OOP Nicolas Nino Lozano: Sample Documentation, 2nd Assignment

Nicolas Nino Lozano A0T4ZR a0t4zr@inf.elte.hu Group 5	**2nd assigment/0.Task**	27th April 2024

Task

Hobby animals need several things to preserve their exhilaration. Steve has some hobby animals:

- Tarantulas,
- Hamsters
- Cats.

Every animal has a name and their exhilaration level is between 0 and 70 (0 means that the animal is dead). If their keeper is joyful, he takes care of everything to cheer up his animals, and their exhilaration level increases this way:

- Tarantulas by 1
- Hamsters by 2
- Cats by 3.

On a usual day, Steve takes care of only the cats (their exhilaration level increases by 3), so the exhilaration level of the rest decreases this way:

- Tarantulas by 2
- Hamsters by 3

On a blue day, every animal becomes a bit sadder and their exhilaration level decreases this way:

- Tarantulas by 3
- Hamsters by 5
- Cats by 7.

Steve's mood improves by one if the exhilaration level of every alive animal is at least 5.
Every data is stored in a text file. The first line contains the number of animals. Each of the following lines contain the data of one animal: one character for the type (T – Tarantula, H – Hamster, C – Cat), name of the animal (one word), and the initial level of exhilaration.
In the last line, the daily moods of Steve are enumerated by a list of characters (j – joyful, u – usual, b – blue). The file is assumed to be correct.

- ❖ List the animals of the highest exhilaration level at the end of each day.

Analysis

Steve is the owner of a certain number of animals, which consist of three types:
Tarantulas, cats, and hamsters.
Each of them has a name and an exhilaration level that can be obtained. It can be examined what happens when Steve's mood changes. His mood influences his animals, as it determines how much he cares for them, resulting in a change in the animals' exhilaration levels and in the same way but in the opposite direction, his animals change his mood as well. This can be explained in the following way:

- Tarantula:

Mood	**Exhilaration**
Joyful	+1
Usual	-2
Blue	-3

- Hamster:

Mood	**Exhilaration**
Joyful	+2
Usual	-3
Blue	-5

- Cat:

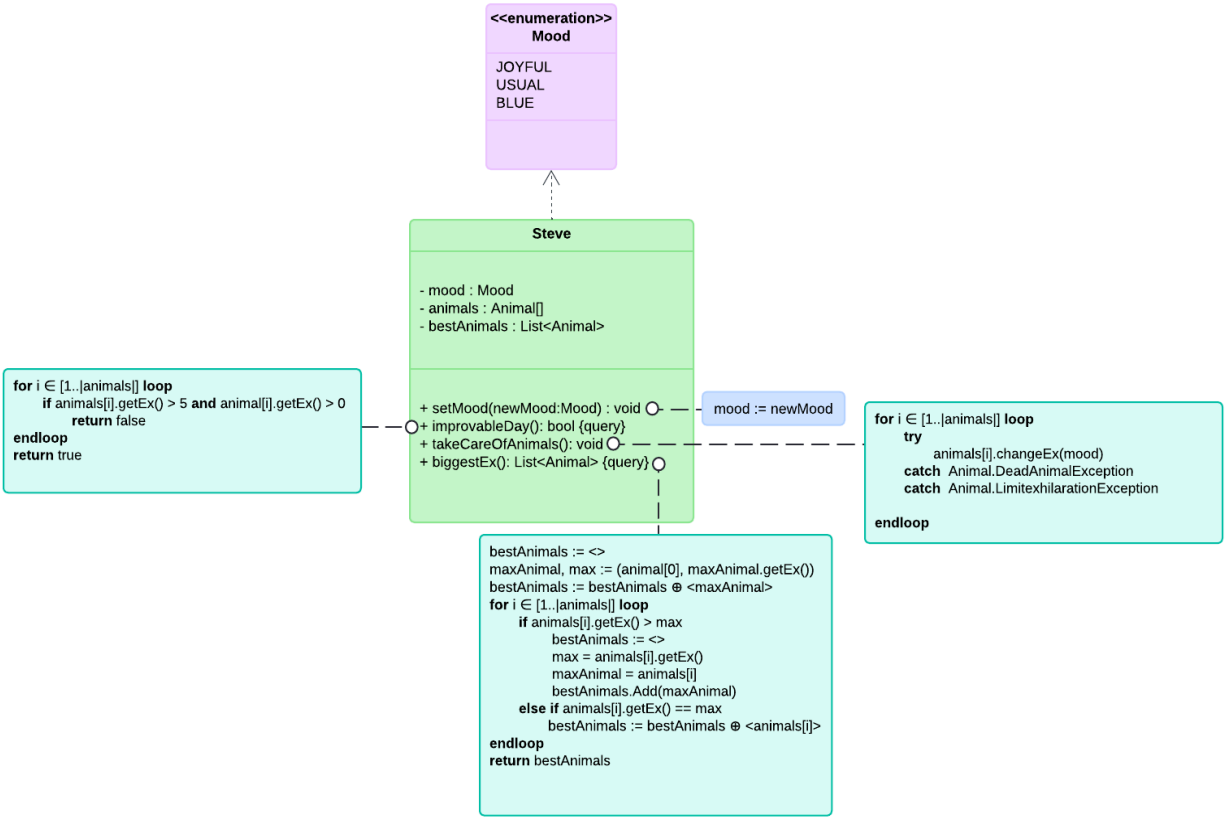
Mood	**Exhilaration**
Joyful	+7
Usual	-3
Blue	-7

