap in 71047 milliseconds
Ayrton_Senna: current personal best is 71047; current position is 1
James_Hunt completed the lap in 80649 milliseconds
James_Hunt: current personal best is 80649; current position is 4
James_Hunt completed the lap in 71841 milliseconds
James_Hunt: current personal best is 71841; current position is 2
Ayrton_Senna completed the lap in 99303 milliseconds
Ayrton_Senna: current personal best is 71047; current position is 1
Fernando_Alonso completed the lap in 95106 milliseconds
Fernando_Alonso: current personal best is 84030; current position is 5
Kimi_Raikkonen completed the lap in 97888 milliseconds
Kimi_Raikkonen: current personal best is 77201; current position is 4
Sebastian_Vettel completed the lap in 90617 milliseconds
Sebastian_Vettel: current personal best is 73966; current position is 3
Ayrton_Senna completed the lap in 81740 milliseconds
Ayrton_Senna: current personal best is 71047; current position is 1
Ayrton_Senna completed the lap in 82619 milliseconds
Ayrton_Senna: current personal best is 71047; current position is 1
James_Hunt completed the lap in 75975 milliseconds
James_Hunt: current personal best is 71841; current position is 2
Sebastian_Vettel completed the lap in 99538 milliseconds
Sebastian_Vettel: current personal best is 73966; current position is 3
James_Hunt completed the lap in 70664 milliseconds
James_Hunt: current personal best is 70664; current position is 1
Kimi_Raikkonen completed the lap in 99347 milliseconds
Kimi_Raikkonen: current personal best is 77201; current position is 4
Sebastian_Vettel completed the lap in 97309 milliseconds
Sebastian_Vettel: current personal best is 73966; current position is 3
Fernando_Alonso completed the lap in 96873 milliseconds
Fernando_Alonso: current personal best is 84030; current position is 5
James_Hunt completed the lap in 86078 milliseconds
James_Hunt: current personal best is 70664; current position is 1
Kimi_Raikkonen completed the lap in 90040 milliseconds
Kimi_Raikkonen: current personal best is 77201; current position is 4
Fernando_Alonso completed the lap in 76980 milliseconds
Fernando_Alonso: current personal best is 76980; current position is 4

#####

Results:

#####

1. James_Hunt 70664
2. Ayrton_Senna 71047
3. Sebastian_Vettel 73966
4. Fernando_Alonso 76980
5. Kimi_Raikkonen 77201

Some Important Rules

In order to get a full credit, your programs must be efficient and well presented, presence of any redundant computation or bad indentation, or missing, irrelevant comments are going to decrease your grades. You also have to use understandable identifier names, informative introduction and prompts. Modularity is also important; you have to use functions wherever needed and appropriate.

Since you will use dynamic memory allocation in this homework, it is very crucial to properly manage the allocated area and return the deleted parts to the heap whenever appropriate. Inefficient use of memory may reduce your grade.

When we grade your homework we pay attention to these issues. Moreover, in order to observe the real performance of your codes, we may run your programs in *Release* mode and **we may test your programs with very large test cases**. Of course, your program should work in *Debug* mode as well.

You are allowed to use sample codes shared with the class by the instructor and TAs. However, you cannot start with an existing .cpp or .h file directly and update it; you have start with an empty file. Only the necessary parts of the shared code files can be used and these parts must be clearly marked in your homework by putting comments like the following. Even if you take a piece of code and update it slightly, you have to put a similar marking (by adding "and updated" to the comments below.

```
/* Begin: code taken from ptrfunc.cpp */
```

```
...
```

```
/* End: code taken from ptrfunc.cpp */
```

What and where to submit (PLEASE READ, IMPORTANT)

You should prepare (or at least test) your program using MS Visual Studio 2012 C++. We will use the standard C++ compiler and libraries of the abovementioned platform while testing your homework. It'd be a good idea to write your name and last name in the program (as a comment line of course).

Submissions guidelines are below. Some parts of the grading process are automatic. Students are expected to strictly follow these guidelines in order to have a smooth grading process. If you do not follow these guidelines, depending on the severity of the problem created during the grading process, 5 or more penalty points are to be deducted from the grade. Name your solution, project, cpp file that contains your main program using the following convention (the necessary file extensions such as .sln, .cpp, etc, are to be added to it):

“SUCourseUserName_YourLastname_YourName_HWnumber”

Your SUCourse user name is actually your SUNet user name which is used for checking sabanciuniv e-mails. Do NOT use any spaces, non-ASCII and Turkish characters in the file name. For example, if your SUCourse user name is cago, name is Çağlayan, and last name is Özbugsızkodyazaroğlu, then the file name must be:

Cago_Ozbugszikodyazaroglu_Caglayan_hw2

In some homework assignments, you may need to have more than one .cpp or .h files to submit. In this case add informative phrases after the hw number. However, do not add any other character or phrase to the file names.

Now let us explain which files will be included in the submitted package. Visual Studio 2012 will create two *debug* folders, one for the solution and the other one for the project. You should delete these two *debug* folders. Moreover, if you have run your program in release mode, Visual Studio may create *release* folders; you should delete these as well. Apart from these, Visual Studio 2012 creates a file extension of *.sdf* ; you will also delete this file. The remaining content of your solution folder is to be submitted after compression. Compress your solution and project folders using WINZIP or WINRAR programs. Please use "zip" compression. "rar" or another compression mechanism is NOT allowed. Our homework processing system works only with zip files. Therefore, make sure that the resulting compressed file has a zip extension. Check that your compressed file opens up correctly and it contains all of the solution, project and source code files that belong to the latest version of your homework. Especially double-check that the zip file contains your cpp and (if any) header files that you wrote for the homework.

Moreover, we strongly recommend you to check whether your zip file will open up and run correctly. To do so, unzip your zip file to another location. Then, open your solution by clicking the file that has a file extension of *.sln*. Clean, build and run the solution; if there is no problem, you could submit your zip file. Please note that the deleted files/folders may be regenerated after you build and run your program; this is normal, but do not include them in the submitted zip file.

You will receive no credits if your compressed zip file does not expand or it does not contain the correct files. The naming convention of the zip file is the same. The name of the zip file should be as follows:

SUCourseUserName_YourLastname_YourName_HWnumber.zip

For example, zubzipler_Zipleroglu_Zubeyir_hw2.zip is a valid name, but

Hw2_hoz_HasanOz.zip, HasanOzHoz.zip

are **NOT** valid names.

Submit via SUCourse ONLY! You will receive no credits if you submit by other means (e-mail, paper, etc.).

Successful submission is one of the requirements of the homework. If, for some reason, you cannot successfully submit your homework and we cannot grade it, your grade will be 0.

Good Luck!

Albert Levi, Taha Atahan Akyıldız