Class Platypus (Documentation)

Author: Nick Robaqidze

mob: +995 598 19 07 03

Email: nika42568@gmail.com

Libraries and their usage:

* iostream - For printing data on screen.
* iomanip - To display the data in a beautiful format.
* random - To select random numbers. To imitate the assumption.

When creating a class by default the it's variables are assigned values by this way program does not automatically enter an invalid value (from buffer):

* float weight (kg) A number taken at random from the interval [0,1; 9];
* short age (month) A number taken at random from the interval [0, 11];
* char name A randomly selected letter, including the letters A to Z;
* char gender A randomly selected letter M (male) or F (female) with the same assumption;
* bool alive true (live);
* bool mutant false (No mutant);

Private functions of the class:

* checkWeight - The function returns nothing and you only call inside the class. The function checks the chances of Platypus death and outbreaks. The chance of death is equal to 10 times the weight of a duckling, i.e. a 5 kg duckling has a 50% chance of dying. If the platypus weighs 10 or more kilograms it dies (alive = false). The mutation assumption is 2%, or 1/50 is calculated by the following principle:
* A random number is taken from the interval [1,50]. If this number is equal to 7 (there is no reason to choose a number) the platypus mutates (mutant = true). (The assumption that [1,50] taken is equal to 7 is 1/50, or 2%).

The function is called in the parametric and non-parametric age\_me function.

Public access functions of the class:

* A nonparametric constructor that creates a dead (alive = false) platypus.

Text

Description automatically generated



* A parametric constructor that takes as a parameter the sex, weight, age and name of the platypus. By silence the Platypus is alive and not a mutant.

A picture containing diagram

Description automatically generated



* print - constant because it does not change the value of variables in a class, a function that displays information about Platypus in a beautiful and easy-to-understand form and format.

Text

Description automatically generated

* The function is overloaded, it is set as a parameter for printing on screen, output stream (ostream), as well as the "<<" operator, for easy calling of the function.

Text

Description automatically generated

* age\_me - a function that changes the age of the Platypus and is overloaded. It is initially checked whether the Platypus is alive. The checkWeight function will be called upon completion. In a nonparametric function, the number n is randomly selected from the interval [1, 11), then the age of the platypus increases by n months.

Text

Description automatically generated

If the Platypus is dead a message appears on the screen. (red):

Text

Description automatically generatedThe parameter age\_me function performs the setter function. The function assigns a modulus (age = abs(age)) of the parameter passed to the Platypus because the age cannot be negative.

Text

Description automatically generated

* + The screen shows the result of age change (yellow). Coincidentally, the Platypus died (assumption ½ due to weight) and was released (assumption 1/50). When this happens, a message appears (red) (checkWeight working result).

The class has three getter functions that are called in the fight function:

* + getWeight –  return weight of Platypus.
  + getName - return name of Platypus.
  + getAlive - return status of alive of Platypus(Is it a live or not).

A screenshot of a computer

Description automatically generated with medium confidence

* fight - A function that takes another Platypus as a parameter. It mimics the quarrel of the Platypus. The Platypus on which the fight function is called is calling Platypus. At fist is checking the memory address of both Platypus because the platypus cann't quarrel with itself. Also, whether the calling\_platypus Platypus is alive, then the life of the Platypus who was summoned to the quarrel. Next, (calling\_platypus) the weight of the platypus is multiplied by the weight of second. A random number from the interval [1,100). If the ratio of forces is greater than this randomly selected number calling\_platypus - wins, otherwise the winner is second. The result of the fight is displayed on the screen.

Text

Description automatically generated

The running program shows both cases when calling\_platypus wins and the second platypus wins.

Text

Description automatically generated

The screen shows a case when at beginning of fight one of the platypus is dead.

A screenshot of a computer

Description automatically generated with low confidence

Also, the case where the Platypus decides to fight with himself.

* eat - a function that increases the weight of the platypus. The function is overloaded. Initially, it is checked whether the Platypus is alive. In the nonparametric function, the number n is randomly selected from the interval [0,1, 5), then the weight of the duck card increases by n% of the current weight of the platypus. As soon as you gain weight, a message appears on the screen showing how much the duck weight has increased by % (Yellow).

Text

Description automatically generated

* eat - function also takes weight as a parameter, as much as platypus weight should be increased, but this parameter should not be mistaken for the following interval: this parameter should be not less than 0.1% of platypus weight and more than 5% of its weight.Text

  Description automatically generated

Graphical user interface, text, application

Description automatically generated

The picture shows a case where a user tries to eat a dead platypus.

* hatch - The function mimics the hatching of a platipus. Its settings have silent values, it means, if the user does not enter the name and / or gender of the newly released Platypus, it will be saved automatically. Also, the newly hatched platypus is alive (alive = true), age (age = 0) 0 months and it is not mutant (mutant = false).

A picture containing text

Description automatically generated