



BIRZEIT UNIVERSITY  
COMP232/COMP2321/COMP242 (Spring 2016/2017)

### 3<sup>rd</sup> Project

## AVL Tree & Hashing

### Building a Data Structure for Country's Tourist Places

- Given a file that contains information about the top 100 tourist cities in the world. The file is formatted as follows: Rank\*City\*Country\*Number of Tourists (Millions)
  - Using the given data file, build an AVL tree of Country nodes (use Country name as key).
  - Please note that in each country may be more than one tourist city (e.g., Guangzhou, Shanghai, Beijing, Hangzhou, Zhuhai, Suzhou, Guilin, Nanjing ARE all in China), so the country may be repeated more than once in the file. If the country found in the AVL tree (already inserted before), add the new tourist city to this country (you may create a list of tourist cities in each country node).
  - Calculate the total number of tourists in each country (if the country found in the tree: insert a new node to the city list, and add the number of tourists in to the total number of tourists in that country). Here is an example: Guangzhou, is the first city in china, the total number is 7.9, when Shanghai is read from the file, it will be added to city list in china NODE and the total becomes  $7.9+6.5 = 14.4$ . When all cities added to china node, the total must be 30 million.

14	Guangzhou	China	7.9
20	Shanghai	China	6.5
25	Beijing	China	5
41	Hangzhou	China	3.4
54	Zhuhai	China	2.7
63	Suzhou	China	2.5
80	Guilin	China	2
			30

- Implement the following functions on countries AVL tree:
  - Print out countries sorted.
  - Search for a specific Country.
  - Insert a new Country.
  - Delete a specific Country.
  - Calculate tree height.
- Read the countries information from the AVL tree as: country name, number of tourist cities, total number of Tourists.
- Create a Hash Table using the countries data you got from the previous step (use the city name as a key).
- Implement the following functions on the hash table:
  - Print hashed table (including empty spots).

- Print out table size.
- Print out used hash function.
- Insert a new record to hash table.
- Search for a specific record.
- Delete a specific record.
- Save hash table back to file.

**Good Luck!**