CMP-210: Data Structures and Algorithms **Instructor:** Waheed Iqbal

---

Assignment 02: Search Engine and Web Server Log Processing

Submission Deadline: Sunday 11:00pm, 28th December, 2015

Problem 01: Search Engine

Suppose you have developed a small search engine that can help users to search over the Internet. Your search engine provides set of URLs in response to specific search query from the users. We have provided you two files:

a. **`search-results.txt`:** Each line in this file provides search query and set of URLs containing the result for the search query
b. **`search-history.txt`**: Each line in search-history contains the search queries issued by the users.

You need to build a Binary search tree based on **`search-results.txt`** by using the following guidelines:

1. You need to identify top 10,000 search queries based on maximum occurrence from the **`search-history.txt`**
2. Use search query as a key in the binary search tree.
3. The comparison between keys should follow lexicographic (dictionary) order.
4. Each node contains a set of URLs against the key.
5. For top 10,000 nodes you should have the set of URLs in memory but for the rest of the nodes, each node points to a file containing set of URLs against the search query.

Once you build the tree based on the above guidelines, your program should read an input from the user as a search query and print appropriate set of URLs by querying the binary search tree. The output should also mention that URLs are read from the memory or from the file. For example:

```
Please enter search query: Aaru
Output: URL-2661, URL-21975, URL-519, URL-17318, URL-20031, URL-8652, URL-22149, URL-0
URLs Read From: memory
```

Problem 02: Web Server Log Processing

Whenever a user browses a website, the web server hosting the website maintains a log file to record the user activity. A web server log file contains various important fields e.g., IP address of user, date time, request method, protocol, page name, response code, and response size in bytes against each request. In this assignment your objective is to read a web server log file line by line and process it to identify total number of requests and average response size against each web page. The input log file would be in the following format:

```
10.223.157.186 - - [15/Jul/2009:14:58:59 -0700] "GET / HTTP/1.1" 403 202
10.211.47.159 - - [06/Jan/2010:23:35:03 -0800] "GET /crm/campaigns_contacts.php?c=22&_search=false& 50&page=1 HTTP/1.1" 200 231
10.211.47.159 - - [06/Jan/2010:23:35:03 -0800] "GET /crm/ campaigns_contacts?c=22nd=1262849703513&rows=50 HTTP/1.1" 200 4162
10.121.241.75 - - [06/Jan/2010:23:28:59 -0800] "GET /robots.txt HTTP/1.0" 404 208
10.121.241.75 - - [06/Jan/2010:23:28:59 -0800] "GET / HTTP/1.0" 200 12758
………
………
```

If you split each line with space you will get page name on index number 6 and response size on index number 9. You need to write a program that should output the following for this input file:

```
#Page Name    Total Requests    Average Response Size
/                         2                6480
/crm/campaigns_contacts.php            2                2196.5
/robots.txt    1                208
```

Each line in the output represents page name, total requests, and average response size separated by tab (\t).
The input file is provided to you next to the assignment at course homepage.

Important!

1. If you see a question mark (?) in page name, you need to truncate the rest of the page name. For example you can see the output for page name /crm/campaigns_contacts.php related to the sample input.
2. You may only use Map data structure from STL. You need to use your own container to store the data.
3. You need to read the input file name from command line. You may need to see a quick tutorial at http://www.cplusplus.com/articles/DEN36Up4/
4. Your program output should not contain any irrelevant output.
5. The input access log file may contain few lines that do not provide you appropriate number of tokens. Therefore your program must ignore those lines.