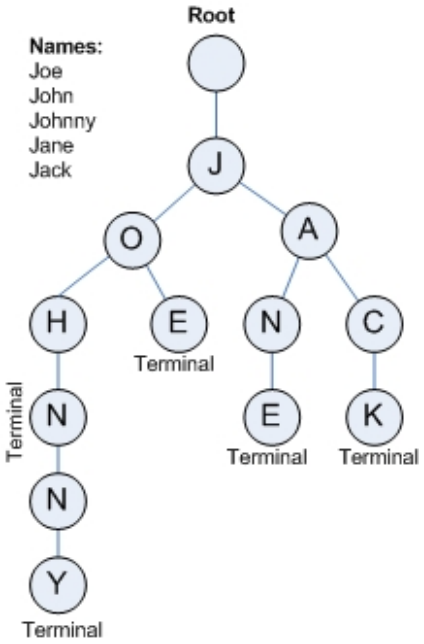
Saman Arora
10434371

CS600 - Project ReadMe

This Search Engine going to take **1 Base URL** and read all the other hyperlinks present in the webpage. Once it has full list of links (including the Hyperlinks), The Search Engine scrapes the paragraphs from all the webpages and extracts words and stores these words into a Trie data-structure. Now, to find the word entered by the user, trie data structure would be used to search as follow:



The Search Engine uses a Treemap is used to find the count of occurrences of a word from each webpage. The Treemap would display their word count respectively and if it is not present in one of the webpages the count is shown as zero. As specified the Search Engine would remove all the stop words, such as articles, prepositions, and pronouns, while searching the Keywords in the webpages. This action is performed by the class StopWord.java will takes an input file named “stopwords.txt, if the word is not present in the input file it inserts the word into the Trie data-structure.

