

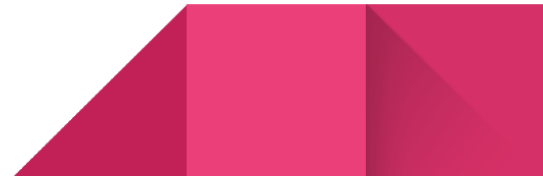


Programing Technology

Kamila Kubala
GUM6BS

1st Assignments - Race

6.10.2020



Description

There is a race for creatures, which takes place on several consecutive days. **Who wins the race? (In other words, which creature can go farthest and remain live?)** At the beginning, each creature has an amount of water, and a distance of 0 from the start. There are three different kind days could occur: sunny, cloudy, rainy. The movement and the water level of a creature are affected by the type of the day and the creature. At first, a creature changes its water level according to the day, and if it is still alive, it moves. A creature dies if it runs out of water (water level drops to 0 or below). A dead creature doesn't move... Properties of creatures: name of the creature (string), water level (integer), maximum water level (integer), living (boolean), distance (integer).

The types of creatures on the race are: sandrunner, sponge, walker.

The following table contains the properties of the creatures.

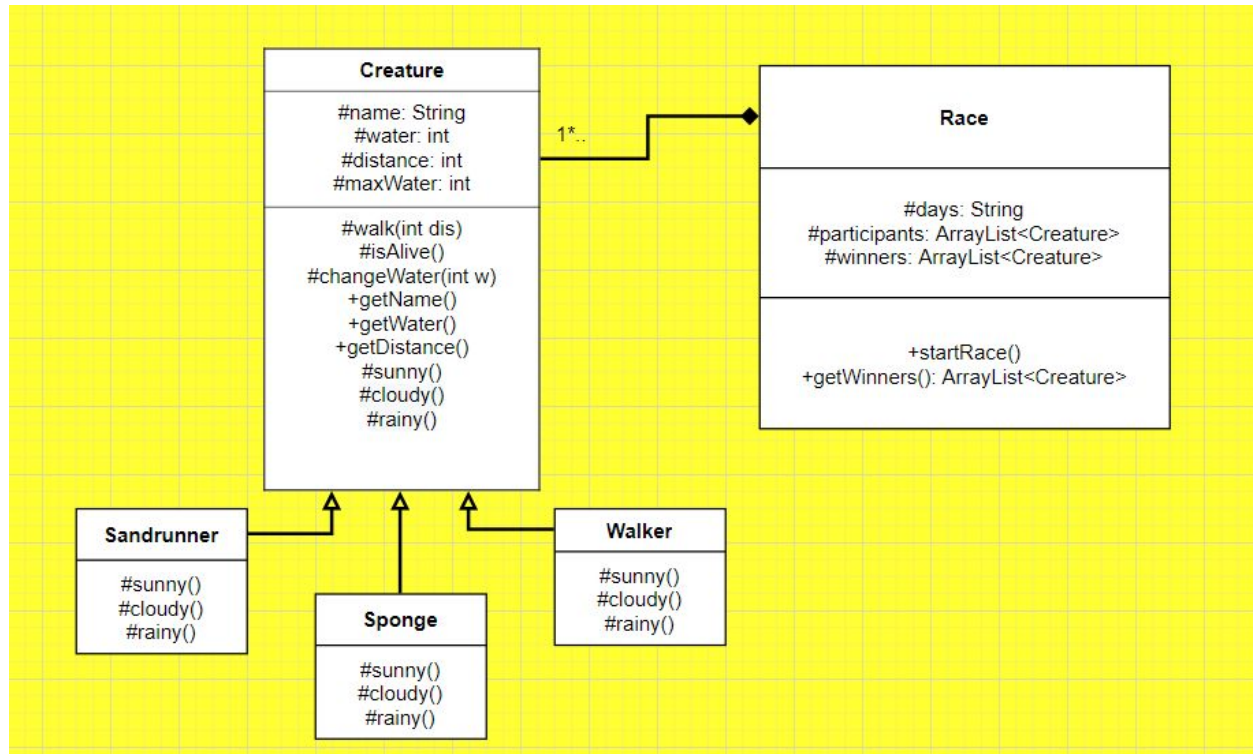
	water change			distance			max. water
	sunny	cloudy	rainy	sunny	cloudy	rainy	
sandrunner	-1	0	3	3	1	0	8
sponge	-4	-1	6	0	1	3	20
walker	-2	-1	3	1	2	1	12

