

75 pts

Name 1: _____

Name 2: _____

Class Day / Time: _____

Due Date: _____

Lab #6 – Structs

Write a program that will first receive from the user as input the name of an input file and an output file. It will then read in a list of names, id #s, and balances **from the input file specified** (call it **InFile.txt**) and initialized the array of structs. The input file will be created from the data provided below. The program will then execute some tasks using the array of structs:

- 1) Find the person with the higher balance and output the person information (name, id # and balance) in the **output file**.
- 2) Obtain the sum of all balances and output the result in the **output file**.
- 3) The program will then prompt the user for a name to search for, when it finds the name it will **output to a file** (call it **OFile.txt**) the person's id#, name, and balance. Please see the next page for the expected format of your output file. The program should prompt for another name until the word **done** is entered. Calculate and output the **sum** of all balances for the found names.

Finally, create a header file for all your preprocessor directives, prototypes, constants and your struct (structs should be defined in the same location as enums and typedefs). **NOTE: For this assignment do not make the array size a GLOBAL constant → pass it as a parameter as necessary.**

NOTICE: This is very similar to your parallel arrays assignment except with structs.

Your program will need the following functions:

- 1 – **print heading function** – prints your heading to console (**cout**) and **OFile.txt**.
- 2 – **input function** - that will propagate the array of structs (reads in all the data) – you will need **one ARRAY of a struct** for this assignment (you will need to declare a **struct** with three members (**names, id, and balance**). This function should read all the data from **InFile.txt**.
- 3 – **search function balance** - this function will search for the largest **balance** and return the proper index (of type int) to the calling function.
- 4 – **sum function** – this function will add all balances and return a type float with the result to the calling function.
- 5 – **search function name** - this function will search for a **name** and return the proper index (of type int) to the calling function.

Turn in (IN THIS ORDER)

1. The **first page** of this lab
2. screen I/O – pasted into a text file
3. output file
4. Your header file
5. a listing of main.cpp (conforming to style discussed in class)
6. a listing of your input/output, add and search functions
7. a listing of your print heading function (should be in a separate file)
8. input file

TEST INPUT – Test each one of these cases!

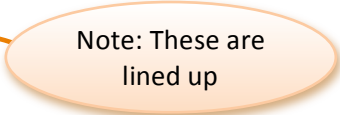
Search for the following input in this order:

Steve Woolston
Jacques Rousseau
Chris Carroll
Lisa Covi
Florence Rousseau
Frankie Lane
done

Screen Input/Output should be formatted as follows:

Note: I have provided the input/output for the **first two test cases and the last test case**. ALL test cases should be formatted as follows.

What input file would you like to use? InFile.txt
What output file would you like to use? OFile.txt



Note: These are lined up

Who do you want to search for (enter done to exit): Steve Woolston
Found.

Who do you want to search for (enter done to exit): Jacques Rousseau
Jacques Rousseau was not found.

... Also search for the following names here. Continue to format the I/O as described above.

Chris Carroll
Lisa Covi
Florence Rousseau
Frankie Lane

... The following should be your last input.

Who do you want to search for (enter done to exit): done


Thank you for using my program.

Input File: (your input file should look exactly as follows)

Jean Rousseau
1001 15.50
Steve Woolston

```
1002 1423.20
Michele Rousseau
1005 52.75
Pete McBride
1007 500.32
Florence Rousseau
1010 1323.38
Lisa Covi
1009 332.35
Don McBride
1003 12.32
Chris Carroll
1008 32.35
Yolanda Agredano
1004 356.00
Sally Sleeper
1006 32.36
```

Output File: should include **all of the names** that were found (not just these) in the order in which they were found. The output should be formatted as follows.



```
Higher Balance:
ID #      NAME      BALANCE DUE
----      -
1002      Steve Woolston    $    1423.20

Total Balance for all persons:
$    4080.48

Search Names:
ID #      NAME      BALANCE DUE
----      -
1002      Steve Woolston    $    1423.20
1008      Chris Carroll     $     32.35
...

Total Balance Due: $    3111.28
```