Mini Project #2

(PageRank)

1. Objective

a) To understand the PageRank algorithm

b) To implement the algorithm using adjacency matrix and inverse adjacency lists

2. Problem

Two reference source files, i.e., pagerank_matrix.c and pagerank_list.c, are implemented partially. In this assignment, you are going to implement completely the given source files.

1) pagerank_matrix.c

This file calculates the PageRank scores using the adjacency matrix. In this file, two functions, generateTPMatrix() and calculatePageRank(), are not implemented. The generateTPMatrix() is for generating the transition probability matrix from the adjacency matrix. In this step, you need to handle carefully pages with no outlinks. In the calculatePageRank() function, you need to implement the 'power iteration' given the vector-matrix version of the PageRank equation. The remaining functionalities, which include allocating memory, loading the adjacency matrix, evaluating the processing time, writing the result file, and printing the top-k scored pages, are provided for you.

2) pagerank_list.c

This file calculates the PageRank scores using the inverse adjacency lists. In this file only one function, calculatePageRank(), is not implemented. The function is for calculating the PageRank scores using the inverse adjacency matrix. In this function, you need to implement the 'power iteration' given the original PageRank equation. The remaining functionalities, which include defining the data structures for the inverse adjacency lists, allocating memory, loading the inverse adjacency lists, evaluating the processing time, writing the result file, and printing the top-k scored pages, are provided.

The two files share the same command-line arguments, which enable you to test the programs with various input files and parameters without modifying the original source codes. For more details, please refer to the comments in the files.

3. Dataset

We provide two classes of dataset, i.e., Wikipedia and movie. Each class of dataset includes the adjacency matrix and the inverse adjacency lists. For the movie dataset, we additionally provide 'node' and 'adjacency list' files for more information on the dataset.

1) Dataset/wikipedia/

The Wikipedia dataset, which consists of 11 web pages, is identical to the example graph that you can find in the presentation file, '[20150604]PageRank MiniProject#2.pptx'. Please use this dataset to verify your implementations of the PageRank algorithm.

2) Dataset/movie/

The movie dataset consists of 5,757 web pages. You need to report the two result files, each of which is obtained by the writeVector() function, given this dataset. Also, please report the processing time for each source file.

4. Requirements

a. Language: ANSI C

b. Please do not modify the main() function (if necessary, modification of codes related to the command-line arguments is allowed).

5. Report

The report should include one report document, two result files, and source files. Please create and submit a zip-file that contains all the files.

1) Document

The report document should include:

- Three functions that you implement for this assignment
- Brief description of how you implement the functions
- Screenshots of results for each pagerank_matrix.c and pagerank_list.c file

2) Result files

Please name the result files with the following format, e.g., 2012010704_wikipedia_matrix.txt.

- [studentNumber]dataset_matrix.txt: PageRank scores obtained by the pagerank_matrix.c
- [studentNumber]dataset_list.txt: PageRank scores obtained by the pagerank_list.c

3) Source files

Please attach the solution directory containing the pagerank_matrix.c and pagerank_list.c