Creatures cannot have water more than their maximum water level. Read the data of the race from a text file. The first line of the file contains the number of competitors (lets say N). Each of the following N lines contains a competitor: name, type, initial water level. The properties are separated by spaces; and the type is represented with one character: r - sandrunner, s - sponge, w - walker.

The last line of the file contains the type of the days on the race: s - sunny, c - cloudy, r - rainy.

The program should ask for the name of the file, and it has to print out the name of the winner (we can assume that the file is existing and its format is valid).

Class Diagram



Classes

1. **Creature:**

Attributes:

- **name**
- **distance**
- **water**
- **maxWater**

Methods:

- **Initializer** - The initializer sets the given name, maxWater, water. Distance is setting to 0.
- **walk(int dis)** - increases the distance by the given amount **dis** if the character is alive.
- **isAlive()** - checks whether the character is alive, i.e. whether the water level is greater than 0
- **changeWater(int w)** - changes the water level, ensuring that it does not exceed the maximum value.
- **getName()** - returns the name of the character.
- **getWater()** - returns the name of the character.
- **getDistance()** - returns the name of the character.

Subclasses:

1. Sponge

Methods:

- **Initializer Sponge(String name, int water)** - invokes the super class initializer with parameters name, water and as a maxWater use 20.
- **Sunny()** - if the character is alive, it reduces the water level by 4.
- **Cloudy()** - if the character is alive, it reduces the water level by 1 and the distance increases by 1.
- **Rainy()** - if the character is alive, it increases the water level by 6 and the distance increases by 3.

2. Walker

- **Initializer Walker(String name, int water)** - invokes the super class initializer with parameters name, water and as a maxWater use 12.
- **Sunny()** - if the character is alive, it reduces the water level by 2 and the distance increases by 1.
- **Cloudy()** - if the character is alive, it reduces the water level by 1 and the distance increases by 2.
- **Rainy()** - if the character is alive, it increases the water level by 3 and the distance increases by 1.

3. Sandrunner

- **Initializer Sandrunner(String name, int water)** - invokes the super class initializer with parameters name, water and as a maxWater use 8.
- **Sunny()** - if the character is alive, it reduces the water level by 1 and the distance increases by 3.
- **Cloudy()** - if the character is alive, it increases distance by 1.
- **Rainy()** - if the character is alive, it increases the water level by 3.

2. Race

Attributes:

- **days**
- **participants**
- **winners**

Methods:

- **startRace()** - the method simulates the course of a race.
- **getWinners()** - returns the list of winners of the race.

3. Database

Attributes:

- **creatures_list**
- **days**

Methods:

- **read(String filename)** - reads the data from the file and, if the file is correct, creates an appropriate list of creatures taking part in the race. Finally, it writes the set of days into the variable days.
- **getDays()** - returns set of **days**.
- **getCreatureList()** - returns **creatures_list**.
- **clear()** - clearing **creatures_list**.

TESTS

1. data.txt - all are winners

expected output:

wanderer is alive and his distance is: 12

walk is alive and his distance is: 12

slider is alive and his distance is: 12

sneaky is alive and his distance is: 12

2. data2.txt - only 2 winners

expected output:

wanderer is alive and his distance is: 24

luck is alive and his distance is: 24

3. data3.txt - they are all dead, no winners

4. data4.txt - all dead from the beginning, no winners

5. data5.txt - only 1 winners

expected output:

mike is alive and his distance is: 18

6. data6.txt - file doesn't exist

expected output:

"File not found"