Refresher

|  |  |
| --- | --- |
| C:\Users\Doug\Documents\a-oprfhs\12-13\AP Computer Science\GridWorld\GridWorldCode\GridWorldCode\framework\info\gridworld\actor\Bug.gifWhen a **Bug** acts, it either moves or turns. It uses a two-way, if-else logic. Bug public void act() {  if (canMove())  move();  else  turn();  } | When a **Critter** acts, it goes through a five-step sequential process. Steps one and two get and interact with other Actors in the grid. Steps three through five pick a location to move to and perform the move.  C:\Users\Doug\Documents\a-oprfhs\12-13\AP Computer Science\GridWorld\GridWorldCode\GridWorldCode\framework\info\gridworld\actor\Critter.gif Critter public void act() {  if (getGrid() == null)  return;  ArrayList<Actor> actors = getActors(); (1)  processActors(actors); (2)  ArrayList<Location> moveLocs = getMoveLocations(); (3)  Location loc = selectMoveLocation(moveLocs); (4)  makeMove(loc); (5)  }  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) {...} |

**Important!** Your job when making Critters is to extend Critter, and override one or more of the five methods that get called by act(). Do not override act()!

Hints

**ChameleonKid**: If you are tempted to override processActors() to write ChameleonKid, please reconsider. In fact I will not accept this approach because another approach is more streamlined. Is it better for processActors to receive a list of all neighbor Actors and select a random Rock or Flower from it, or receive a list that only has neighbor Rocks and Flowers? (The latter is better.)