

Department of Computer Science and Engg.- NITK Surathkal
CO-360 Advanced Data Structures
Assignment sheet- 2 Due Date: 17th March 2019

Implementation of Dijkstra's Algorithm Using Binary Heaps, Binomial Heaps and Fibonacci Heaps

The objective of this assignment is to compare the performance of binary heaps and binomial queues in implementing the single source shortest path algorithm of Dijkstra, which computes the costs and paths of the shortest cost paths from a source vertex to every other vertex in a labeled digraph with non-negative weights. Note that the dynamic set $V - S$ is implemented using binary heaps or binomial heaps.

Input graph

The graph that is input to the algorithm is either through a simple text file or is generated randomly. In the first case, assume the input to be in the following format:

- Line 1: Number of vertices in the graph
- Line 2: Source vertex (an integer between 1 and n , where n is the number of vertices)
- Line 3: List of immediate neighbours of Node 1 with the weights on associated arcs
- Line 4: List of immediate neighbours of Vertex 2 with the weights on associated arcs
- *etc* ...

In the second case, to generate a random digraph, assume three inputs:

1. Number of vertices, n
 2. Average degree of a node
 3. Range of (integer) weights to be randomly generated for the directed arcs
- Choose an appropriate data structure for the graph.

What is to be done ?

Implement Dijkstra's algorithm, using binary heaps , using binomial heaps and using fibonacci heaps. Make sure to use the standard array data structure for binary heaps and an efficient, appropriate data structure for binomial heaps (for example, look at the book by Alan Weiss). Attempt to do it in C++ but C is also good enough. As usual, special care should be taken to structure your program according to best practices in software engineering: use of good abstractions, smart algorithms, discipline in coding, documentation, provision for exceptions (for example, negative weights should be detected immediately; errors in input should be flagged asap; etc.).

For the input graph and the source vertex, print the following:

- shortest cost path to each vertex and the cost of this shortest path
- Execution times of the program with binary heaps, binomial heaps and fibonacci heaps
- Total number of heap operations executed with binary heaps and binomial queues (this

includes: probes, swaps, link traversals, etc.)

Note: Make groups of 3 students in each team. Each member implement the different heaps for the Dijkstra's Algorithm.