ADS Project Report

Aishwarya Barai
UFID: 9195-1975
Email address: aishwarya.barai@ufl.edu

Source code consist of following files :

1. **FibonacciHeap.java :** *Defines implementation for Fibonacci Heap data structure.*
2. **FibonacciHeapNode.java:** *Defines domain object for FibonacciHeap Data node .*
3. **hashtagcounter.java :***Defines the main program which defines the project workflow .*

Classes Structure:

FibonacciHeap.java

Instance Variables:

- **FibonacciHeapNode maxNode:***stores the maximum node in Fibonacci Heap.*
- **int numNodes:** *keeps track of the total number of nodes in Fibonacci Heap***.**

Functions:

- **insert(FibonacciHeapNode node) :** *inserts the data(fibonacciHeapNode) into Fibonacci Heap .*
- **max():***returns the fibonacciHeapNode with the largest key i.e hashtag frequency.*
- **extractMax():** *lemoves the node with largest key from the Fibonacci Heap .This causes the tree to be consolidated if necessary .*
- **consolidate():***link the roots of equal degrees until only one node of each degree left.*
- **link(FibonacciHeapNode y,FibonacciHeapNode x):***link the y fibonacciHeapNode as child of x FibonacciHeapNode*
- **increaseKey(FibonacciHeapNode x, double k):** *increases the key value for a heap node with some k amount. Consolidation is not required in increase key, Adjustments are made based on the child cut value* **.**
- **cut(FibonacciHeapNode x, FibonacciHeapNode y):***removes x from the child list of y.*
- **cascadingCut(FibonacciHeapNode y):** *cuts y from its parent and then does the same for its parent, and so on up the tree.*

FibinacciHeapNode.java

Instance Variables:

- **double key:** *key which keeps the counter of the hashtag and on the basis of this key we will build the fibonacci heap*
- **String data:** *stores the hashtag*
- **FibonacciHeapNode parent:** *reference to the parent node of the current fibonacci heap node*
- **FibonacciHeapNode firstChild:** *refers to the first child node of the current fibonacci heap node*
- **FibonacciHeapNode leftNode:** *refers to the left sibling in fibonacci heap*
- **FibonacciHeapNode rightNode:** *refers to the right sibling in fibonacci heap*
- **boolean childCut:** *stores the child cut value*

- **int degree:** *stores the degree of the fibonacciNode(number of child a fibonacciHeap Node have).*

Functions:

- **FibonacciHeapNode(String data, double key):** *parameterized constructor which Initializes the right and left pointers along with childcut and degree value making this a circular doubly-linked list.*
- **getKey():** *returns the counter of the hashtag from the Fibonacci Heap Node***.**
- **getData():** *returns the hashtag from the Fibonacci Heap Node.*

hashtagcounter.java:

Functions:

- **main(String args[]):***Main function which defines the project workflow . It defines how to find the n most popular hashtags which appear on facebook or twitter from the input file using Fibonacci Heap and hashtable datastructure . Please find the below flow diagram of how it is done in hashtagcounter class .*