IDEAS

Use ArrayList for the creation of Hunters and Prey

- Create objects for n objects with ArrayList <as seen in ./knowledge_base-samples/objArrTest.java>
- Enter n via UI (use slider)
- use arraylist or array

create class Actor

- int atk
- int def
- visCone()
- movement()
- isFriendly()
- attack()
- onOverlap()
- movement speed defined

HunterActor

`DERIVED FROM ACTOR`

- int hp (Do we need HP on actor Actor)
- starvation()
    - subtract x hp from hunter actor instance
    - x is determined through gui
- movePoint()
- isInGroup()
- howling()
    - scan array in 'circle' around HunterActor instance -> if hunter found set their move point
- isGroupNeeded()
    - only use objects knowledge
    - if yes do howling()
- soloHunting()
- groupHunting()
    - runs isInGroup() in background for each object
    - as soon as 1 or more hunter are onOverlap() with prey the prey is conmsidered dead
        - delete PreyActor from Stack/Array
- isAttacked() *check if the Hunter gets attack if yes invoke run()
- run()
    - look for biggest of the near hunter groups
- isPreyVisible()
    - check if Prey is in visCone()

    - invoke isGroupNeeded()
- killedPrey()
    - send out signal for to all other Hunter Instances in Group radius

PreyActor

DERIVED FROM ACTOR

- canRetaliate()
    - if prey is bigger than hunter than it can retaliate
        - killHunter()
- killHunter()
    - checks if prey.getAtk() is bigger than predator.getDef() delete Predator from Stack/Array
- isHunterVisible()
    - check if Hunter is in visCone()
    - invoke canRetaliate()
    - invoke run() [THIS DIFFRENT FROM HUNTER SINCE PREY CAN'T ESCAPE TO BIGGER GROUP]

Initilizations

- if autoGenPrey == true
    - use random Gen of stats for each prey
    - use a random position on the grid (use gridx and gridY)
    - how to auto gen ervery cycle ?
        - use thread sleep

parameters entered with gui

- int starverate -> how many cycles can a hunter survive without food
- int baseMoveSpeed -> movement speed
- int gridX and int gridY -> x and y size
- int predatorCount = input -> Initial pradator count
- boolean autoGenPrey -> prey auto generate toggle
- int simulationSpeed -> speed of the simulation (how long should a cylce last)
- int groupRadius -> Predator group radius

Gui - visual representation

- colored tile based grid
    - is dependent on the color schema selected on color drop down
- movementt speed should be shown (and set?)
- a trail is NOT required
    - a trail of movement might be a nice help for debuging
- there has to be a GUI for the user to enter parameters that influence the GUI
    - there needs to be color options for the
    - hunter and prey should be diffrentiable

- grid size is dependednt on ui entered parameter
- predator count
    - add: death stats like avg death and death rate (how many preddator die per cycle)
- avg food gained per iteration
    - add?: avg food gained by cycle
- simulation speed (slider)