CS302 - Problem Set 9

1. [11 points] Prove the BFS shortest path algorithm I described in class is correct. That is, prove that if the shortest distance from \(s\) to \(v\) is \(n\), then \(L[v] = n\) (and if \(s\) and \(v\) are not connected by a path, then \(L[v] = \infty\)).

   (See final page for hints.)

2. [6 points] What is the runtime of the BFS shortest path algorithm I described in class if the graph is given as an adjacency matrix? (Let \(n\) be the total number of vertices in the graph, \(m\) the total number of edges, \(n_s\) the number of vertices connected to \(s\) by a path, and \(m_s\) the number of edges connected to \(s\) by a path.)
Hint for proof: I have two loop invariants: one regarding the order in which elements are added to the queue (needed for maintenance), and one regarding the correct assigning of shortest lengths (needed for termination)