- Steve:

Exhilaration	Mood	New Mood
If the exhilaration of every living animal is at least 5	Blue	Usual
If the exhilaration of every living animal is at least 5	Usual	Joyful

Plan

The method changeEx is the only one that is overridden by child classes, which uses polymorphism to adapt the method for each kind of animal's needs.



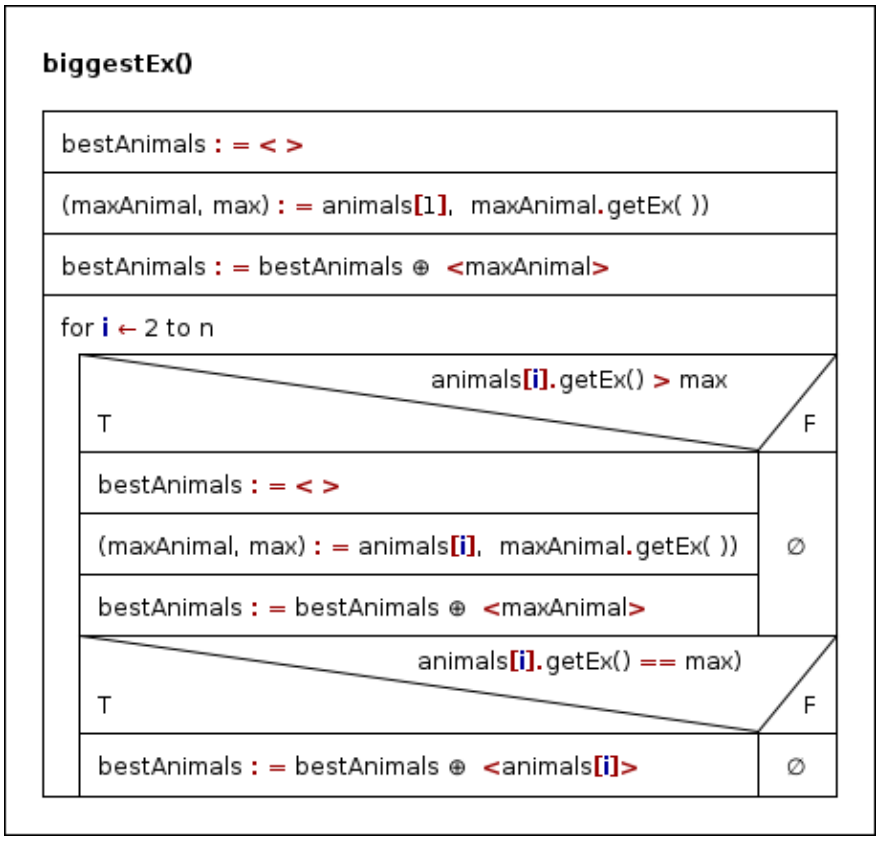
Specification

Since the program aims to obtain the animals with the biggest exhilaration each day from a file which contains all the needed information for the program to work faultlessly, a class that works as a file reader and an enumerator was created for this purpose.
Therefore, the file reader class initializes a Steve class, its mood, and its animals by reading the information from the file. So, the only two elements we are going to have in our state space are the sequence of Moods and the information related to Steve.
Also, it is important to consider that the enumerator just returns the names (string) of those animals that have the biggest exhilaration, not the list itself because this one is obtained from Steve's class from the method called biggestEx. Thus, the main program class just deals with the decision of printing multiple animals or just one single animal as the animal with the biggest exhilaration.
Hence, it is important for the sake of the understanding of how this program works, to define the specification of the biggesEx method and the Enumerator which are the backbone of this program.

BiggestEx Method:

A = animals: $Animal^n$, bestAnimals: $Animal^*$, maxAnimal: *Animal*, *max*: $\mathbb{Z}$
Pre = animals= animal'
Post = Pre $\wedge$ (bestAnimals,max = *MAX* i=1..n animals[i].getEx()) $\wedge$
bestAnimals = bestAnimals $\oplus$ maxAnimal $\wedge$
