Jumper

测试报告
Version 1.0.0

目录

1 测试计划

测试项目：

a. What will a jumper do if the location in front of it is empty, but the location two cells in front contains a flower or a rock?

- Expect: Jumper will turn right of 45 degrees.

b. What will a jumper do if the location two cells in front of the jumper is out of the grid?

– Expect: Jumper will turn right of 45 degrees.

c. What will a jumper do if it is facing an edge of the grid?

– Expect: Jumper will turn right after 2 steps.

d. What will a jumper do if another actor (not a flower or a rock) is in the cell that is two cells in front of the jumper?

– Expect:

e. What will a jumper do if it encounters another jumper in its path?

– If they encounters adjacent to each other face to face -- Expect: Both jump to the location behind them

– If they encounters two cells away and face to face – Expect: Both turn right

f. What if there is a flower two cells in front of the Jumper?

– Expect: Arrive at the location and eat the flower

g. What will the Jumper do if there is an empty location two cells in front of the Jumper?

– Expect: Jump 2 cells in 1 step.

分别设计以上 9 种场景，观察几步后 Jumper 的状态是否和预期相同，如果相同则为测试通过，不满足预期即未通过测试。

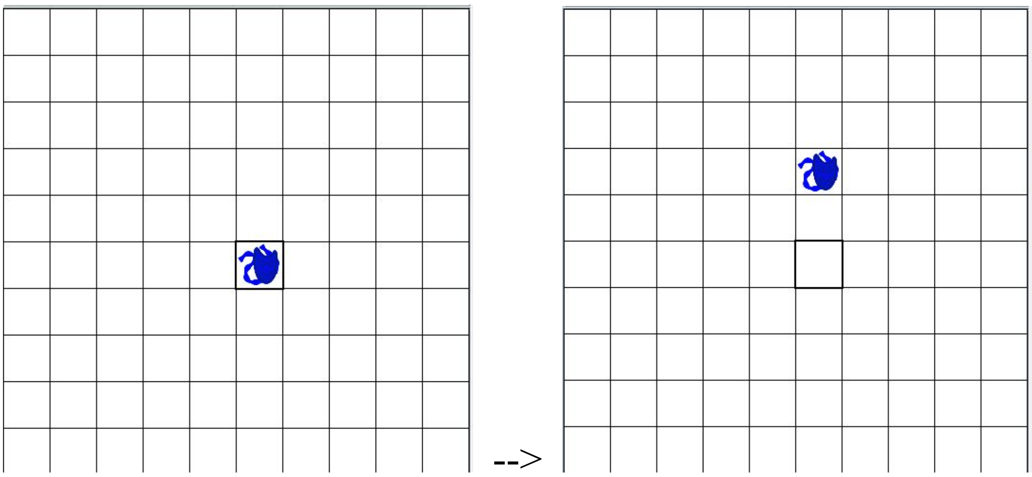
2 用例 1

测试目的：测试 Jumper 是否能在前方两格处在界内且空白的情况下，一步跳两格。

2.1 测试用例

```
Initial: Jumper Location (5, 5) Direction(NORTH)
Expect: Jumper Location (3, 5) Direction(NORTH)
```

