This project report is for Analysis and Design of Algorithms Lab for Fall 2020, AUC.
SID: 900183162
Name: Youssef M. Hussien
The project repo is available at: JoHussien/SearchEngine

Deliverable One: Source code (.cpp files)

The cpp files are attached to the submission and available also at the repo.

Deliverable Two: Executable file

Also attached to the submission

Third DEliverable: The pseudo-code for your indexing and ranking algorithms

Indexing Algorithm:

First we go over each source website in the webgraph.csv file,
Then the first encountered vertex is at index 0 and henceforth,
This indexing sticks to the vertex, hence when the score causes the pages to appear in different order their indexing is the same,

Ranking Algorithm:

First we create a vector of initial ranking values, ech value is equal to (1 / number of vertices)
Then we do one iteration to update the rank of each vertex,
We run over each vertex and then run on each child of the certex's children
At each child we divide the previous rank (initial rank of that child on the number of children of this child)
Number of children is already stored in a vector which was computed by running over the whole graph.
Hence the ranking algorithm runs for one iteration and has a time complexity of O(n*m) where n is the number of vertices in the graph and m is the maximum number of children for a given vertex.

This project report is for Analysis and Design of Algorithms Lab for Fall 2020, AUC.
SID: 900183162
Name: Youssef M. Hussien
The project repo is available at: JoHussien/SearchEngine

Fourth Deliverable: A time and space complexity analysis for your indexing and ranking algorithms

Indexing algorithm is of O(E) time complexity as it runs over all edges in the WebGraph.csv file.

The space complexity is O(n) where n is the number of vertices as we create a vector to store the indices.

Ranking Algorithm has a time complexity of O(n*m) where n is the number of vertices in the graph and m is the maximum number of children for a given vertex.

Space complexity analysis of the ranking algorithm is O(n) as we create a number of extra vectors to store the rank of each vertex.

Fifth Deliverable: The main Data structures used

I used mainly unordered_map, linked lists and this was instead of using a Graph.

Moreover, I used vectors mainly instead of dynamic arrays.

Sixth Deliverable: Trade Offs,

There were no trade offs to my knowledge