Tuguldur Sosorbaram
EERDEY
Sutu0318@gmail.com
Group 11

2023.06.11

Documentation

Task

We are simulating the animals of the tundra. There are colonies of prey and predator animals. The number of animals in a colony affects the number of animals in other colonies. There are three predator species: the snowy owl, the arctic fox, and the wolf. There are three kinds of prey: the lemming, the arctic hare, and the gopher.

If the number of prey animals increases, predators can reproduce more quickly. If the number of preys is very large, most of them will wander away because they cannot find enough food. If the number of predators is large, the number of the prey decreases quicker as they are preyed upon.

Each colony has a name, a species, and the number of animals in the colony. The prey species are affected by the different predator species as follows. The number of animals in their own colony changes first, then they influence the predators.

Lemming: If they are preyed upon by a predator colony, the number of animals in their colony decreases by four times the number of animals in the predator colony. The number of animals in their colony doubles every second turn. If there are more than 200 animals in the colony, the number of animals in the colony decreases to 30.

Hare: If they are preyed upon by a predator colony, the number of animals in their colony decreases by double the number of animals in the predator colony. The number of animals in their colony grows by 50 percent (to one and a half times their previous number) every second turn. If there are more than 100 animals in the colony, the number of animals in the colony decreases to 20.

Gopher: If they are preyed upon by a predator colony, the number of animals in their colony decreases by double the number of animals in the predator colony. The number of animals in their colony doubles every fourth turn. If there are more than 200 animals in the colony, the number of animals in the colony decreases to 40.

Predators choose and attack a prey colony randomly in each turn. If there are not enough animals in the attacked colony (for example, there are not four times the number of predators in a lemming colony), the number of predators also decreases: every fourth predator out of the ones who didn't get prey perishes. Predators have offspring every eighth turn. Normally, the snow owls have 1 offspring per 4 animals, the foxes have 3 offspring per 4 animals, and the wolves have 2 offspring per 4 animals.

The program should read the colonies from a text file. The first line contains the number of prey and predator colonies separated by space. Each of the next lines contains the data of one colony separated by space: their name, their species, their starting number of animals. The species can be: o - owl, f - fox, w - wolf, l - lemming, h - hare, g - gopher.

Tuguldur Sosorbaram
EERDEY
Sutu0318@gmail.com
Group 11

Simulate the process until the number of animals in each predator colony decreases to below 4, or the number of predators doubles compared to its starting value. Print the data of each colony in each turn.

Analysis

In this task I need a class named Colony consist of 2 different subclasses Prey, Predator. Prey and Predator have 3 different inherited classes. Each one represents a different species of tundra animal.

Each colony has a name, species name, and population count. And depending on if it is predator or prey, they have different methods. Each species of prey reacts to predator in a different way.

Lemming decreases 4 times the number of predators when attacked. Also, they multiply twice themselves every 2nd turn. But when they get more than 200, they decrease to 30.

Hare and gopher decreases twice the number of predators when attacked. Also hare increases by 50 percent of themselves each 4th turn gopher increases twice every 4th turn. Hare decreases to 20 if it is more than 100, gopher decreases to 40 when it is more than 200.