2.2 测试结果



2.3 结果分析

每个 step 前进两步，实际与预期结果一致。

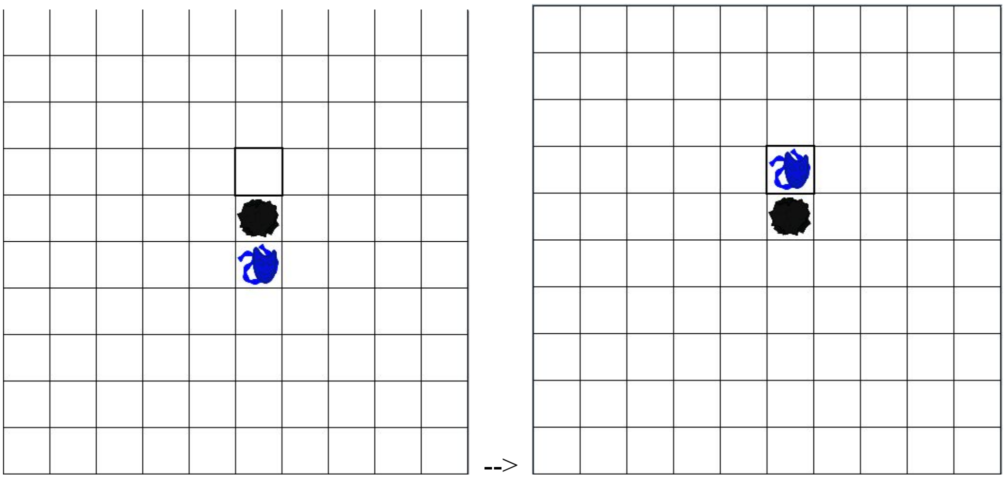
3 用例 2

测试目的：测试 Jumper 是否能在前方一格处有石头的情况下跳过石头。

3.1 测试用例

```
Initial: Jumper Location (5, 5) Direction(NORTH)
         Rock Location (4, 5)
Expect: Jumper Location (3, 5) Direction(NORTH)
```

3.2 测试结果



3.3 结果分析

Jumper 跳过石头，实际与预期结果一致。

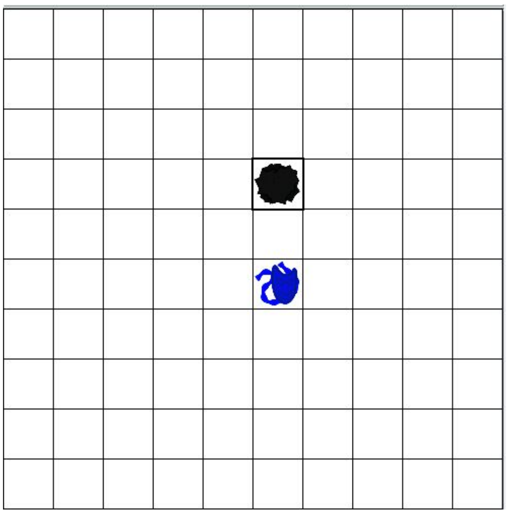
4 用例 3

测试目的：测试 Jumper 是否能在前方一格处有石头的情况下跳过石头。

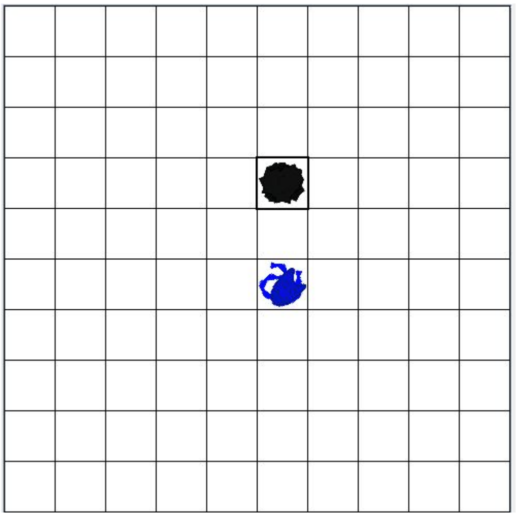
4.1 测试用例

```
Initial: Jumper Location (5, 5) Direction(NORTH)
          Rock Location (3, 5)
Expect: Jumper Location (5, 5) Direction(NORTHEAST)
```

