ASSIGNMENT 3

Subject : Linked Lists
TAs: Alaettin Uçan, Merve Ozdes, Pelin Canbay, Selma Dilek
Programming Lanuage: C++
Due Date: 16.12.2018 23:59:59

1   Introduction

Subject of this experiment is usage of linked lists. A linked list is a linear data structure, in which the elements are not stored at contiguous memory locations. The elements in a linked list are linked using pointers as shown in the Figure 1.
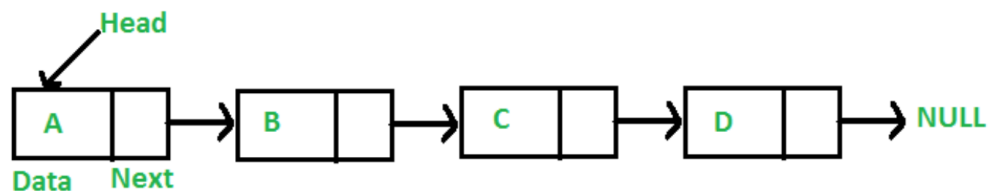

Figure 1: Representation of single linked list

In simple words, a linked list consists of nodes where each node contains a data field and a reference(link) to the next node in the list.

In a single linked list, every node has link to its next node in the sequence. So, we can traverse from one node to other node only in one direction and we can not traverse back. We can solve this kind of problem by using double linked list.

Double linked list is a sequence of elements in which every element has links to its previous element and next element in the sequence. In double linked list, every node has link to its previous node and next node. So, we can traverse forward by using next field and can traverse backward by using previous field. Every node in a double linked list contains three fields and they are shown in the Figure 2.
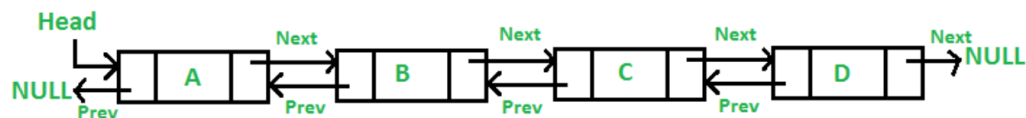

Figure 2: Representation of double linked list

2   Problem Definition

In this assignment, you will have input.txt which has footballer's information on each line about a football league. In each row, there are the name of the footballer, the team name of footballer who scored, the name of the away team, minute of the scored goal and the match ID. Each line is read and parsed one by one, you need to create your linked lists according to this information and add the data you have read from the file to the linked list by the adding method. The point to note is that the information can be mixed. The linked list structure you will build is designed as follows.
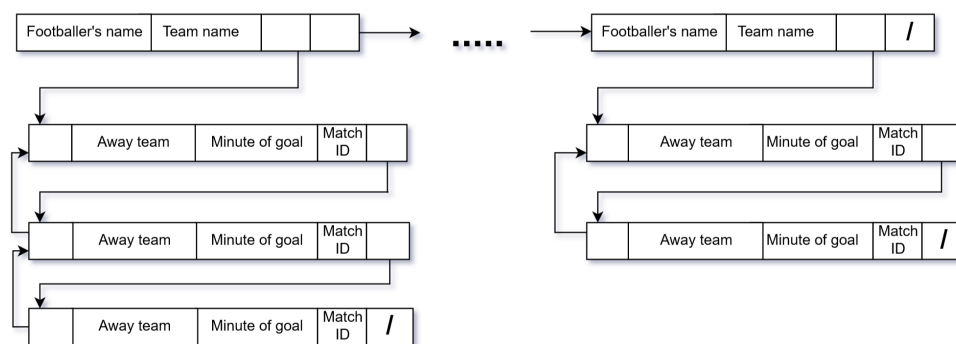

Figure 3: Linked List Structure

As shown in Figure 3, each node should contain footballer's name, team name, pointer which points to the another linked list and one pointer. The last pointer always points to the next member of the list. If the pointer is NULL, then it is the last node in the list. In the second linked list, there are away team name, minute of goal scored, match ID and two pointer. This pointer also shows the next node as it is in the other linked list.

