3. This question involves reasoning about the GridWorld case study. Reference materials are provided in the appendixes. In part (a) you will write a method to return an array list of all empty locations in a given grid. In part (b) you will write the class for a new type of Critter.

(a) The GridWorldUtilities class contains static methods. A partial declaration of the GridWorldUtilities class is shown below.

```
public class GridWorldUtilities
{
   /**  Gets all the locations in grid that do not contain objects.
    *   @param grid a reference to a BoundedGrid object
    *   @return  an array list (possibly empty) of empty locations in grid.
    *            The size of the returned list is 0 if there are no empty locations in grid.
    *            Each empty location in grid should appear exactly once in the returned list.
    */
   public static ArrayList<Location> getEmptyLocations(Grid<Actor> grid)
   {   /* to be implemented in part (a) */   }

   // There may be instance variables that are not shown.
}
```

Write the GridWorldUtilities method getEmptyLocations. If there are no empty locations in grid, the method returns an empty array list. Otherwise, it returns an array list of all empty locations in grid. Each empty location should appear exactly once in the array list.

WRITE YOUR SOLUTION ON THE NEXT PAGE.

Complete method `getEmptyLocations` below.

```
/**  Gets all the locations in grid that do not contain objects.
 *   @param grid a reference to a BoundedGrid object
 *   @return  an array list (possibly empty) of empty locations in grid.
 *            The size of the returned list is 0 if there are no empty locations in grid.
 *            Each empty location in grid should appear exactly once in the returned list.
 */
public static ArrayList<Location> getEmptyLocations(Grid<Actor> grid)
```

Part (b) begins on page 14.

GO ON TO THE NEXT PAGE.