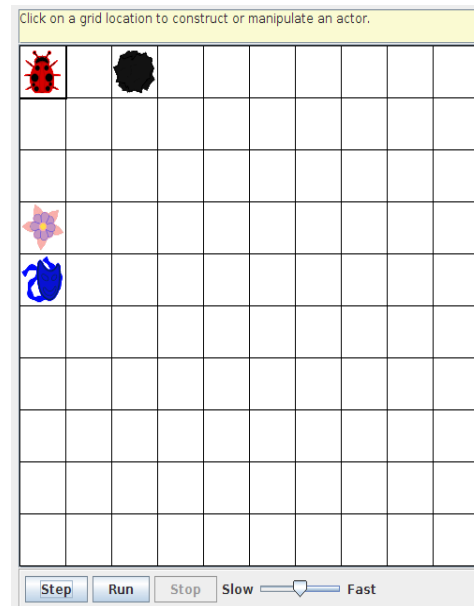


## Test report

Initial world, the jumper's location is (4,0)

```
ActorWorld world = new ActorWorld();
Jumper jumper = new Jumper();
world.add(new Location(4, 0), jumper);
world.add(new Location(0, 0), new Bug());
world.add(new Location(3, 0), new Flower());
world.add(new Location(0, 2), new Rock());
world.show();
```



Test 1:

If there is another actor one cell in the front of the jumper, the jumper will jump over it.

```
jumper.act();
assertEquals(new Location(2, 0), jumper.getLocation());
```

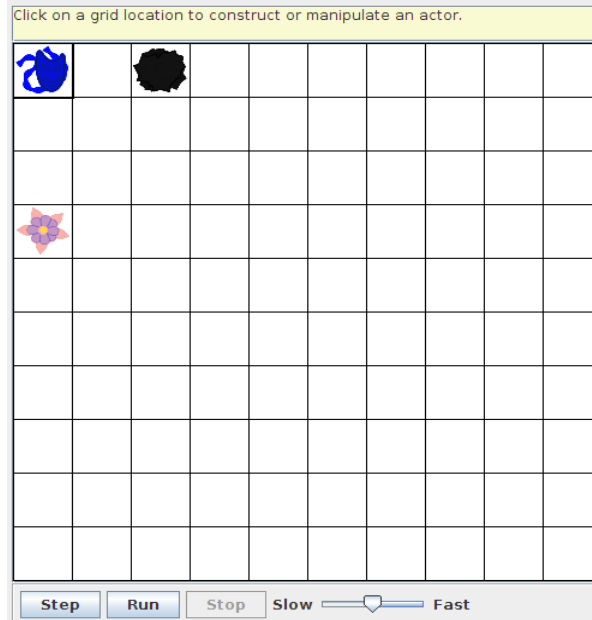
Jumper acts once, test if the jumper's new location is (2,0)



Test 2: If a bug is two cells in front of the jumper, the jumper will jump and eat the bug.

```
jumper.act();  
assertEquals(new Location(0, 0), jumper.getLocation());
```

The jumper jumps and eats the bug, so its new location is the bug's location.

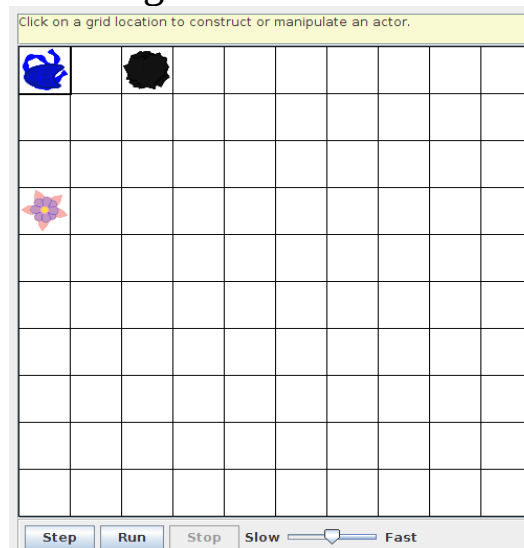


Test 3:

If the jumper is facing the edge of grid, it will turn right 90 degree.

```
jumper.act();  
assertEquals(90, jumper.getDirection());
```

Because the jumper's previous direction is north, it's 0 degree. After turn right 90 degree, its direction is 90 degree.



```
jumper.act();
assertEquals(135, jumper.getDirection());
```

Click on a grid location to construct or manipulate an actor.

The grid world environment is a 10x10 grid. The top row is a yellow header with the text "Click on a grid location to construct or manipulate an actor." The grid contains the following objects:

- A blue robot at (0,0).
- A black rock at (1,0).
- A purple flower at (0,3).
- A tooltip at (6,8) indicating "(6, 8) is s".

The bottom of the interface features a control bar with the following elements:

- Buttons: Step, Run, Stop.
- A slider between Slow and Fast.
- A button: Fast.

```
jumper.act();
assertEquals(new Location(2, 2), jumper.getLocation());
```

Click on a grid location to construct or manipulate an actor.

(6, 8) is empty

The final result of Junit test

```
File Edit View Search Terminal Help
JUnit version 4.9
.
Time: 0.647

OK (1 test)
```