bestAnimals = $\oplus i$=1..n <animals[i]> (animal[i].getEx() = max)

This method basically finds the Maximum Animal depending on its exhilaration, then this maxAnimal is appended to bestAnimals and if there is an animal with the same exhilaration as this one, then it is appended as well. In the following chart you will find the implementation:



Analogy:

i = m+1 .. n	*i = 2 .. n*
f(i)	animals[i].getEx()
ind	*max*
max	*Animal*

Enumerator:

A = t:enor(String), st: Status, e : Mood, end: $\mathbb{L}$, current: string, steve:Steve
Pre = t= t' $\wedge$ st=st' $\wedge$ e=e'
Post = steve.setMood(e) $\wedge$ end =(st'' = abnorm) $\wedge$
¬end = $\forall$ i$\in$[1..|steve.animals|]:(current,(st,e,x)= $\oplus$e$\in$(e'',x''))
steve.biggestEx()[i].toString

The enumerator is going to basically return the existing animals with the biggest exhilaration using the summation pattern in a string.

Analogy:

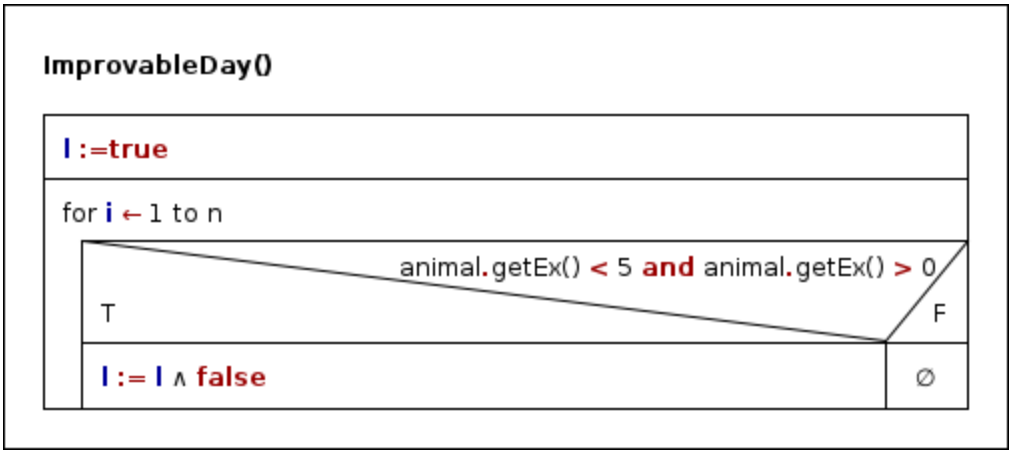
enor(E)	*enor(string)*
f(i)	steve.biggestEx()[i].toString
s	*current*
H,+,0	*current,* $\oplus$, ""

It is important also to remark that initially we read a char from the file for obtaining a mood. That's why we immediately convert this char to one of the enum constants of Mood and then once it is read we assign it to Steve's mood, also we improve Steve's mood using the method *improvableDay()* if every living animal has an exhilaration bigger or equal than 5 while reading the mood from the file . So the Read() method obtains the mood from the file and assigns it to Steve's then the Next() method uses the *takeCareOfAnimals()* method to change the Animals' exhilaration. The methods *improvableDay() and takeCareOfAnimals()* are contained in Steve Class and they will be explained below.