3   Experiment

In this experiment, you are supposed to develop a simple search engine that uses linked list mentioned above. Firstly, you will read input.txt file and create linked list according to the ascending order of the names of the footballers. The information of football players in the file are not sequential. So you should make a ranking according to the name of the footballer while you are creating the linked list. The input.txt is shown and detailed in next section. Using these linked lists, write the C++ program that makes the following calculations.

1. The period in which the most goals are scored in the league (0-45 minutes in the first half, 46-90 minutes in the second half). If the first half, print 0. Otherwise, print 1.

2. Find the top goal scorer and print his name on the screen. There is no player who scored his own goal. (There may be more than one footballer with the same goal count, in this case all should be printed.)

3. To print the names of footballers who scored hat-trick. (Hat-trick is the scoring of three and more goals in a game by one player)

4. Print the team list in the league

5. Print the list of footballers

6. Matches and goals of given footballer

7. Sort by match ID in ascending order of given footballer

8. Sort by match ID in descending order of given footballer

You should use the linked list that you created for the above operations. In the final homework, all these operations should be listed on the output.txt.

4   Execution

The name of the compiled executable program should be "assignment3.cpp". Your program should read input/output file names from the command line, so it will be executed as follows:

```
[merve@rdev ~]$ g++ assignment3.cpp
[merve@rdev ~]$ ./a.out input.txt operations.txt output.txt
```

You will be given sample input and output files. You can get these files from piazza. Different input files will be used when evaluating the assignment. So it is recommended that you test your program on sample input and output files. The program must run on DEV(dev.cs.hacettepe.edu.tr) UNIX machines.

5   Input and Output

You will have two input files which are input.txt and operations.txt and one output files in this assignment. Input file is shown in Figure 4. As shown in Figure 4, each line has a footballer information. The data in each line is separated by commas. You should read and split the data by comma and add it to the linked list. Parameters for items 6, 7 and 8 are given in the

```
Burak Bulut,Denizli,Pamukkale,37,0
Lionel Maradona,Pamukkale,Denizli,55,0
Karim Benzema,Tavas,Municipality Sports,61,1
Robert Lewandowski,Municipality Sports,Tavas,67,1
Carlos Tevez,Municipality Sports,Tavas,90,1
Christiano Gomez,Denizli,Tavas,45,2
Diego Costa,Tavas,Denizli,75,2
Sergio Aguero,Pamukkale,Municipality Sports,01,3
Carlos Tevez,Municipality Sports,Pamukkale,12,3
Burak Bulut,Denizli,Municipality Sports,15,4
Robin Fernando,Denizli,Municipality Sports,21,4
Diego Costa,Tavas,Pamukkale,17,5
Diego Costa,Tavas,Pamukkale,25,5
Sergio Aguero,Pamukkale,Tavas,45,5
Diego Costa,Tavas,Pamukkale,60,5
Robin Fernando,Denizli,Pamukkale,45,6
Alexandre Lacazette,Municipality Sports,Tavas,90,7
Karim Benzema,Tavas,Denizli,76,8
Robert Lewandowski,Municipality Sports,Pamukkale,37,9
Luis Suarez,Pamukkale,Municipality Sports,45,9
Robert Lewandowski,Municipality Sports,Pamukkale,84,9
Robert Lewandowski,Municipality Sports,Pamukkale,90,9
```

Figure 4: input.txt

operations.txt file. In operations.txt, each line corresponds to one item. And each line has two parameters separated by commas. As shown in Figure 5, there are three line for sixth, seventh and eighth item. For example, for the sixth item you should print matches and goals of Diego Costa and Karim Benzema. You should do similar operations for seventh and eighth items. As output, you should print all the items mentioned above output.txt file.

```
Karim Benzema,Sergio Aguero
Diego Costa,Robert Lewandowski
Robin Fernando,Robert Lewandowski
```