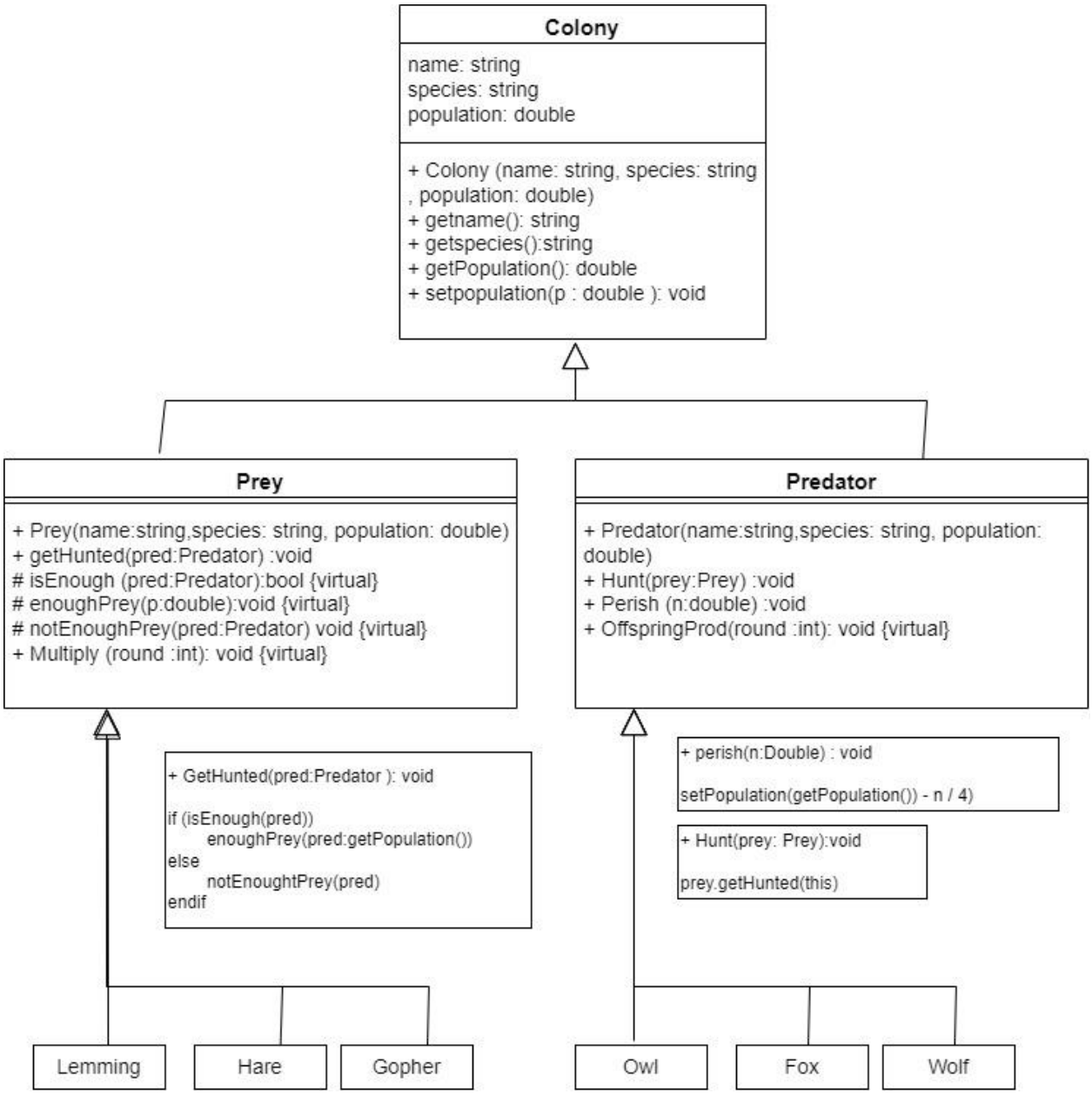
Also, when predators don't have enough prey to hunt then every 4th of the predators who didn't get prey perishes.

