Q1:

Input: A directed graph $G = (V, E)$ in adjacency list

output: Reverse of graph G^R

Algorithm: // intially create the graph with edge set empty.

Creat the graph $G^R = (V, E^R)$ with the edge-set

E^R (initially is empty).

// then, execute the for loop for each edge of u.

for each $u \in V$:

// then, execute the for loop for each edge of (u, w)

for each $(u, w) \in E$:

// add the edge into the list

add edge (w, u) to E^R

// Return the reversed graph

return G^R

// The time required to generate all edge is $O(V+E)$,

and the time required for modification of edges and

make new adjacency list is $O(E)$.

Thus, the time algorithm is in linear time.