4.2 测试结果



-->



4.3 结果分析

Jumper 发现石头向右转 45 度， 实际与预期结果一致。

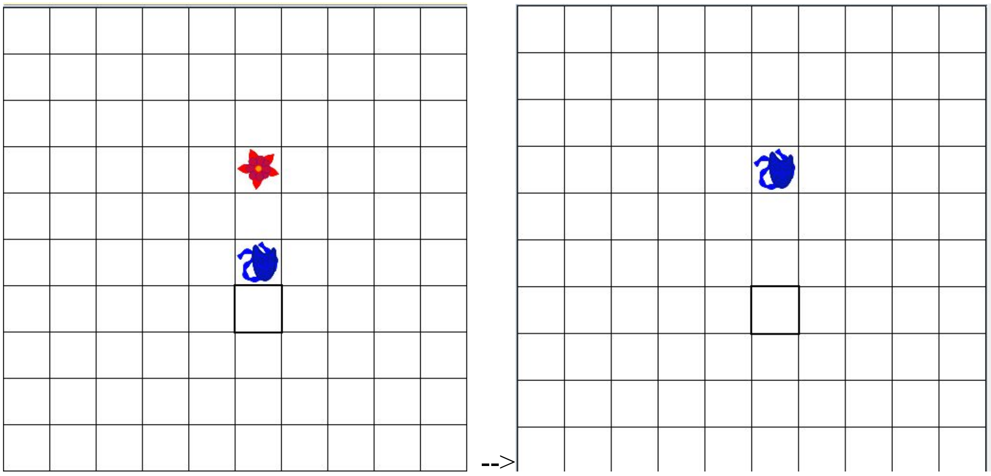
5 用例 4

测试目的：测试 Jumper 是否吃掉下一步到达位置的花朵。

5.1 测试用例

```
Initial: Jumper Location (5, 5) Direction(NORTH)
         Flower Location (3, 5)
Expect: Jumper Location (3, 5) Direction(NORTH)
```

5.2 测试结果



5.3 结果分析

Jumper 到达花朵所在位置，并且将花朵移除场景，实际与预期结果一致。

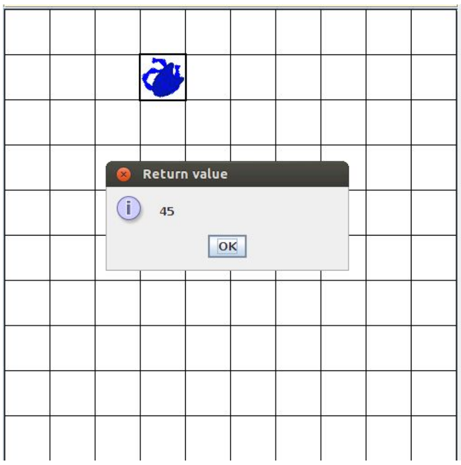
6 用例 5

测试目的：测试 Jumper 是否能在下一步位置出界情况下转向。

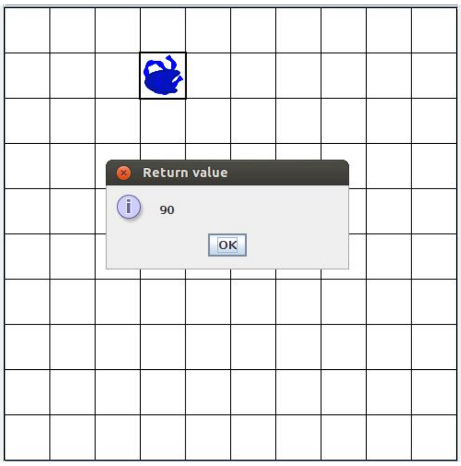
6.1 测试用例

```
Initial: Jumper Location (1, 3) Direction(NORTHEAST)
Expect: Jumper Location (1, 3) Direction(EAST)
```

