Module 2 - Graphs: Basic Written HW

1. Describe an algorithm that, given a directed graph G represented as an *adjacency matrix*, returns whether or not the graph contains vertex with in-degree $|V| - 1$ and out-degree 0. In other words, does the graph have a node such that every other node points to it, but it does not point to any other node. Your algorithm must be $O(V)$. Note that there are $\Theta(V^2)$ cells in your adjacency matrix so you'll need to be clever here.

2. Let's say a graph G's *circumference* is the number of edges in the shortest cycle in G. Describe an efficient algorithm to find the circumference of a graph in $\Theta(V \times E)$. (Note: you must make use of algorithms studied in this module, and not re-invent the wheel.)

3. Let G be an undirected graph with n nodes (let's assume n is even). Prove or provide a counterexample for the following claim: If every node of G has a degree of at least $\frac{n}{2}$, then G must be connected.

4. For a given undirected graph G, prove that the depth of a DFS tree cannot be smaller than the depth of the BFS tree. (Clearly state your proof strategy or technique.)