Homework: Graph Algorithms

Due April 15th. No extensions.

1. Give an efficient algorithm (in Python pseudo-code) that calculates the transpose of a directed graph $G = (V, E)$ given using an adjacency list. The transpose is the same graph, but the direction of all the vertices is changed. Mathematically, the transpose is $G^T = (V, E^T)$ with $E^T = \{(u, v) \mid (v, u) \in E\}$.

2. For a directed graph $G = (V, E)$ prove the triangle inequality for the shortest path distance: $\forall u, v, w \in V \; : \; \delta(u, v) + \delta(v, w) \geq \delta(u, w)$.