Plan

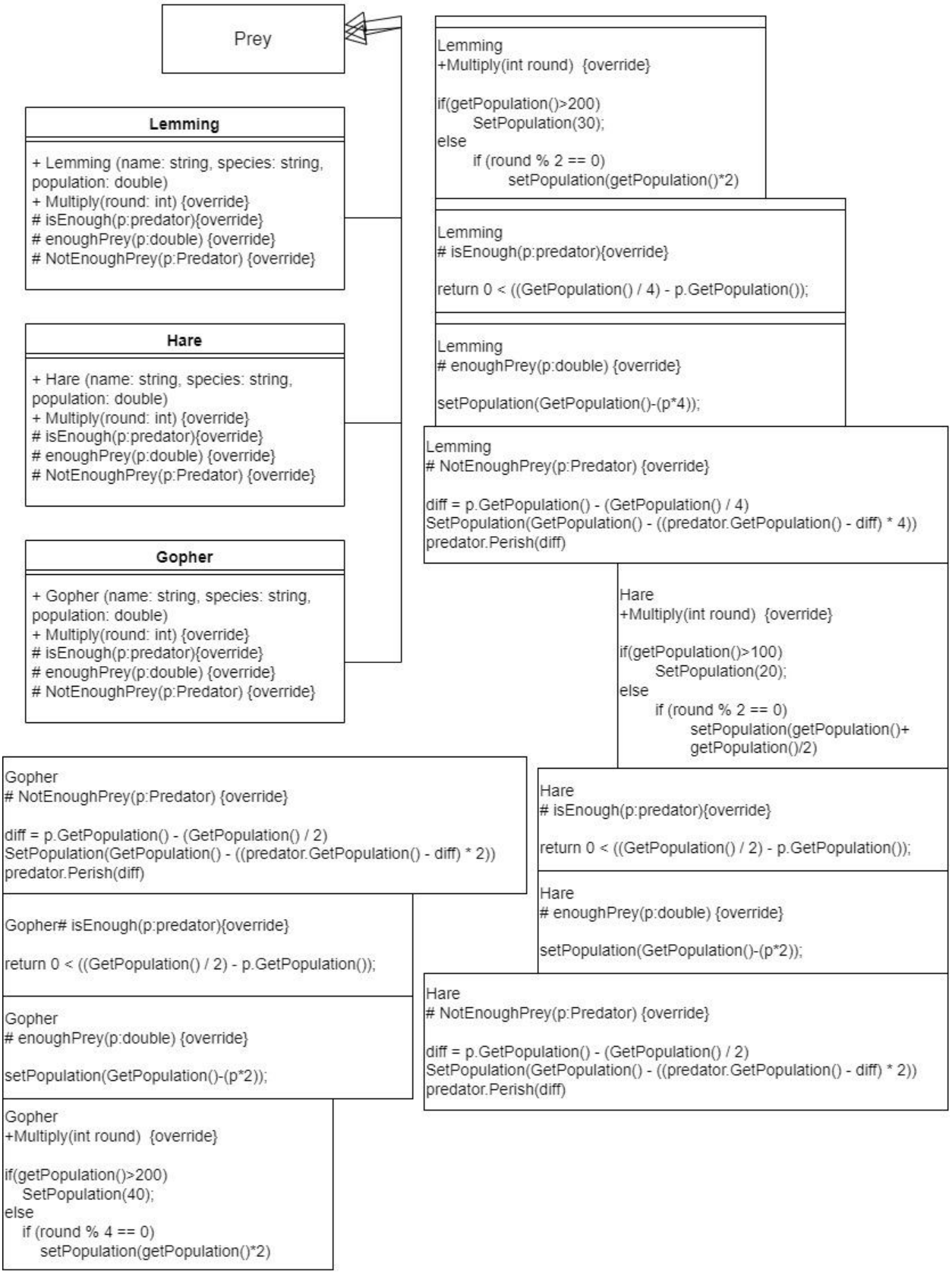
Prey and Predator subclasses of Colony do not have their unique attributes. They just inherit it from Colony. Prey has 3 inherited children which are Lemming, Hare, and Gopher. Prey has getHunted() method which uses template method. Its algorithm is divided into protected submethods and those submethods are overridden in each children of Prey. Predator have the Hunt method which passes itself as parameter of getHunted() method of the Prey. (Dependency Injection and Template Method) getHunted() check if there is enough prey for predators and if it is Predators hunt fully but if it is not then it calls Perish() method from Predator class. Perish () method takes number of predators who couldn't find prey as a parameter and reduces the population of that predator by fourth of the given parameter. Also there is offspringproduction() in predator and multiply() in prey. These methods are responsible for reproduction of each animal colonies, and they take integer as a parameter to check the turn and respond appropriately.

