1. The single source shortest path algorithm we studied (Dijkstra’s algorithm) requires time \( \Theta((|E| + |V|) \log |V|) \) to find the minimum distance \( d_i \) from the source vertex to vertex \( i \), \( 1 \leq i \leq n \). Checking the answer seems to be easier than finding it. Give a linear time algorithm that, given a graph \( G \) and a sequence of distances \( d_i; 1 \leq i \leq n \), will verify whether these \( d_i \)'s are indeed the minimum distances from the source to each vertex \( i \). Argue the correctness of your algorithm and analyze its complexity.

2. Text page 531, exercise 25.2-2. The second part of the question can be answered relative to either the proof given in class or the proof of 25.10 in the book.


5. [“Spreadsheet Evaluation”] Let \( G \) be a directed acyclic graph having a \( k \)-ary function symbol \( f_v \) (like plus, times, etc.) associated with each vertex \( v \) of out-degree \( k \). (A 0-ary “function” is just an integer constant.) Define the “value” of a vertex \( v \) to be the value obtained (recursively) by applying the function \( f_v \) to the values of the vertices pointed to by \( v \)'s out edges.

(a) Give an algorithm to compute the values of each vertex in \( G \).

(b) Breadth First Search wouldn’t be a good choice for this task. Give a simple example and briefly sketch why (or prove me wrong).