Step7

1. **What methods are implemented in Critter?**

```
   public void act();
   public  ArrayList<Actor>  getActors();
   public void processActors( ArrayList<Actor>  actors);
   public ArrayList<Location>  getMoveLocations();
   public location selectMoveLocation(ArrayList<Location>  locs);
   public void makeMove(Location  loc);
```

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

(1) Get the valid actors that are adjacent to the critter;
(2) Process on the actors that are adjacent to the critter;
(3) Get the empty Locations that are adjacent to the critter;
(4) Select one Location to move from the Locations that are adjacent to the critter;
(5) move to the Location that are selected in (4).

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

Yes, because different critters have different action on the actors adjacent them. Some critters may only need the actors that are in front of them and behind them, some may need the actors that are on the left of them and on the right of them, and others may need all actors that are adjacent to them. So we have to override the getActors method for different critters.

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

(1) Change the color of actors;
(2) Remove actors from the grid;
(3) Put actors to another locations;
(4) Eat the actor in the location that the critter is moving toward.

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

(1) getMoveLocations(); //to get the locations that are empty and adjacent to the critter;
(2) selectMoveLocation();   //select a location from the adjacent location for the critter;
(3) makeMove(Location);   //move to the new location.

6. **Why is there no Critter constructor?**

Because the Critter class has a default constructor, ant Critter has no property to initialize in
the constructor;so it only need a default constructor.

Step8

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

Because the ChameleonCritter overrides the processActors method, which will cause a
different operation on the processors.

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

Because ChameleonCritter has the same movement as the Critter, and it just needs to call the super.makeMove().

3.How would you make the ChameleonCritter drop flowers in its old location when it moves?
Change the makeMove method as following:

```
public void makeMove(Location loc)
{
    setDirection(getLocation().getDirectionToward(loc));
    Location locc = getLocation();
    Grid<Actor> gr = super.getGrid();
    if (null == gr)
       return;
    super.makeMove(loc);
    Flower flower = new Flower(getColor());
    flower.putSelfInGrid(gr, locc);

}
```

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

Because ChameleonCritter needs to get all the actors adjacent to it, just as what Critter does.

5. Which class contains the getLocation method?

The Actor class.

6. How can a Critter access its own grid?

Call getGrid method.

Step9

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

Because it takes the same action toward actors, which are provided by getActors method, as the Critter does, which is to eat the actors.

2. Describe the process a CrabCritter uses to find and eat other actors. Does it always eat all neighboring actors? Explain.

The CrabCritter call getActors method to get all the actors adjacent to it except CrabCritter and Rock, and then it call processActors method to eat the actors it has just got. An CrabCritter does’t eat all neighboring actors; it just eats the neighboring actors in front of it, left-front of it, and right-front of it.

3. Why is the getLocationsInDirections method used in CrabCritter?

Because the CrabCritter needs to find the actors in front of it, left-front of it, and right-front of it.

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

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

Similarities: Both CrabCritter and Critter randomly choose a location from the locations provided by getMoveLocation method.

Differences:A CrabCritter only moves to the right of left, when a Critter may move to any position neighboring to it. When a CrabCritter has no way to move, it will turn;but when a Critter has no way to move any more, it will turn.

6.How does a CrabCritter determine when it turns instead of moving?
   When the location selected by the selectMoveLocation method equals to the location the CrabCritter settles in now, the CrabRritter will turn; otherwise, it will move.

7.Why don't the CrabCritter objects eat each other?
   Because CrabCritter inherits the processActors method from Critter, and in the processActors method, it makes that a Critter doesn't eat another Critter.