ImprovableDay:

A = animals: Animal^n, l:$\mathbb{L}$
Pre = animal = animal$'$
Post = l = $\bigwedge i = 1..n$ (animal.getEx() < 5 && animal.getEx() > 0) *false*

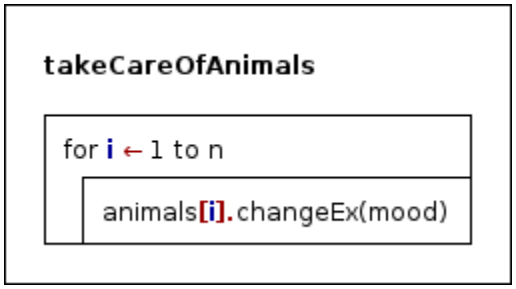
This method will return true if all of the living animals have an exhilaration bigger or equal than 5 and if not it will return false, as soon l becomes false, l is returned and the loop is broken.



takeCareOfAnimals:

A = animals: Animal^n, mood: Mood
Pre = |animal| > 0 ∧ mood = mood'
Post = animals =$\sum$ i=1..n (animals[i].changeEx(mood))

This method basically calls the method changeEx with Steve's mood as a parameter for all of the animals in the array animals.



It is important to remark that the implementation of changeEx in each kind of animal class is basically a switch block with different values as parameters, so no algorithm pattern was required for this.

Testing

Black Box test cases:

changeEx(mood: Mood)			
Test Case	***Animal and exhilaration***	***Result***	***New Exhilaration***
Change exhilaration with mood joyful	• Cat 7 • Tarantula 9 • Hamster 10	• 10 • 10 • 10	• 10 • 10 • 10
Change exhilaration with mood usual	• Cat 7 • Tarantula 7 • Hamster 8	• 10 • 5 • 5	• 10 • 5 • 5
Change exhilaration with mood blue	• Cat 10 • Tarantula 8 • Hamster 10	• 3 • 5 • 5	• 3 • 5 • 5

TestAnimalExceptions()			
Test Case	***Animal and exhilaration***	***Result***	***New Exhilaration***
Change exhilaration with mood joyful to a dead animal	Cat 0	DeadAnimal Exception()	0
Change exhilaration with mood joyful to an animal with a maximum exhilaration	Cat 70	Limitexhilaration Exception()	70

improvableDay()		
Test Case	***Animal Array***	***Result***
Check if all alive animals have an exhilaration bigger or equal than 5	[Cat 50, Tarantula 50, Hamster 0]	True
Check if all alive animals have an exhilaration bigger or equal than 5	[Cat 50, Tarantula 50, Hamster 0, Cat2 2]	False

BiggestEx()		
Test Case	***Animal Array***	***Result***
Obtain all of the animals with the biggest exhilaration	[Cat 50, Tarantula 50, Hamster 0]	<Cat 50, Tarantula 50>
Obtain all of the animals with the biggest exhilaration	[Cat 10, Tarantula 30, Hamster 0, Cat2 2]	<Tarantula 30>

takeCareOfAnimals()			
Test Case	***Steve's mood***	***Animal Array***	***New Array***
Take care of animals depending on Steve's mood	joyful	[Cat 50, Tarantula 50, Hamster 0]	[Cat 53, Tarantula 51, Hamster 0]
Take care of animals depending on Steve's mood	usual	[Cat 30, Tarantula 20, Hamster 10]	[Cat 33, Tarantula 18, Hamster 7]
Take care of animals depending on Steve's mood	blue	[Cat 8, Tarantula 4, Hamster 6]	[Cat 1, Tarantula 1 , Hamster 1]