Technical Documentation of the Graph Class for the CPSC 374 Bonus Project

**AbstractGraph**

* AbstractGraph is the abstract class for the graph class and implements the BaseOpsInt interface. The methods AbstractGraph make abstract from BaseOpsInt are size(), isEmpty(), and clear(). All the other methods from BaseOpsInt are null code.
  + int size()
    - returns the number of vertices in the graph class
  + boolean isEmpty()
    - returns true if the graph class contains no vertices. Returns false otherwise.
  + boolean clear()
    - clears the graph of all values. Always returns true.
* AbstractGraph adds 3 more abstract methods
  + boolean containsEdge(Object src, Object dest)
    - This method returns true if there is an edge between the source node and the destination node. Returns false otherwise.
  + boolean containsVertex(Object value)
    - This method returns true if value is found in the graph. Returns false otherwise.
  + boolean removeVertex(Object value)
    - This method returns true if value is located and successfully removed from the graph data structure. Returns false otherwise.
    - This method also removes all the edges associated with this value.

**Graph Class**

* The graph class works by using a HashMap that maps a value to a singly linked list to other values that are the edges for that value.
* A linked list was used to hold the edges of the mapped value because a linked list is better for removing and inserting values which the graph class must use.
* The graph class can be either unidirectional or bidirectional.
* The graph class implements 5 new methods
  + boolean addEdge(Object src, Object dest, boolean direction)
    - This method creates a new edge between two vertices with a specified direction.
    - direction is true for unidirectional or false for bidirectional.
    - Will add the new vertices to the graph if they are not already part of the graph.
    - Returns true if the edge was successfully added or false if the edge already exists or the edge was not successfully added.
  + int checkEdgeDirection(Object src, Object dest)
    - This method checks the direction of an edge for any 2 vertices.
    - Returns 1 for a unidirectional edge from source to destination
    - Returns 2 for a bidirectional edge between the two vertices
    - Returns 0 if there is no edge between the two vertices or if there is no edge from source to destination.
  + Boolean removeEdge(Object src, Object dest, boolean direction)
    - This method removes an edge between two vertices
    - Set direction to true to remove a unidirectional edge or to remove one direction to a bidirectional edge.
    - Set direction to false to remove a bidirectional edge.
    - Returns true if an edge was deleted. Returns false otherwise.
  + boolean addVertex(Object value)
    - adds a vertex to the graph without specifying any edges with any other vertices.
    - always return true.
  + SinglyLinkedList getEdgesOfVertex(Object value)
    - This method returns a singly linked list of all the vertices this vertex is connected to.
    - Returns null if there is no such vertex in the graph.
  + toString overload
    - returns a string that neatly formats each vertex and its associated edges.
    - The string is as follows

20 – 10, 30 (20 has edges with 10 and 30)

40 – 10, 30 (40 has edges with 10 and 30)

10 – 20, 40 (10 has edges with 20 and 40)

30 – 20, 40 (30 has edges with 20 and 40)