DAA Problem Set 7

Pradeesha Ashok

January 2025

1 Graph Algorithms

1. Given a weighted directed graph G (with only positive edges) and a node s, find the distances of the shortest paths and the count of the number of the shortest paths from s to every other node in the graph.

2. Given a Weighted Directed Acyclic Graph (DAG) and a source vertex s in it, find the longest distances from s to all other vertices in the given graph. (Note: the longest path problem is NP-Hard for a general graph, however it can be done in polynomial time for a DAG).

3. Given a directed graph, a weakly connected component (WCC) is a subgraph of the original graph where all vertices are connected to each other by some path, ignoring the direction of edges. Give an algorithm to find the weakly conected components in a directed graph.

4. A transformation sequence from word beginWord to word endWord using a dictionary wordList is a sequence of words beginWord $\rightarrow$ s1 $\rightarrow$ s2 $\rightarrow$ ... $\rightarrow$ sk such that:

   - Every adjacent pair of words differs by a single letter.
   - Every si for $1 \leq i \leq k$ is in wordList. Note that beginWord does not need to be in wordList.
   - sk == endWord

   Given two words, beginWord and endWord, and a dictionary wordList, return the number of words in the shortest transformation sequence from beginWord to endWord, or 0 if no such sequence exists.

5. Given an m x n grid of characters board and a string word, return true if word exists in the grid.

   The word can be constructed from letters of sequentially adjacent cells, where adjacent cells are horizontally or vertically neighboring. The same letter cell may not be used more than once.

6. Given a directed weighted graph, find whether the given graph contains any negative-weight cycle or not.

7. Given a weighted directed graph, give an algorithm to find the second shortest path in the graph (this path should be a path with a weight strictly greater than the shortest path).

8. Given a weighted connected undirected graph, give an algorithm to remove edges such that the remaining edges in the graph form a tree with the maximum possible weight.