CS6905 (AGA) Winter 2023 – Assignment 1 (Minor)
Due Friday January 20, 2023, by 5pm.

The `GraphWtAL.java` file (on Desire2Learn) provides the `GraphWtAL` class, used to build and store a static undirected graph with edge weights. It uses node indexing (from 0) and adjacency lists (built using an internal class `GNode`) as the underlying data structure, and also has an array of integers (one for each node) to be used to mark nodes. The two adjacency list entries for each edge are linked, and there is also a mark field on each of them that can be used.

Its constructor sets up the array of adjacency lists and array of vertex marks.

`reset()` takes an integer parameter and sets all vertex marks to that integer. This method is to be used to initialize marks to the desired default value.

`addEdge()` takes a pair of indices (x and y) as parameters, and inserts the edge (x, y) between them into the adjacency list representation.

`toString()` converts the adjacency lists (and current vertex marks) into a string, with one vertex+list per line.

You need to write a `GraphReach` class that extends `GraphWtAL`, to add the following method:

- `reach(int, int)`: takes a pair of vertex indices as its parameter, and returns a Boolean that is **true** if the second vertex can be reached from the first vertex, **false** otherwise.

Your class will also need a constructor to pass on the size parameter to the `GraphAL` constructor, and should also have private methods and classes as appropriate. Your class should not have any additional variables declared globally for the class.

Ensure that your code works with the provided `DriverAGA1.java` code, which will be used to test your submitted solution. Organize and comment your code appropriately.

Submit on D2L: your `GraphReach.java` file, and the I/O from one test run of your solution.