6.2 测试结果


Return value

45


Return value

90

6.3 结果分析

Jumper 检测到下一步出界，所以右转 45 度，实际与预期结果一致。

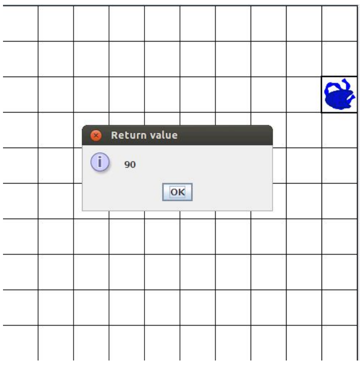
7 用例 6

测试目的：测试 Jumper 是否在面对边界的情况下转向。

7.1 测试用例

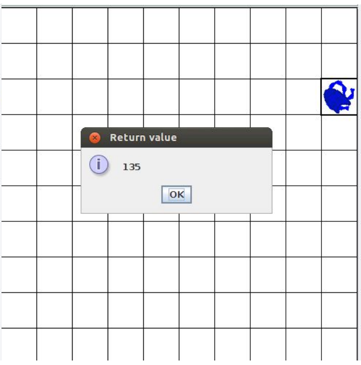
```
Initial: Jumper Location (2, 9) Direction(EAST)
Expect: STEP1: Jumper Location (2, 9) Direction(SOUTHEAST)
        STEP2: Jumper Location (2, 9) Direction(SOUTH)
```

7.2 测试结果


Return value

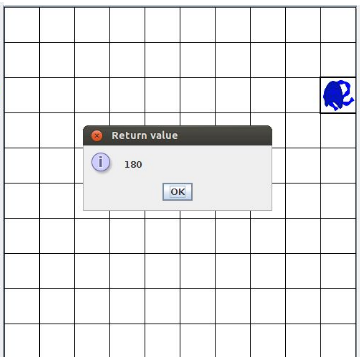
90

OK


Return value

135

OK


Return value

180

OK

7.3 结果分析

Jumper 在面对边缘的情况下，两步下右转 90 度，实际与预期结果一致。

8 用例 7

测试目的：测试 Jumper 是否能正确处理与其他角色（bug）的相遇问题。总共包括三种情况。

8.1 测试用例

三种情况： Jumper 在下，Bug 在上两格，同时向上：Bug 先活动，给 jumper 移开置，Jumper 正常前进；

Jumper 在上，Bug 在下两格，同时向下，Jumper 先活动，被挡住所以右转，Bug 正常前进；

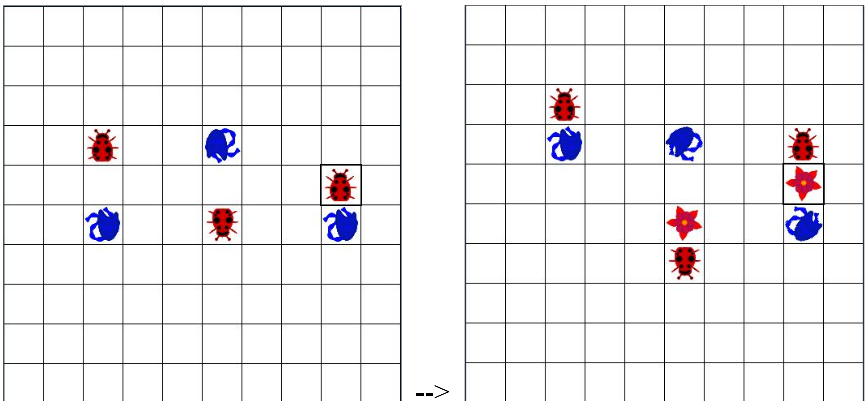
Jumper 在下，Bug 在上一格，同时向上，Bug 先前进一步，挡住了 Jumper 的路，Jumper 转向。

Initial: (1) Jumper Location (5, 2) Direction(NORTH)
Bug Location (3, 2) Direction (NORTH)
(2) Jumper Location (3, 5) Direction(SOUTH)
Bug Location (5, 5) Direction (SOUTH)
(3) Jumper Location (5, 8) Direction (NORTH)
Bug Location (4, 8) Direction (NORTH)

Expect: (1) Jumper Location (3, 2) Direction(NORTH)
(2) Jumper Location (3, 5) Direction (SOUTHWEST)

```
          (3) Jumper Location (5, 8) Direction (NORTHEAST)
```

