Instructor: Waheed Iqbal
Data Structures and Algorithms

Assignment 02: Search Engine Using Binary Search Tree

Search Engine

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

a. **`search-results.txt:`** each line provides a search query and a set of URLs containing the search query
b. **`search-history.txt`**: Each line in search-history.txt 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 the top 10000 search queries based on probability 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 the top 10000 nodes, you should have the set of URLs in memory but for the rest of the nodes, each node points to a file containing a set of URLs against the search query.

Once you build the tree based on the above guidelines, your program should read input from the user as a search query and print an 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 the search query: Aaru
Output:  URL-2661,  URL-21975,  URL-519,  URL-17318,  URL-20031,
URL-8652, URL-22149, URL-0
URLs Read From: memory
```

Submission;
1. Prepare a small report listing the top 50 queries with their search probability score.
2. Profile the time for at least 10 queries served through memory and served from the file.
3. Submit your source code.