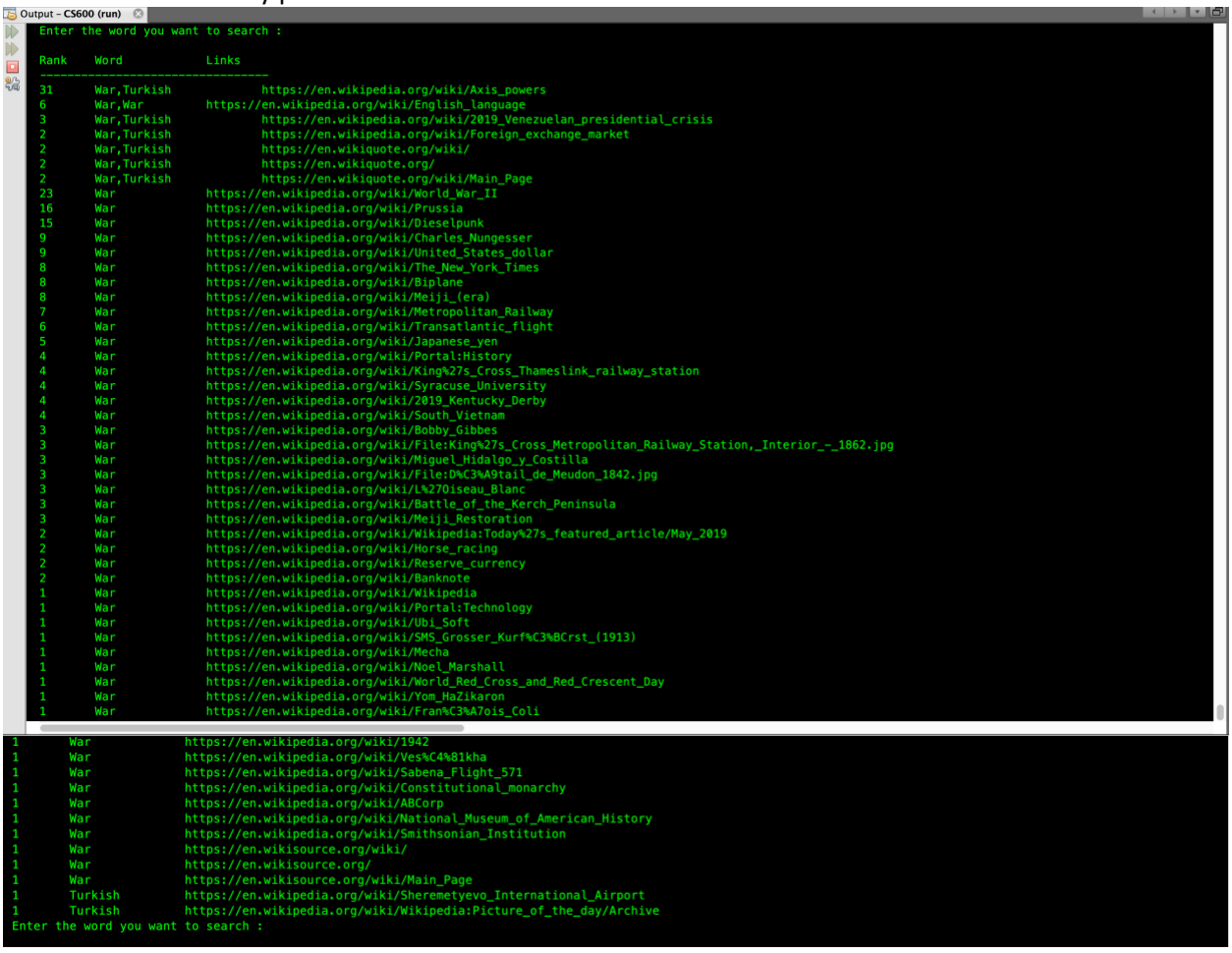
Once, the webpages are been scraped and the desired data for the output is ready for printing, it will then Rank the webpages according to the occurrence of the words in document. This uses Page Ranking Algorithm to rank the pages. The webpages are compared based on number of occurrences of the word on a page. For ranking, webpages are sorted using merge sort of it to be efficient and fast. Also, this considers the total number of all the words from all the documents and maintains a separate map which displays the words as well as their total occurrences found from all the documents.

```
pages.sort(new PageComparatorMulti());
        System.out.println("Rank\t" + "Word\t\t" + "Links\n" + "---------------------------------");
        for (PageRankingAlgo page : pages) {
           System.out.println(page.getRank() + "\t" + page.getWords() + "\t\t" + page.getName());
        }
```

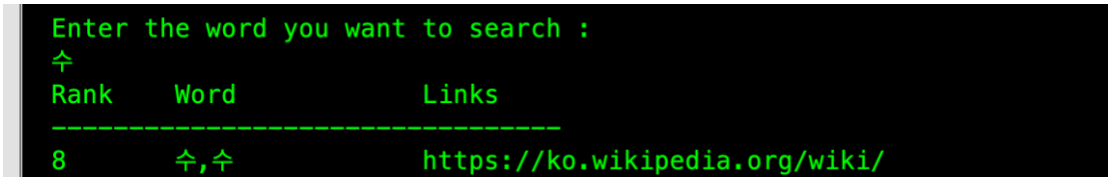
Note:- Pages is a list of webpages and it calls in-built sort algorithm which uses a comparator to compare and the algorithm uses Merge sort for sorting.

