```java
//*********************************************************************
//
// File:    CricketObserver.java
// Package: ---
// Unit:    Class CricketObserver
//
//*********************************************************************

/**
 * Class observes a group of crickets for a given number of time ticks and
 * keeps track of whether or not they have chirped or not.
 *
 * @author Jimi Ford (jhf3617)
 * @version 3-31-2015
 */
public class CricketObserver {

    /**
     * the number of crickets being observed
     */
    public final int crickets;

    /**
     * the number of time ticks observing for
     */
    public final int ticks;

    // private data members
    private boolean[][] chirps;

    /**
     * Construct a cricket observer
     * @param crickets the number of crickets to observe
     * @param ticks the number of time ticks observing for
     */
    public CricketObserver(int crickets, int ticks) {
        this.crickets = crickets;
        this.ticks = ticks;
        chirps = new boolean[ticks][crickets];
    }

    /**
     * called by a cricket to inform the observer that he has chirped
     * @param tick the time tick at which the cricket is chirping
     * @param n the unique identifier of the cricket
     */
    public void reportChirp(int tick, int n) {
        chirps[tick][n] = true;
    }

    /**
     * lookup a given time and cricket to see if it chirped at that moment
     * @param tick the moment in time to lookup
     * @param cricket the unique identifier of the cricket to check
     * @return true if it chirped
     */
    public boolean chirped(int tick, int cricket) {
        return chirps[tick][cricket];
```

```
59      }
60
61      /**
62       * get the time tick at which all the crickets being observed synchronized
63       * @return a number >= to 0 if they synchronized, -1 if they didn't
64       */
65      public int sync() {
66          int row = 0;
67          while(row < ticks) {
68              if(sync(row)) return row;
69              row++;
70          }
71          return -1;
72      }
73
74      /**
75       * determine whether the crickets were synchronized at a given time tick or
76       * not
77       * @param tick the time tick to test
78       * @return true if every cricket at this time tick chirped
79       */
80      private boolean sync(int tick) {
81          boolean retval = true;
82          for(int i = 0; i < crickets && retval; i++) {
83              retval = chirps[tick][i];
84          }
85          return retval;
86      }
87  }
88
```