8.2 测试结果



8.3 结果分析

世界按照从上到下，从左到右的顺序处理 actors 的行为，所以优先考虑处理顺序靠前的 actor。实际与预期结果一致。

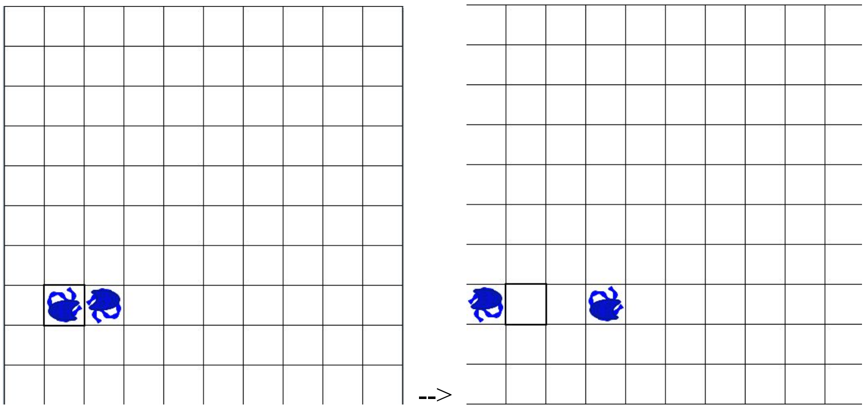
9 用例 8

测试目的：测试 Jumper 是否互相跳过相邻的 jumper。

9.1 测试用例

```
Initial:  Jumper1 Location (7, 1) Direction(EAST)
          Jumper2 Location (7,2) Direction (WEST)
Expect:   Jumper1 Location (7, 3) Direction(EAST)
          Jumper2 Location (7, 0) Direction (WEST)
```

9.2 测试结果



9.3 结果分析

两只 Jumper 相邻相遇对面时，互相跳过，实际与预期结果一致。

10 用例 9

测试目的：测试 Jumper 是否正确处理和其他 jumper 相遇问题。

10.1 测试用例

三种情况： Jumper1 在下，Jumper2 在上两格，同时向上： Jumper1 先活动往前两格，Jumper2 也有空间前进，所以两个都正常前进；

Jumper1 在下，Jumper2 在上两格，同时向下： Jumper1 先活动，向下被阻挡，于是转向，Jumper2 前进两格；

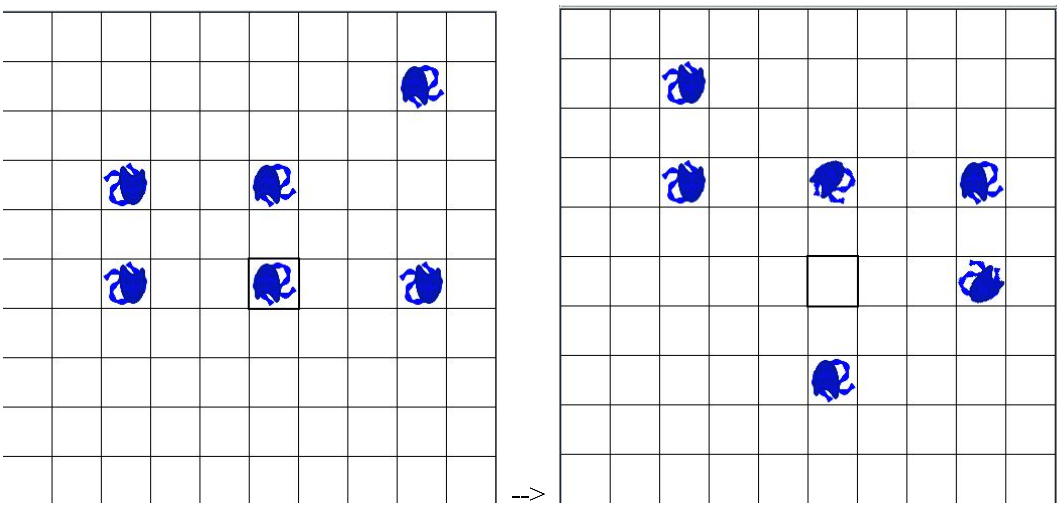
Jumper1 在上方向向下，Jumper2 在下四格方向向上，两个 Jumper 争抢中间位置，Jumper1 先活动抢到位置，Jumper2 右转。

