Anthony Rusignuolo

CS 600WS – Advanced Algorithms

Reza Peyrovian

Homework 6

I pledge my honor that I have abided by the Stevens Honor System.

1. R-13.7 Would you use the adjacency list structure or the adjacency matrix structure in each of the following cases? Justify your choice.

a. The graph has 10,000 vertices and 20,000 edges, and it is important to use as little space as possible.  
b. The graph has 10,000 vertices and 20,000,000 edges, and it is important to use as little space as possible.  
c. You need to answer the query are Adjacent as fast as possible, no matter how much space you use.

* 1. Adjacency lists (AL) have O(n+m) space complexity (SC) and O(deg(v)) lookup time (LT). Adjacency matrices (AM) have O(n2) space complexity and O(1) lookup time.  
     a. AL SC < AM SC 🡪 Adjacency list  
     b. AL SC < AM SC 🡪 Adjacency list  
     c. AM LT < AL LT 🡪 Adjacency matrix

1. C-13.11 The directed version of the BFS algorithm classiﬁes non-tree edges as being either back edges or cross edges, but it does not distinguish between these two types. Given a BFS spanning tree, T, for a directed graph G, and a set of non-tree edges, , describe an algorithm that can correctly label each edge in E’ as being either a back edge or cross edge. Your algorithm should run in O(n + m) time, where n is the number of vertices and m is the number of edges.   
   Hint: Consider ﬁrst constructing an Euler tour traversal of the tree T.
   1. Assuming you already have the graph and the non-tree edges separate, you can perform the Euler Tour Traversal to get a list of nodes visited and in what order. For a complete three level binary tree whose nodes are ordered sequentially from left to right top to bottom, that would look like [1,2,3,2,4,2,1,5,6,5,7,5,1]. Iterate through the list of edges (e(a,b)) and if ETT[first occurrence of a:last occurrence of a] is within ETT[first occurrence of b:last occurrence of b] then b is an ancestor of a and therefore edge e(a,b) is a back edge. Otherwise, they are cross edges.
2. A-13.6 A company named RT&T has a network of n stations connected by m high-speed communication links. Each customer’s phone is connected to one station in his or her area. The engineers of RT&T have developed a prototype video-phone system that allows two customers to see each other during a phone call. In order to have acceptable image quality, however, the number of links used to transmit video signals between the two parties cannot exceed 4. Suppose that RT&T’s network is represented by a graph. Design an efﬁcient algorithm that computes, for each station, the set of stations it can reach using no more than 4 links.
   1. The Floyd Warshall Algorithm doesn’t seem to have an easy mechanism for limiting itself to 4 links. Therefore, modify BFS to stop running after i = 4, i.e. either throw “if i = 4 then break” or “if i = 5 then break” above or below the line the says “i 🡨 i + 1”
3. C-14.7 Suppose you are given a connected weighted undirected graph, G, with n vertices and m edges, such that the weight of each edge in G is an integer in the interval [1, c], for a ﬁxed constant c>0. Show how to solve the single-source shortest-paths problem, for any given vertex v, in G, in time O(n + m).  
   Hint: Think about how to exploit the fact that the distance from v to any other vertex in G can be at most O(cn)=O(n).
   1. Run Dijkstra’s algorithm, but instead of using a priority queue, use a lookup table.
4. A-14.2 Suppose that CONTROL, a secret U.S. government counterintelligence agency based in Washington, D.C., has built a communication network that links n stations spread across the world using m communication channels between pairs of stations. Suppose further that the evil spy agency, KAOS, is able to eavesdrop on some number, k, of these channels and that CONTROL knows the k channels that have been compromised. Now, CONTROL has a message, M, that it wants to send from its headquarters station, s, to one of its ﬁeld stations, t. The problem is that the message is super secret and should traverse a path that minimizes the number of compromised edges that occur along this path. Explain how to model this problem as a shortest-path problem, and describe and analyze an efﬁcient algorithm to solve it.
   1. Apply a weight of 0 to uncompromised edges and a weight of 1 to compromised edges. Therefore, the goal is to find the path with the smallest weight, or shortest path, which can be found using Dijkstra’s Algorithm.
5. A-14.5 As your reward for saving the Kingdom of Bigfunnia from the evil monster “Exponential Asymptotic,” the king has given you the opportunity to earn a big reward. Behind the castle there is a maze, and along each corridor of the maze there is a bag of gold coins. The amount of gold in each bag varies. You will be given the opportunity to walk through the maze, picking up bags of gold. You may enter only through the door marked “ENTER” and exit through the door marked “EXIT.” (These are distinct doors.) While in the maze you may not retrace your steps. Each corridor of the maze has an arrow painted on the wall. You may only go down the corridor in the direction of the arrow. There is no way to traverse a “loop” in the maze. You will receive a map of the maze, including the amount of gold in and the direction of each corridor. Describe and analyze an efﬁcient algorithm to help you pick up the most gold in this maze while traversing a path from the start to the ﬁnish.
   1. Represent the maze with an acyclic digraph whose nodes are the corners and paths are the edges of the graph. The edges are weighted with the amount of gold along the way. Then you can run Dijkstra’s Algorithm on the graph, but, instead of searching for the shortest path, you should search for the highest weight, or most gold.