Tuguldur Sosorbaram
EERDEY
Sutu0318@gmail.com
Group 11

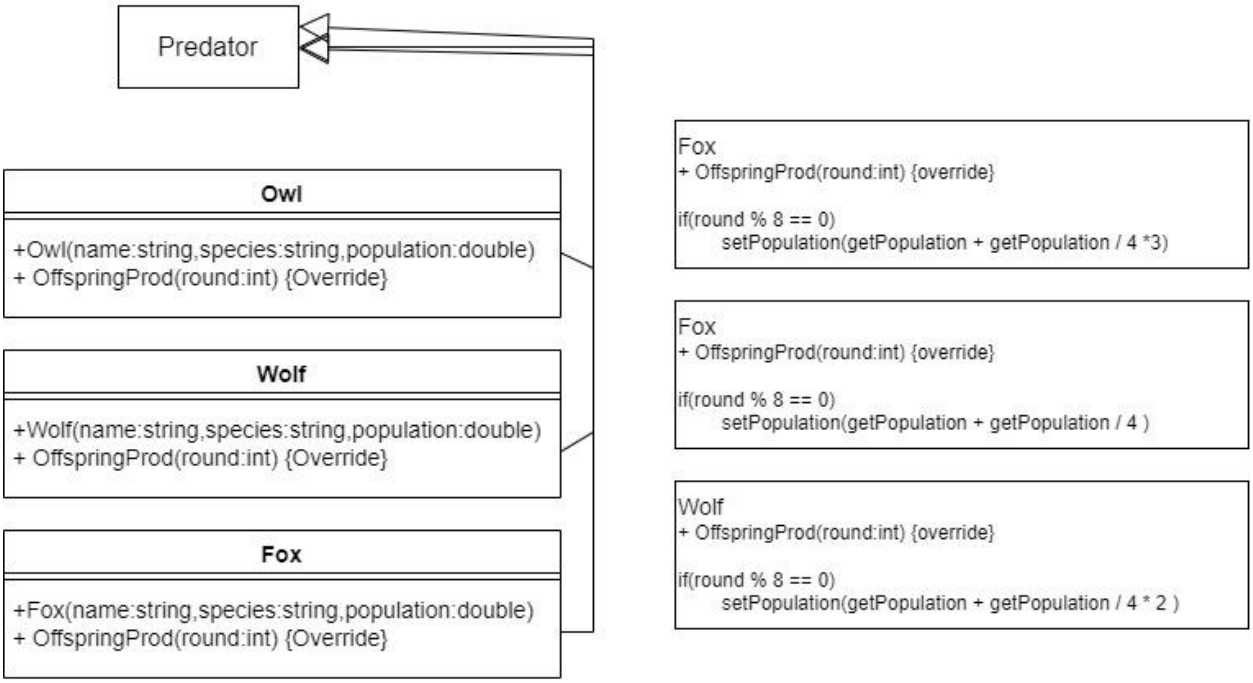
2023.06.11



Tuguldur Sosorbaram
EERDEY
Sutu0318@gmail.com
Group 11



Tuguldur Sosorbaram
EERDEY
Sutu0318@gmail.com
Group 11



Algorithm:

Tasks did not require algorithmic patterns, so instead I will show what algorithmic patterns I used in helper class Turn, which saves initial number of total number of predator in the input colonies. And it has 2 important methods, 1st one is Start() which updates round number and number of each colonies offsprings. Lastly it handles hunting of each predator with random prey.

2nd method, End(), checks if program should stop or not. It uses summations and linear search.

Firstly it goes through the list of predators and adds their population.

Analogy:

f() = predator.getPopulation

s = sum

n = predators.count()

Sum = 0
i = 0 .. predators.count()
Sum = sum + Predators[i].getPopulation

EERDEY
Sutu0318@gmail.com
Group 11

Then it checks if total number if double the initial number of predators, if it is, then we stop the simutation. Otherwise we check if each of predators are less than 4 with optimistic linear search algorithms.

Analogy:

Cond() = predators[i].getpopulation > 4

l = true, i = 0;	
l==true ∧ i<predators.Count()	
	l = predators[i].getPopulation < 4
	i = i + 1

Testing:

Test Perish and offspring production:

1. Perish when enough and not enough
    a. Wolf
    b. Fox
    c. Owl
2. Perish 0,1 and more
    a. Wolf
    b. Fox
    c. Owl
3. Offspring production
    a. Wolf
    b. Fox
    c. Owl

Test Multiply:

1. Multiply when reached limit
    a. Lemming
    b. Hare
    c. Gopher
2. Multiply before limit
    a. Lemming

EERDEY
Sutu0318@gmail.com
Group 11

   b. Hare
   c. Gopher

Test Hunting:

1. Hunt enough prey
   a. Wolf
   b. Fox
   c. Owl
2. Hunt not enough prey
   a. Wolf
   b. Fox
   c. Owl

Test getHunted:

