

CSC263: Problem Set 7

1 Problem 2

- (a) We will start by converting the adjacency matrix to a adjacency list. This occurs in linear time because we simply scan each row and append each edge (where it exists) to the created adjacency list. We scan only the rows (or columns) because this is an undirected graph, so the matrix is symmetric. To find vertices whose removal would not disconnect the graph, we start with the graph and make a list called *removable_vertex* which keeps track of all the vertices whose removal will not disconnect the graph. First keep a copy of the default graph without the removal of a vertex. Then we remove the first vertex x . Once vertex x has been removed, we use the BFS algorithm choosing the s vertex at random (can use any vertex from $G.V - \{x\}$), and the BFS will also be augmented to track of all the vertices visited in a list. Once the BFS algorithm is done running it will return a list of all vertices visited, we compare the list to $G.V - \{x\}$. If the list contains all the vertices that are in $G.V - \{x\}$ then removal of vertex x will not disconnect the graph and therefore this vertex should be added to the *removable_vertex* list. Afterwards the graph should be reset its default state (before removal of vertex x) and the process continues from the start where a vertex which has not been removed will be removed and checked if the vertex is a removable vertex or not. This will continue until all vertices have been removed once and checked if their removal causes the graph to become disconnected. Once this is complete, *removable_vertex* will have a list of vertices whose removal will not disconnect the graph.
- (b) This algorithm is correct since you remove every vertex once and then run BFS which keeps a list of all vertices visited to see if every vertex in $G.V - \{\text{The vertex removed}\}$ is visited. This is done for every vertex in $G.V$ to identify all removable vertices.
- (c) The worst case analysis of this algorithm is $O(|V| + |V| \cdot (|V + E| + |V|^2))$. The initial $|V|$ comes from converting the adjacency matrix to an adjacency list. Then, BFS runs in $O(|V + E|)$ (page 597), and is repeated $|V|$ times since there are $|V|$ vertices in V . The $|V|^2$ comes from comparing the list of vertices visited to $G.V - \{x\}$ each time the algorithm is run. Note: appending to the list occurs in constant time, we do not include that. So this algorithm runs in $O(n^3)$.