```java
// File:    Vertex.java

import java.util.ArrayList;

/**
 * Class Vertex represents a single vertex in a graph. Vertices can be connected
 * to other vertices through undirected edges.
 *
 * @author Jimi Ford
 * @version 2-15-2015
 */
public class Vertex {

    // private data members
    protected ArrayList<UndirectedEdge> edges = new ArrayList<UndirectedEdge>();

    /**
     * The unique identifier for this vertex
     */
    public final int n;

    /**
     * Construct a vertex with a unique identifier <I>n</I>
     *
     * @param n the unique identifier to distinguish this vertex from
     *          all other vertices in the graph
     */
    public Vertex(int n) {
        this.n = n;
    }

    /**
     * Get the number of edges connected to this vertex
     *
     * @return the number of edges connected to this vertex
     */
    public int degree() {
        return edges.size();
    }
```

```
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
```