Programming Assignment #5 (due 23:59:59 01/09, 2015)

Course Website: http://54.68.45.250/~ec2-user/index.html

Problem: Bank Data Manager

In this assignment, you need to write a C++ programming that manages the account information of the three banks. At these banks, users would have an ID and money they saves in one of the banks someday. To simplify the problem, the program would just record the user ID and the amount of money into a list, executing SORTMONEY() for three banks' list respectively, and MERGEID() three lists.

Functions

SORTMONEY() takes a bank's name as input. The list of that bank should be sorted by the amount of **money** in non-decreasing order, and return the sorted result.

MERGEID() should merge the three banks' lists, and return merged list in non-decreasing order by sorting user **ID**.

Provided files: (1) main.cpp, (2) bankdatamanager.cpp, (3) bankdatamanager.h, (4) user.h, (5) bank1_example, (6) bank2_example, (7) bank3_example, (8)example_answer

- **main.cpp** – It executes the two functions, and checks the answer when executing **SORTMONEY** or **MERGEID** functions. It can be changed if necessary for you to debug.
- **bankdatamanager.cpp & bankdatamanager.h** – These are program files you need to implement.
- **user.h** – It defines the class USER, which you need not modify.
- **bank1_example, bank2_example, bank3_example** – users' information files, recording user ID and money. User ID is a positive integer. Note that there are **six digits** after the decimal point for the money, and the amount of money is larger than zero. ID and money would not duplicate in all of the lists.

INPUT

An example sequence is shown below:

```
694009 1577956.950570
904897 2126075.367416
182493 2054227.107933
929436 1144497.775421
652049 266287.995974
83324 922936.947933
155049 269546.589312
```

Platform

You may develop your software on UNIX/Linux.

Compile: $g++ main.cpp bankdatamanager.cpp

Execution: $./a.out

Submission

Please update the following materials to online judge system on the course website by the deadline, specifying your account and check the leader board.

(1) bankdatamanager.h

(2) bankdatamanager.cpp

Grading

(1) example correct: 40%

(2) 100% accuracy: 20%

(3) leader board ranking: 20% (ranking priority: accuracy > runtime)

(4) hidden cases ranking: 20% (ranking priority: accuracy > runtime)