TOR 6th February 2019

Python Hand ins

There are 4 compulsory hand ins which also will useful for the exam.

All hand ins are individually.

All assignments must be uploaded to Moodle as a single .pdf file.

Please note that links will not be approved.

The hand ins must contain :

Your name

A short description of your solution

Necessary parts of Python code

Selected screen dumps demonstrating the functionality of the solution

1st Python Hand in : Pirates of the Caribbean



Remember to test the program carefully !

A bunch of pirates had found a treasure.

They decided that one of them should have the whole treasure.

Therefore they placed themselves in a circle.

They chose a number and starting at a random pirate they counted the pirates clockwise.

The pirate with the chosen number should walk the plank.

The counting started again with the next pirate in sequence.

The final pirate got the treasure.

1. Make a class Pirate. Each pirate should have a name and a pointer to another pirate.

2. Make pirate objects.

Each pirate should point to the pirate following him in the circle. (It could be useful to start with a dummy pirate).

When all pirates have been added the circle should be closed by making the last pirate point to the first.

3. When the circle is closed choose the initial pirate and the counting number (fixed or random values).

4. Find and print the pirates walking the plank one by one and to find and print the winner. (How does a pirate know when he or she is the winner ? – in Python, not in real life).

2nd Python Hand in : Graphics of foxes and rabbits

Remember to test the program carefully !

Make a program that can simulate populations of foxes (predators) and rabbits (pray).

The populations can be calculated one time step at a time :

Rabbit(tn+1) = Rabbit(tn) \* (1 + a – b \* Rabbit(tn) – c \* Fox(tn))

Fox(tn+1) = Fox(tn) \* (1 – d + e \* Rabbit(tn) – f \* Fox(tn)))

Try a = 0.1 b = 0.00002 c = 0.01 d = 0.01 e = 0.00002 f = 0.0001

(b and f represent damping)

Initial amount of rabbits = Rabbit(t0) = 500

Initial amount of foxes = Fox(t0) = 10

Time from 0 to 400

Make a graph showing how the populations changes in time.

What are the steady state values ?

Try also

Initial amount of rabbits = Rabbit(t0) = 600

Initial amount of foxes = Fox(t0) = 16

You can experiment with the parameters and initial populations.

Remember that if a population at a time is 0 (or negative) the species is extinct.

3rd Python Hand in : Website with database

Remember to test the program carefully !

Make a database with a “Car” table.

The car should at least have an ID, Model and MaxSpeed (like the one from Arnauld Weil’s book).

Make a website that can make CRUD for the table.

The website should also show the fastest car in the table.

4th Python Hand in : The game of Nim

Remember to test the program carefully !

Make a program that can play the game of Nim with you using an optimal strategy.

Modified description from Wikipedia :

*The normal game is between two players and played with a number of heaps of any number of objects. The two players alternate take any number of objects from any single one of the heaps. The goal is to be the last to take an object.*

*The following example is a game played between* [*Bob and Alice*](https://en.wikipedia.org/wiki/Bob_and_Alice) *who start with heaps of three, four and five objects.*

Sizes of heaps Moves

A B C

3 4 5 Bob takes 2 from A

1 4 5 Alice takes 3 from C

1 4 2 Bob takes 1 from B

1 3 2 Alice takes 1 from B

1 2 2 Bob takes entire A heap

0 2 2 Alice takes 1 from B

0 1 2 Bob takes 1 from

0 1 1 Alice takes 1 from B

0 0 1 Bob takes entire C heap and wins.