Sample Output:

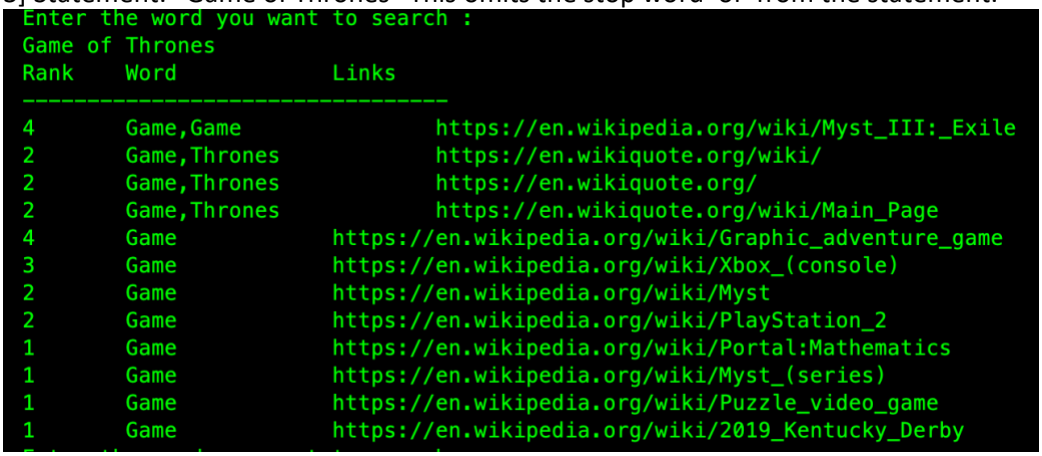
1] Here I have searched “War Turkish”, We get the pages where war and Turkish is present also war and Turkish are individually present



2] Word ” 수” which means Number in Korean, is present on Single link only

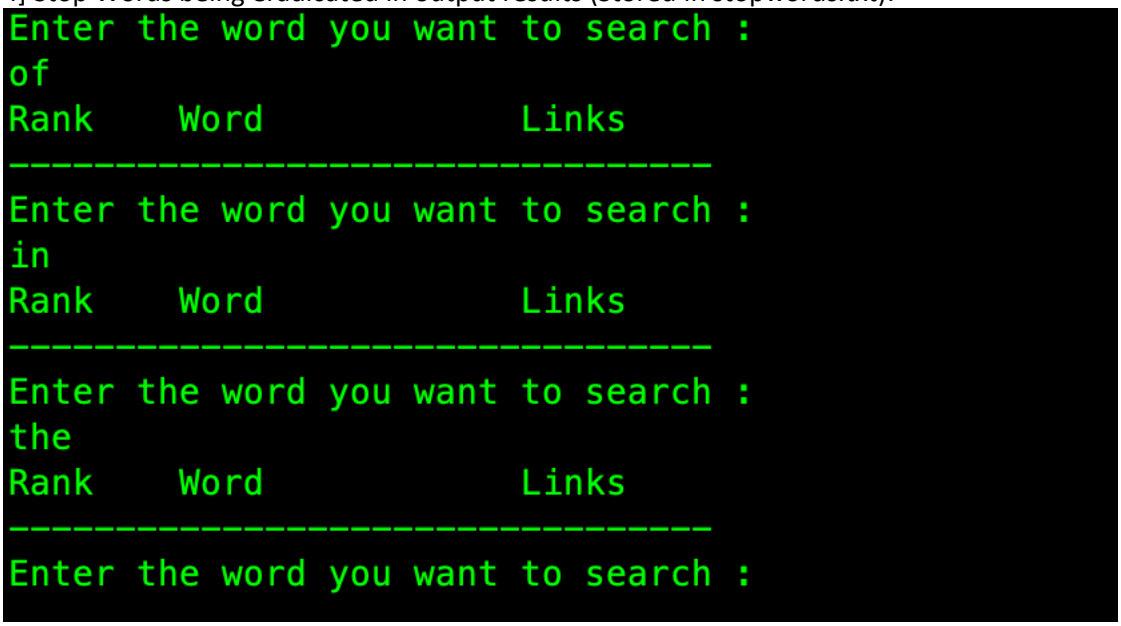


3] Statement: “Game of Thrones” This omits the stop word ‘of’ from the statement.



ss

4] Stop Words being eradicated in output results (Stored in stopwords.txt):



Note:- As shown above, this search engine works for single words as well as a String of statements.