Introduction to Data Science / Data Intensive Computing (CIS 4930/6930)

Project III

Instructor: Dr. Sanjay Ranka

TA: Yupeng Yan *yupeng@cise.ufl.edu*

March 28, 2014

Department of Computer and Information Science and Engineering

University of Florida

1. Project Description

(1) Task 1

Write a parallel PageRank program in OpenMP (please see **Resources** Part to find the requirement of input Web graph file). Assume that only one of the threads reads the file. The pagerank values are initialized to a normalized identity vector by all the threads, and then updated using a matrix vector product. The process continues until the page ranks do not change significantly (you should explicitly give the condition or threshold in your report).

(2) Task 2

Write a parallel reducer (from MapReduce) using MPI. Each processor has a table of key-value pairs. Each key and value is an integer. Also, the size of the table in each processor is equal. The output should be a partitioned table. In this table, each key in the input appears only once, and the associated value of this key is the sum of all values associated with it in the input. You can use a hash or a sort for your local reduce. At the end of this step each key only appears once. For the second step, assume that each key is mapped to a processor (*e.g.* using cyclic or block distribution). This step will require a bulk communication to ensure that the number of messages from each processor to every other processor is at most one. In the final step, each processor does a second local reduction using all the messages received. Test your algorithm for 100,000 on key value pairs. An input file (please see **Resources** Part) will be provided for testing.

2. Project Submission

This is an *individual* project. Your project submission should include source codes and a project report. In the project report you need to describe the implementation and performance results of the programs in both tasks. Your project submission layout should be as follows:

Project3_LastName_FirstName.tar (or .zip)

Report.pdf

Task 1

src/all the source code

Task 2

src/all the source code

You should tar your source code files and the report in a single tar file, in which the codes for different tasks are put in different folders.

3. Grading Guidelines

You should be responsible to make sure that you follow all the requirements in **Project Description** Part. Your project will be graded to 0 if either of your program cannot read the provided input files.

Components	Task 1	Task 2	Report
Grade (%)	40	40	20

4. Late Submission Policy

Every late submission will be penalized 20% for each day late for up to a maximum of 3 days from the due date (**Monday, April 11th**).

5. Resources

(1) The Web graph to be used in **Task 1** is *ego-Facebook* (or *ego-Twitter*). You can download it from Stanford Network Repository (http://snap.stanford.edu/data/index.html#web). It is in *Social networks* table.

(2) You can find the input file (*100000_key-value_pairs.csv*) for **Task 2** on Resources page. http://www.cise.ufl.edu/class/cis6930sp14ids/resources.