**Test**

In this exercise you will be asked to develop a small safari field, involving four types of animals: Lion, Tiger, Wolf and Deer. The field board contains 100 cells (10x10) and is populated with one of these animals in each cell.

Once the board is randomly populated with these types of animals, a specific cell is randomly picked. Your goal is to calculate how many cells can be conquered by the animal populating the chosen cell. This animal tries to conquer all the cells adjacent to it, and if it succeeds - all the cells adjacent to these cells, etc, until it reaches either the boarder of the board or an animal that it can't defeat or animal of its own type. Adjacent cells are below, above, beside and diagonal in reference to a given cell:

|  |  |  |
| --- | --- | --- |
|  | x |  |
|  |  |  |

Following are the relations between all animals:

1. Lion defeats Tigers, Wolves and Deer
2. Tiger defeats Wolves and Deer
3. Wolf defeats Deer
4. Deer doesn't defeat anyone

Following are the methods required for this exercise:

1. Creation of classes representing the animals above, using hierarchy/polymorphism for efficient coding.
2. Random population of the board with animals.
3. Output of the board to the screen as follows (see example in next page)
4. Random pick of a cell in the board and the animal populating it.
5. Determine which continuous cells can be conquered by the chosen animal. The method here must be recursive. Afterwards print the board again, where conquered cells are represented by "-".

**Note: You are not only tested on the final result, but on the way you've gotten there. Make sure your solution is elegant and efficient, written in clear and clean code. Comment your classes and functions, as well as main logical junctions in the code.**

Example:

Board at start:

T L L W T L W L D L

T W T D D D D L L T

W L T D D W T T D L

W D T D W D L D T L

W L D W D W W W D W

D D D W T W W L W W

W L W D T W L L D D

T W D W W L W W W T

D W T T T T W T T D

L L D T L D W D T W

Chosen cell: [10,9]: T

Board after conquer:

T L L - T L - L D L

T - T - - - - L L T

- L T - - - T T - L

- - T - - - L - T L

- L - - - - - - - -

- - - - T - - L - -

- L - - T - L L - -

T - - - - L - - - T

- - T T T T - T T -

L L - T L - - - - -