Assignment 3
CSL 201 Data Structures, Sem 1, AY 17/18, IITRPR

40 marks | Due on 31st October, 11:59 PM

Objective

The objective is to design and implement two anstract data types: (1) Hash Tables, (2) AVL Trees.

Instructions (IMPORTANT)

1. You are to use C++ programming language to complete the assignment.
2. Provide a Makefile to compile your final code.
3. This is an individual assignment. You can have high level discussions with other students, though.
4. Include a “Readme.txt” file with required explanation and instructions to compile and run the code.
5. Upload your submission to moodle by the due date and time. There will be 10% penalty for late submission for first three days. After that, your assignment will not be evaluated.
6. If any student asks for deadline extension, he or she will get 5% penalty.
7. I should be able to use the class you have implemented in my program the way we use STL classes. There will be a master program to test the classes. If the class is templated, declare it in the hpp file itself.

[20 marks] Problem 1

Design two hash tables, one to store indian names (key=name, value=meaning) and other to store indian phone numbers (key=phone number, value=balance). Justify your design choices in terms of: (1) all four hash codes (Integer Casting, Component sum, Polynomial sum, Cyclic sum) (2) all three hash functions (Division, MAD, Multiplication). For each of the cases above, count the number of collisions (percent). Take the table size to be 743 buckets in the first case (phone numbers) and 2531 in the second case (names).

The class interfaces for both the hash tables are given in file number_map.hpp and name_map.hpp.

Input Data

Input will be given as raw text files:
names.txt - each line contains one entry, first word is the name and the remaining words describe the meaning.
numbers.txt. - each line contains an entry in the following format “number balance”.

Output Data

The output should be two 4X3 matrices. You should display as well as store the results in a file named EntryNumOutput.txt. Also write your observations in less than 200 words.

User Interface

The program should also implement a command-line based interface which takes a key as input and returns its value. See the example below:
$Please enter 1 to know balance of a phone number and 2 to know meaning of a name 1
$Please enter number 9849489137
$Number = 9849489137, Balance = INR 254
$Please enter 1 to know balance of a phone number and 2 to know meaning of a name 2
$Please enter name Tulasi
$Name = Tulsi, Meaning = a sacred plant (basil)

[20 marks] Problem 2

In this assignment you need to design and implement a modified AVL tree class. The modified AVL tree stores entries with integer keys and string values (e.g. roll no and name). The modified AVL Tree class must be defined in mavl.hpp. In addition to the standard AVL Tree functions, the MAVL class must support the additional following functions:

- delete_by_place(int i): Delete from the entry at the ith place (as determined by the order of insertion) among the elements that are in the tree at that moment.
- get_place(int k): Return the place (which is determined by the order of insertion) of x among the elements that are in the ADT at that moment. If x does not exist, return -1

For example, after the following sequence of actions:
insert(3, A), insert(5, B), insert(11, C), insert(4, D), insert(7, E), delete(5)

get_place(7) returns 4, and delete_by_place(2) will delete 11 from the tree.

The MAVL class interface is given in file mavl.hpp. All functions should be implemented in O(1) or O(log n) time. [Hint: You may need to build two balanced trees in your data structure.]

Input Data

The program should read the entries from entries.txt (number, string) and build initial tree.

User Interface

The program should allow the user to insert a key, value pair, delete a key, get_place, and, delete_by_place.
$Please enter 1 to insert, 2 to delete by key, 3 to get the place of a key, and 4 to delete by place of key
…