Single Source Shortest Path using Hadoop

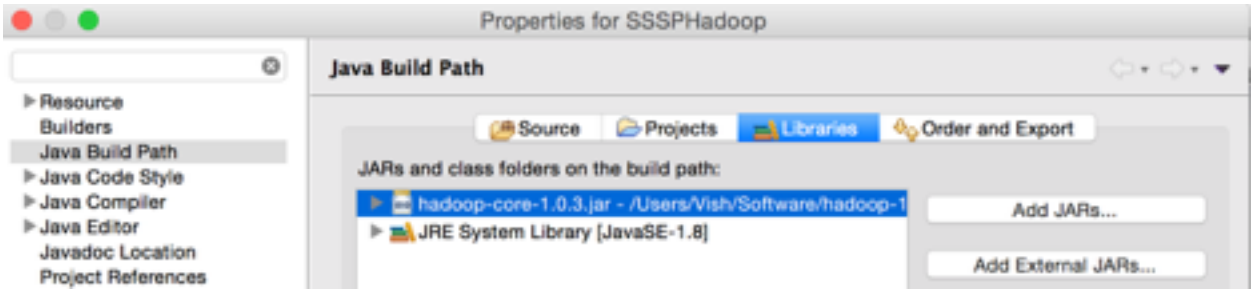
by Vishal Doshi
vdoshi3@uic.edu

Spring 2015

NOTE: The programs have inline comments for easy understanding. I have also included the sssphadoop.jar file for easy

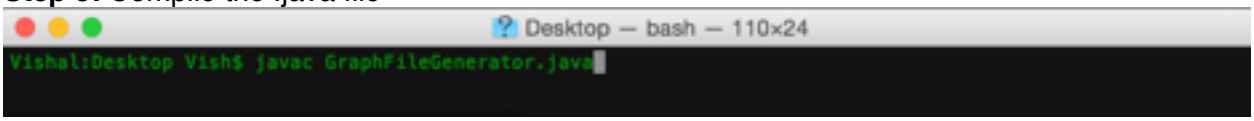
Step 1: Import the project in Eclipse. (659391383_ASSIGN1.zip has the project)

Step 2: Make sure Project Properties> BuildPath > Libraries have hadoop-1.0.3.jar as an imported external library



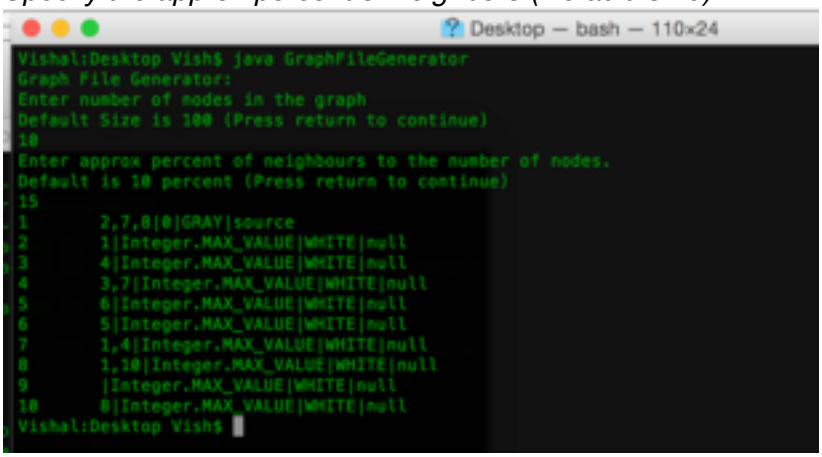
Step 3: Export the project as .jar
Step 4: Paste GraphFileGenerator.java on you desktop. found in Folder name GraphFileGenerator

Step 5: Compile the .java file

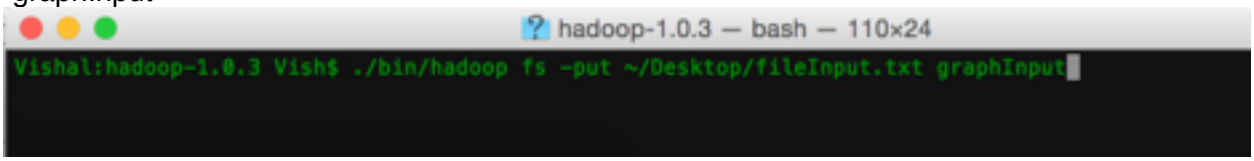


Step6: Run the class file

Specify the number of node (Default is 100)
Specify the approx percent of neighbors (Default is 10)

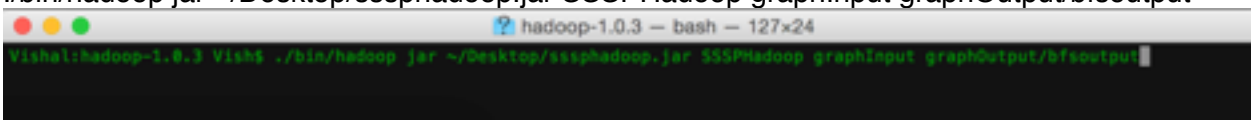


Step 8: Move the fileInput.txt generated on Desktop by GraphFileGenerator to "graphInput"



Step 9: Run the Hadoop job:

./bin/hadoop jar ~/Desktop/sssphadoop.jar SSSPHadoop graphInput graphOutput/bfsoutput



Step 10: Check the output files generated in graphOutput Directory in Hadoop FS

Implementation details:

Mapper: Will get text and pass it to Node class constructor to create Node. Then it will process it by exploring its neighbors and creating them as nodes too.

Partitioner: Will pass the K,V pair to appropriate Reducer [no. of reducers is set to 5]

Reducer: Will club the node with same id into one single node with the darkest col, minimum distance from source and it's predecessor.

Node: node class is used to create nodes.

GraphFileGenerator: is a java program that creates a input file the location where the class is.