13331093
黄雄镖

Set 7
1. What methods are implemented in Critter?

Act, getActors, getMoveLocations, makeMove, processActors, selectMoveLocation

2.What are the five basic actions common to all critters when they act?

GetActors, getMoveLocations, makeMove, processActors, selectMoveLocation

3.Should subclasses of Critter override the getActors method? Explain.

If the subclass of critter want to change another location to get actor, it should override the getActors method.

4.Describe the way that a critter could process actors.

It remove actors except rock and critter.

5.What three methods must be invoked to make a critter move? Explain each of these methods.

GetMoveLocations, selectMoveLocation, makeMove, first, it use the GetMoveLocations to get a list of empty adjacent locations to it, than it use the selectMoveLocation method to select a location, finally, it pass the selected location to the makeMove method to move to the location.

6.Why is there no Critter constructor?

Because the critter class is inherit from the actor class. If there is no critter constructor, it will call the actor's constructor as the super class as default.

Set 8
1.Why does act cause a ChameleonCritter to act differently from a Critter even though ChameleonCritter does not override act?

Because the ChameleonCritter class overrides the processActors and makeMove methods, and the act method will call these methods, so even though ChameleonCritter does not override act, a ChameleonCritter to act differently from a Critter.

2.Why does the makeMove method of ChameleonCritter call super.makeMove?

Because it only want to change its direction toward the given location, then act like the critter, so after it set the direction, it call super.makeMove.

3.How would you make the ChameleonCritter drop flowers in its old location when it moves?

Change the makeMove method of the ChameleonCritter. After the ChameleonCritter move to a new location, drop a flower in its old location. The makeMove method like this:

```
    public void makeMove(Location loc)
    {
     Location oldLoc = getLocation();
        setDirection(getLocation().getDirectionToward(loc));
        super.makeMove(loc);
        if (!oldLoc.equals(getLocation())) {
          Flower flower = new Flower(getColor());
          flower.putSelfInGrid(getGrid(), oldLoc);
        }
    }
```

4.Why doesn't ChameleonCritter override the getActors method?

Because the ChameleonCritter also need to act like that, it isn't necessary to override it.

5.Which class contains the getLocation method?

Actor

6.How can a Critter access its own grid?

Use the getGrid method.

Set 9

1.Why doesn't CrabCritter override the processActors method?

Because the CrabCritter also need to process actors around it like that, it isn't necessary to override it.

2.Describe the process a CrabCritter uses to find and eat other

actors. Does it always eat all neighboring actors? Explain.

A CrabCritter use the getActors to get actors that is found in the locations immediately in front, to the right-front, or to the left-front of it, that use the processActors method to eat them.

3.Why is the getLocationsInDirections method used in CrabCritter?

Because when it eat or move, it can only eat or move on the location in some specify direction. So it need a method to pass a directions array and return an array of the location that the carb can eat or move into.

4.If a CrabCritter has location (3, 4) and faces south, what are the possible locations for actors that are returned by a call to the getActors method?

(4,4) (3,4) (5,4)

5.What are the similarities and differences between the movements of a CrabCritter and a Critter?

Similarities: Both of them select a random directions to move, when they move, they don't face the direction they move.

Differences: The Critter will process all actor around it, the CrabCritter only process the actor that is found in the locations immediately in front, to the right-front, or to the left-front of it. The critter can't turn when it can't move The CrabCritter can only move left or right.

6.How does a CrabCritter determine when it turns instead of moving?

If the loc.equals(getLocation()) is true, it will turn.

7.Why don't the CrabCritter objects eat each other?

Because the processActors check the actors' type, it will only remove the actors except rock and critter, the crabcritter is accritter, so the CrabCritter objects don't eat each other.