Initial: (1) Jumper1 Location (5, 2) Direction(NORTH)
Jumper2 Location (3, 2) Direction (NORTH)
(2)Jumper1 Location (5, 5) Direction(SOUTH)
Jumper2 Location (3, 5) Direction (SOUTH)
(3) Jumper1 Location (1, 8) Direction (SOUTH)
Jumper2 Location (5, 8) Direction (NORTH)

Expect: (1) Jumper1 Location (3, 2) Direction(NORTH)
Jumper2 Location (1, 2) Direction(NORTH)
(2)Jumper1 Location (7, 5) Direction (SOUTH)

```
    Jumper2 Location (3, 5) Direction (SOUTHWEST)
(3)Jumper1 Location (3, 8) Direction (SOUTH)
    Jumper2 Location(5, 8) Direction (NORTHEAST)
```

10.2 测试结果

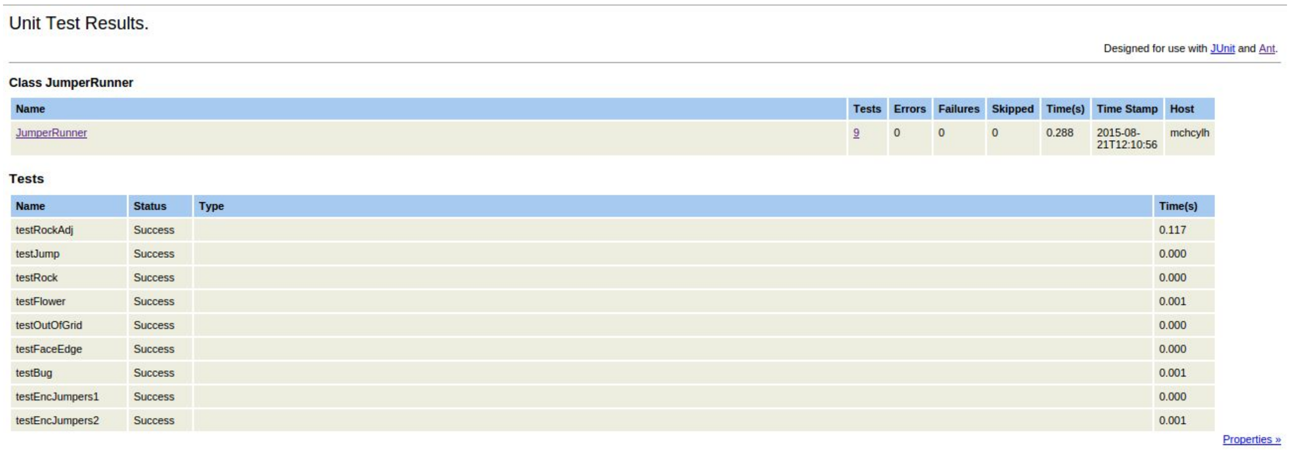


10.3 结果分析

世界按照从上到下，从左到右的顺序处理 actors 的行为，所以优先考虑处理顺序靠前的 actor。实际与预期结果一致。

最终测试结果：

测试样例全部通过



Unit Test Results.

Designed for use with JUnit and Ant.

Class JumperRunner

Name	Tests	Errors	Failures	Skipped	Time(s)	Time Stamp	Host
JumperRunner	9	0	0	0	0.288	2015-08-21T12:10:56	mchcylh

Tests

Name	Status	Type	Time(s)
testRockAdj	Success		0.117
testJump	Success		0.000
testRock	Success		0.000
testFlower	Success		0.001
testOutOfGrid	Success		0.000
testFaceEdge	Success		0.000
testBug	Success		0.001
testEncJumpers1	Success		0.000
testEncJumpers2	Success		0.001

Properties »