Skyler Benjamin

Graph Complexity

Map and Set are assumed to be HashMap and HashSet implementations

numVertices: O(1)

* Map.size() is a constant time op

numEdges: O(1)

* Edges is a store variable that can be instantly recalled

clear: O(1)

* Map.clear() is a constant time op

addVertex: O(1)

* Map.contains() and map.put() are constant time ops
* Map.contains() is a get op that throws away the result

addEdge: O(1)

* Map.contains() and map.get() are constant time ops
* Map.contains() is a get op that throws away the result

getVertices: O(1)

* Map.keySet() is a constant op because it returns the full object without iterating

getNeighbors: O(1)

* Map.contains() and map.get() are constant time ops

containsVertex: O(1)

* Map.containsKey() is a get op that evaluates and disposes of the result

edgeExists: O(1)

* Map.contains() and map.get() are constant time ops
* Set.contains() is also constant because of Hashing

degree: O(1)

* Map.containsKey(), map.get(), set.size() are all constant time ops

toString: O(n\*m) , n = vertices, m = edges

* For each map entry (n), we loop through its children (m) to build the string