

Quiz 18

Name: \_\_\_\_\_ Grade: \_\_\_\_\_

1. Fill in the blanks to specify LiftInterface in the left column and Lift in the right column:

```

public interface LiftInterface {
  /* You don't need to specify anything about these constants. */
  public final int UP = 1;
  public final int STOP = 0;
  public final int DOWN = -1;

  /**** MODEL FIELDS ****/
  // @ public model instance _____;

  // @ public model instance _____;

  // @ public model instance _____;

  // @ public model instance _____;

  // @ public model instance _____;

  // @ public model instance _____;

  /**** INVARIANTS ****/
  /* @
  @
  @ public invariant _____;
  @
  @ public invariant _____;
  @
  @ public invariant _____;
  @
  @ public invariant _____;
  @
  @ public invariant _____;
  @
  @ public invariant _____;
  @ */
  
```

```

class Lift implements LiftInterface {
  /**** REPRESENTATION ****/
  private int location; // @ in _____;

  // @ private represents _____;

  private boolean[] floorRequestLight;
  // @ maps floorRequestLight[*] \into requests;
  /* @ private represents requests \such_that
  @ (\forall int i; 0<=i && i<floorRequestLight.length;
  @ floorRequestLight[i] <==> requests.contains(i+1));
  @
  @ private represents topFloor <- floorRequestLight.length;
  @ */

  private boolean upArrowLight; // @ in arrows

  /* @ private represents arrows \such_that
  @ upArrowLight <==> (arrows == UP);
  @ */

  private boolean downArrowLight; // @ in _____;

  /* @ private represents _____
  @
  @ _____;
  @
  @ */

  private int nextDirection; // @ in _____;

  // @ private represents _____;

  private boolean doorsOpen; // @ in _____;

  // @ private represents _____;
  
```

## LiftInterface

```

/*@
@ public initially -----;
@
@ public initially -----;
@
@ public initially -----;
@
@ public initially -----;
@
@ public initially -----;
@*/

/*@
@ requires -----;
@
@ assignable -----;
@
@ ensures -----;
@*/
public void buttonPressed(int floor);

/*@
@ requires -----;
@
@ assignable -----;
@
@ ensures -----;
@*/
public boolean isFloorRequestLit(int floor);
// rest elided ...

```

## Lift class

```

/*@
@ requires -----;
@
@ assignable -----;
@
@ ensures -----;
@*/
public Lift(int numberOfFloors) {
    this.location = 0;
    this.floorRequestLight = new boolean[numberOfFloors];
    this.upArrowLight = false;
    this.downArrowLight = false;
    this.nextDirection = STOP;
    this.doorsOpen = false;
}

public void buttonPressed(int floor) {
    this.floorRequestLight[floor - 1] = true;
}

public boolean isFloorRequestLit(int floor) {
    return this.floorRequestLight[floor - 1];
}

// rest elided ...

```