CSE 204
Offline: Graph
Date: 16/01/2023
Due: 20/01/2023Time: 23:30 pm

Task 1: Implement Breadth First Search Algorithm and output the BFS tree
Task 2: Implement Depth First Search Algorithm and output the DFS tree

Input: The first line of the input file will contain the number of vertices *n* (≤ 1000) and the number of edges *m* (≤ 10000) followed by *m* lines each containing origin *u*, end *v* of an edge of the directed graph. The last line will contain a source vertex *s*.

Output:

- The *predecessor subgraph* of G as $G_\pi = (V, E_\pi)$, where
  $$V = \{v \in V : v.\pi \neq \text{NIL}\} \cup \{s\} \text{ and } E_\pi = \{(\text{v}.\pi, \text{v}) : \text{v} \in V_\pi - \{s\}\}$$
  The predecessor subgraph G_π is a *breadth-first tree* if V_π consists of the vertices reachable from s and, for all $v \in V_\pi$, the subgraph G_π contains a unique simple path from s to v that is also a shortest path from s to v in G

- *Predecessor subgraph* of G as $G_\pi = (V, E_\pi)$, where
  $E_\pi = \{(\text{v}.\pi, \text{v}) : \text{v} \in \text{V and v}.\pi \neq \text{NIL}\}$.(The edges in E_π are called tree edges.)