Figure 5: operations.txt

Your program should write outputs to the output.txt file. Your output should be as shown in Figure 6. Your outputs will be evaluated automatically. So you should create your outputs by taking this format into consideration. As shown in 6, you must have results in a bottom line of each item definition. Results for items 6, 7 and 8 should be splitted by commas.

```
1)THE MOST SCORED HALF
0
2)GOAL SCORER
Diego Costa
Robert Lewandowski
3)THE NAMES OF FOOTBALLERS WHO SCORED HAT-TRICK
Diego Costa
Robert Lewandowski
4)LIST OF TEAMS
Municipality Sports
Denizli
Tavas
Pamukkale
5)LIST OF FOOTBALLERS
Alexandre Lacazette
Burak Bulut
Carlos Tevez
Christiano Gomez
Diego Costa
Karim Benzema
Lionel Maradona
Luis Suarez
Robert Lewandowski
Robin Fernando
Sergio Aguero
6)MATCHES OF GIVEN FOOTBALLER
Matches of Karim Benzema
Footballer Name: Karim Benzema,Away Team: Municipality Sports,Min of Goal: 61,Match ID: 1
Footballer Name: Karim Benzema,Away Team: Denizli,Min of Goal: 76,Match ID: 8
Matches of Sergio Aguero
Footballer Name: Sergio Aguero,Away Team: Municipality Sports,Min of Goal: 1,Match ID: 3
Footballer Name: Sergio Aguero,Away Team: Tavas,Min of Goal: 45,Match ID: 5
7)ASCENDING ORDER ACCORDING TO MATCH ID
footballer Name: Diego Costa,Match ID:2
footballer Name: Diego Costa,Match ID:5
footballer Name: Robert Lewandowski,Match ID: 1
footballer Name: Robert Lewandowski,Match ID: 9
8)DESCENDING ORDER ACCORDING TO MATCH ID
footballer Name: Robin Fernando,Match ID: 6
footballer Name: Robin Fernando,Match ID: 4
footballer Name: Robert Lewandowski,Match ID: 9
footballer Name: Robert Lewandowski,Match ID: 1
```

Figure 6: Example of output.txt

6 Report

You must write a report which is related to your program and only pdf format is accepted. you should describe in detail how you created the linked list in your report. (Report Format: ftp://ftp.cs.hacettepe.edu.tr/pub/dersler/genel/FormatForLabReports.doc You must write a report which is related to your program. The topics that should be included in the report are:

- Problem definition

- Methods and solution

- Functions implemented and not implemented

7 Grading Policy

Your work will be graded over a maximum of 100 points according to the grading policy given in the table below.

Task	Points
Report	10
Creation of linked list structure	30
Operations given in experiment section	60
Total	**100**

Notes

- Do not miss the deadline.

- Save all your work until the assignment is graded.

- The assignment must be original, individual work. Duplicate or very similar assignments are both going to be considered as cheating.

- The output of your program will be graded automatically. Therefore, any difference of the output (even a smallest difference) from the sample output will cause an error and you will get 0 from execution. Keep in mind that a program that does not work 100% right is a program that works wrong.

- Write READABLE SOURCE CODE block

- You can ask your questions via Piazza (https://piazza.com/hacettepe.edu.tr/fall/bbm203) and you are supposed to be aware of everything discussed in Piazza.

- You will use online submission system to submit your experiments. https://submit.cs.hacettepe.edu.tr/ Deadline is: 23:59 pm. No other submission method (email or etc.) will be accepted. Do not submit any file via e-mail related with this assignment.

- File hierarchy must be zipped before submitted (Not .rar, only .zip files are supported by the system). You must submit your work with the file hierarchy stated below:

  <StudentID.zip>

  →<src.zip>

  →assignment3.cpp <FILE>

  →<report.pdf>

Policy

All work on assignments must be done **individually** unless stated otherwise. You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an **abstract** way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudocode) **will not be tolerated**. In short, turning in someone else's work (from internet), in whole or in part, as your own will be considered **as a violation of academic integrity**. Please note that the former condition also holds for the material found on the web as everything on the web has been written by someone else.