

ITU Computer Engineering Department
BLG 223E Data Structures, Fall 2020
Homework #2

Problem Definition

You have an input.txt file in this homework. This text file has 2 columns which keep the group and value (ohm) of the resistors respectively in an electric circuit. The first column indicates the group of the resistor in the circuit. The second column indicates the value (ohm) of the resistor in the circuit. Each line indicates each resistor in the circuit. You design and implement an electric circuit using this file's resistor info on A SINGLE LINKED LIST.

Workflow

1. Read the input.txt. If the value of a resistor is positive then add this resistor to the circuit according the following rules:
 - Your circuit will be a linked list carrying 3 types of data fields: group, value and quantity. Add the resistors from the file to this linked list according to the resistor's group.
 - If the group of resistor is not in the list (series connection), you will create a node in a proper space and the quantity of this resistor should be set to 1.
 - If the group of resistor you are adding is already in the list (parallel connection), you will increase its quantity by 1.

Important Note: The value of the resistors connected in parallel (in other words, being in the same group) are all the **same**.

 - At each insertion, the linked list should be in sorted order (in alphabetically ascending order according to group name)
2. If the value of a resistor is negative (-) then remove this resistor from the circuit according to the following rules:
 - If the quantity of that resistor in that group is more than 1 (parallel connection), you will decrease its quantity by 1.
 - If there is only 1 resistor in the related group (series connection), delete this node from the linked list.
 - If there exist no such resistor (with that value or within that group), the program will print out the message "NO_RESISTOR".
3. 'A 0' is the command for the resistor info and calculation of total resistance in the circuit. Whenever, an input line from input file contains this command, print out the resistors info (value and quantity) in ascending order by resistor value, and calculate and print out the total resistance in the circuit. The format is "<value>:<quantity>\n".

4:1

5:2

6.5:1

8.5:2

Total resistance=24 ohm

Note: You can use two or more linked lists. **Arrays are forbidden in this homework. You can only use an array to get main function arguments in the code.**

4. At the end, delete all the nodes from your lists to free up the space.

Example 1

If the input file is as shown below, your circuit should be like in Figure 1.

Input file "input.txt"

```
A 3.5
B 6
C 4
B 6
D 9
E 2
F 3.5
D 9
D 9
```

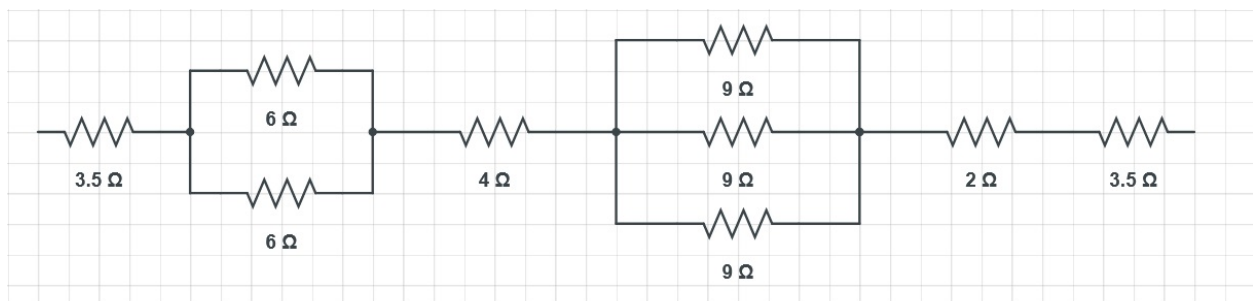


Figure 1: Circuit.

Example 2

If the input file is as shown below, your program output should be like:

Input file "input.txt"

```
A 3.5
B 6
C 4
B 6
D 9
E 2
F 3.5
D 9
D 9
A 0
C -4
B -6
A 0
C -4
B -6
D -9
G 7
A 0
```

Output

```

2:1
3.5:2
4:1
6:2
9:3
Total resistance=19 ohm
2:1
3.5:2
6:1
9:3
Total resistance=18 ohm
NO_RESISTOR
2:1
3.5:2
9:2
7:1
Total resistance=20.5 ohm

```

Implementation

Implement the following methods with appropriate arguments and return types for your structure:

- **create():** Creates the circuit(resistor list).
- **add_resistor():** Adds a resistor to the circuit (workflow 1).
- **remove_resistor():** Removes a resistor from the circuit. If this resistor is needed to delete from the circuit, you should call the delete_resistor function.
- **delete_resistor():** It deletes the resistor node from the circuit.
- **circuit_info():** Prints the resistor info and total resistance in the circuit (workflow 3).
- **clear():** Deletes all of the nodes of the list (workflow 4).

Structure

```

struct resistor{
char group;
double value;
int quantity;
resistor *next;
};

struct circuit{
resistor *head;
void create();
void add_resistor(char, int);
int remove_resistor(char, int);
void delete_resistor(char);
void circuit_info();
void clear();
};

```

Although the parameters and return types of the functions are specified above, you can use different parameters and return types in these functions. You may also add extra functions when necessary.

Submission Rules

- Make sure you write your name and number in all of the files of your project, in the following format:

```
/* @Author  
Student Name: <student_name>  
Student ID : <student_id>  
Date: <date> */
```

- Use comments wherever necessary in your code to explain what you did.
- You are not allowed to include any Standard Template Library (STL) container.
- To develop your implementation, you can use any IDE (Integrated Development Environment) such as Visual Studio etc. But before submitting, your program should be compiled and ran on Linux environment using **g++ (version 4.8.5 or later)**. You can test your program on ITU's Linux Server using SSH protocol. To compile the code, use the following command:

```
g++ -std=c++11 -Wall -Werror main.cpp -o main
```

And you can execute your program by using the following command:

```
./main input.txt
```

Your program will be checked by using **Calico**(<https://bitbucket.org/uyar/calico>) automatic checker.

- **Arrays are forbidden in this homework. You can only use an array to get main function arguments in the code.**

If explanations for this homework is not clear, you can ask your question under the thread that is specially started for homework 2 (About HW2) on the message board for BLG 223E on NINOVA. Please check before writing your question whether your question is asked by someone else.

Do not share any code or text that can be submitted as a part of an assignment (discussing ideas is okay).

- Only electronic submissions through Ninova will be accepted no later than deadline.
- You may discuss the problems at an abstract level with your classmates, but you should not **share or copy code** from your classmates or from the Internet. You should submit your **own, individual** homework.
- Academic dishonesty, including cheating, plagiarism, and direct copying, is unacceptable.
- A different input file will be used for evaluating the homework.
- Note that **YOUR CODES WILL BE CHECKED WITH THE PLAGIARISM TOOLS!**



This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.