1. getHunted when enough prey
   a. Lemming
   b. Hare
   c. Gopher

2. getHunted when not enough prey
   a. Lemming
   b. Hare
   c. Gopher

Test Cases

Perish test

1. Perish when 0:
   Wolf(x,y,0)
   Wolf.Perish(0)
   Wolf.count = 0

   Fox(x,y,0)
   Fox.Perish(0)
   Fox.count = 0

   Owl(x,y,0)
   Owl.Perish(0)
   Owl.count = 0
2. Perish more than there is (expect exception)
   Wolf.perish(1)
   Expect TooManytoPerish

EERDEY
Sutu0318@gmail.com
Group 11

Fox.perish(1)
Expect TooManytoPerish

Owl.perish(1)
Expect TooManytoPerish

3. Enough number of perish
Wolf.setpopulation(100)
Wolf.Perish(100)
Wolf.count = 75

Fox.setpopulation(100)
Fox.Perish(100)
Fox.count = 75

Owl.setpopulation(100)
Owl.Perish(100)
Owl.count = 75

Offspring test

Wolf(x,y,10)
Fox(x,y,10)
Owl(x,y,10)
Wolf.OffspringProd()
Fox.OffspringProd()
Owl.OffspringProd()

Wolf.count = 14
Fox.count = 16
Owl.count = 12

Multiply Test

1. When limit is over
Lemming(x,y,201)
Hare(x,y,101)
Gopher(x,y,201)

Lemming.Multiply()
Hare.Multiply()
Gopher.Multiply()

Lemming.count = 30

EERDEY
Sutu0318@gmail.com
Group 11

Hare.Count = 20
Gopher.count = 40

2. When limit is not reached yet
   Lemming.setPopulation(100)
   Hare.setPopulation(50)
   Gopher.setPopulation(100)

   Lemming.Multiply()
   Hare.Multiply()
   Gopher.Multiply()

   Lemming.count = 200
   Hare.count = 75
   Gopher.count = 200

Testing Hunt

1. There is enough prey
   Wolf(x,y,10)
   Fox(x,y,10)
   Owl(x,y,10)
   Lemming(x,y,100)

   Wolf.Hunt(lemming)
   Lemming.count = 60
   Wolf.count = 10

   Lemming.setPopulation(100)
   Fox.Hunt(Lemming)
   Lemming.count = 60
   Fox.Count = 10

   Lemming.setPopulation(100)
   Owl.Hunt(Lemming)
   Lemming.count = 60
   Owl.Count = 10

2. There is not enough prey
   Wolf(x,y,30)
   Fox(x,y,30)
   Owl(x,y,30)

EERDEY
Sutu0318@gmail.com
Group 11

Lemming(x,y,100)

Wolf.Hunt(lemming)
Lemming.count = 0
Wolf.count = 29

Lemming.setPopulation(100)
Fox.Hunt(Lemming)
Lemming.count = 0
Fox.Count = 29

Lemming.setPopulation(100)
Owl.Hunt(Lemming)
Lemming.count = 0
Owl.Count = 29

Test getHunted

1. When there is enough prey
   Lemming(x,y,100)
   Hare(x,y,100)
   Gopher(x,y,100)
   Wolf(x,y,10)

   Lemming.gethunter(wolf)
   Lemming.count = 60
   Wolf.count = 10

   Hare.gethunter(wolf)
   Hare.count = 80
   Wolf.count = 10

   Gopher.gethunter(wolf)
   Gopher.count = 80
   Wolf.count = 10

2. When there is not enough prey

   Lemming.setpopulation(100)
   Hare.setpopulation(100)
   Gopher.setpopulation(100)

EERDEY
Sutu0318@gmail.com
Group 11

Wolf.setpopulation(30)
Lemming.gethunter(wolf)
Lemming.count = 0
Wolf.count = 29

Wolf.setpopulation(60)
Hare.gethunter(wolf)
Hare.count = 0
Wolf.count = 58

Wolf.setpopulation(60)
Gopher.gethunter(wolf)
Gopher.count = 0
Wolf.count = 58

Tuguldur Sosorbaram
EERDEY
Sutu0318@gmail.com
Group 11