20th Aug, 2015

**Name**：陈炜健　**Student ID**：**13331018**

---

## Do You Know?

1. What is the role of the instance variable sideLength?

answer: The instance variable sideLength is the total step number that the BoxBug can move on each side of the box.

2. What is the role of the instance variable steps?

answer: The instance variable steps is the the step number that the BoxBug has already moved each side of the box.

3. Why is the turn method called twice when steps becomes equal to sideLength?

answer: Using one turn method, the BoxBug will turn right 45 degrees. In this case, because the BoxBug need to turn 90 degrees in order to move a square path, the turn method need to be called twice.

4. Why can the move method be called in the BoxBug class when there is no move method in the BoxBug code?

answer: Because the BoxBug extend the Bug, the BoxBug can inherit the move method which has been defined inside the Bug code.

5. After a BoxBug is constructed, will the size of its square pattern always be the same? Why or why not?

answer: Yes. Its square pattern always be the same because the sideLength of each BoxBug is specified and will not be changed in run time.

6. Can the path a BoxBug travels ever change? Why or why not?

answer: Yes, it can. When a BoxBug comes across a Rock or other BoxBug, it will change its direction.

7. When will the value of steps be zero?

answer: At the beginning, the steps of the BoxBug is zero. At the run time, the steps will be zero when the BoxBug comes cross a blocked object like wall or other bug, or when the steps is qual to sideLength.

## Exercise

1. Write a class CircleBug that is identical to BoxBug, except that in the act method the turn method is called once instead of twice. How is its behavior different from a BoxBug?

answer: If the gird space is big enough, the CircleBug will move with the shape of regular hexagon.

5. Study the code for the BoxBugRunner class. Summarize the steps you would use to add another BoxBug actor to the grid.

answer:

I. Create a BoxBug object with length and a ActorWorld object.

```
BoxBug alice = new BoxBug(6);
ActorWorld world = new ActorWorld();
```

II. Add the BoxBug into the ActorWorld.

```
world.add(new Location(7, 8), alice);
```