

Vertex.java

```
1 //*****
2 //
3 // File:    Vertex.java
4 // Package: ---
5 // Unit:    Class Vertex
6 //
7 //*****
8
9 import java.util.ArrayList;
10
11 /**
12  * Class Vertex represents a single vertex in a graph. Vertices can be connected
13  * to other vertices through undirected edges.
14  *
15  * @author Jimi Ford
16  * @version 2-15-2015
17  */
18 public class Vertex {
19
20     // private data members
21     protected ArrayList<UndirectedEdge> edges = new ArrayList<UndirectedEdge>();
22
23     /**
24      * The unique identifier for this vertex
25      */
26     public final int n;
27
28     /**
29      * Construct a vertex with a unique identifier <I>n</I>
30      *
31      * @param n the unique identifier to distinguish this vertex from
32      *         all other vertices in the graph
33      */
34     public Vertex(int n) {
35         this.n = n;
36     }
37
38     /**
39      * Get the number of edges connected to this vertex
40      *
41      * @return the number of edges connected to this vertex
42      */
43     public int degree() {
44         return edges.size();
45     }
46
47     /**
48      * Get the reference to the collection of edges connected to
49      * this vertex.
50      *
51      * @return the reference to the collection of edges
52      */
53     public ArrayList<UndirectedEdge> getEdges() {
54         return this.edges;
55     }
56
57     /**
58      * Add an edge to this vertex
```

```
59      *
60      * @param e the edge to add
61      */
62      public void addEdge(UndirectedEdge e) {
63          this.edges.add(e);
64      }
65
66      /**
67       * Compare another object to this one
68       *
69       * @param o the other object to compare to this one
70       * @return true if the other object is equivalent to this one
71       */
72      public boolean equals(Object o) {
73          if( !(o instanceof Vertex)) {
74              return false;
75          }
76          if(o == this) {
77              return true;
78          }
79          Vertex casted = (Vertex) o;
80
81          return casted.n == this.n;
82      }
83 }
84
```