CS6905 (AGA) Fall 2023 – Assignment 2 (Major)
Due Wednesday September 27, 2023, by 10pm.

The `GraphWtAL.java` file (on Desire2Learn) provides the `GraphWtAL` class, used to build and store a static undirected graph with edge weights, as described in Assignment 1.

For graphs, we want to be able to calculate the scope of a vertex (as described in the class slides) and also find common scope of multiple vertices.

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

- public array `scope[]` of integers, of the same size as the graph. `scope` needs to be allocated by your constructor and its values initialized to 1. Its contents will be updated by method `findScope()` so that it contains a 1 at each position whose vertex is in the k-scope of all vertices that have been searched so far, and 0 otherwise.
- method `markScope(int, int)`: takes a vertex u and scope size k as parameters, and marks the vertices with their distance (in edges) from u, up to and including distance k (as described in the Scope algorithm). Vertices more than k edges away from u should have a mark of -1. This method should return *void*.
- method `findScope(int, int)`: takes a vertex u and scope size k as parameters, runs `markScope(u,k)` to mark the vertices in the k-scope of the new vertex, and sets `scope`$[v]$ to 0 for any vertices v that are not in u's k-scope. This method should return an integer that is the number of vertices that are still in the common scope of all vertices so far.

Your code should be adapted from your previous code for Assignment 1 (`GraphReach.java`), but not use it separately.

Ensure that your code works with the provided `DrAGA2.java` code, which will be used to test your submitted solution. Organize and comment your code appropriately.

Submit on D2L: your `GraphScope.java` file, and the I/O from one test run of your solution. Your test run needs to show how well your method works.