Used Algorithms
Team 17

Indexer :
Connect to Crawler database
Get the collection of crawled pages
Loop on each document in the collection
Get the url of the document, parse it and start indexing
Wr stem each word, remove punctuation and convert it to lowercase letters.
We used HashMap for indexing, then after indexing all documents we connect to the Indexer database and move all elements in the hashmap to the database
The Indexer database is MongoDB
Indexer database consists of collections which represents words
Each collection contains some documents
Each document represents a web page that contain this page
Each document contains the URL, TF, weight of the word it that URL which is a number representing the highest tag of the word in this page، Array of tags representing the tags that contains this word in this page (to be used to determine the weight after indexing the whole page by checking the most important tag in this list)
and list of positions which represents the positions of this word in the page to be used in the UI

We used HashMap because its complexity is O(1).
=========
Query processing :
We check if the first and last letters are “ so it is phrase else it is Non phrase
In phrase Searching we only retrieve pages the contains the complete phrase (case insensitive )
We get the numbers of occurrences of the phrase in each URL and take it into consideration in ranking.

In Non phrase search we retrieve all pages containing one or more word of the query either in its phase or after stemming
====
Ranker:
In phrase Searching we rank only pages that contain the complete phrase respectively.
We rank based on the frequency of the phrase in the page, the weight of the first word of the query ( as all words comes after each other so we don't need to get the weight of each word into consideration)

In Non phrase Searching we first rank pages that contain one or more words of the query without being stemmed. Then we rank pages that contain one or more word of the page in the stemmed form and don't contain any word of the query in the original case.
We rank based on TF_IDF, sum of weights of words existing in this page from the query and Number of query words inside this page

=========
Crawler:

1-we start from initial links seed , insert into queue
2- we extract a link from queue , then
-check accessibility in robots.txt
-check that it has a unique compact string (combination of h1-h6 , ul,li, a , p  elements )
3- establishing a connection , then parsing html file for url tag <a href> —> parsing then adding all these links into queue

Then storing current url in mongodb crawler database ,as long as following info:
-page title
-url
-compact string
-popularity

,then incrementing no of crawled pages
>And so on, till we reach max count (5000)

If interrupted , we retrieve previous run pages then continue where we left off , we know by first retrieving number of pages then if it's <Max_link_count  we continue crawling

=======