

Praktische Programmieraufgaben

Tag 2

Aufgabenteil 1

Bearbeitet diese Aufgaben in einer Zweiergruppe. Nachdem er sie gelöst habt tauscht euren Code mit der Nachbarsgruppe aus und versucht deren Code nachzuvollziehen. Tauscht euch danach über eure jeweiligen Implementierungen aus und gebt euch Feedback, auch bezüglich der Lesbarkeit und Verständlichkeit eures Codes.

1. Die Fibonacci-Folge ist definiert als

$$a_n = a_{n-1} + a_{n-2} \quad \text{for } n > 2 \quad \text{with } a_1 = 1, a_2 = 1,$$

also als $\{1, 1, 2, 3, 5, 8, 13, 21, 34, 55\}$. Schreibt eine Funktion, welche die n -te Fibonacci Zahl berechnet. Für das Argument 9 sollte also 34 ausgegeben werden, für 10 dann 55, etc.

2. Ein klassisches Spiel in der Spieltheorie ist das *Gefangenendilemma*. In diesem Spiel gibt es zwei Spieler, S_1 und S_2 , beide haben die Wahl zwischen zwei Strategie, **Defektion** oder **Kooperation**. Der Payoff jedes Spielers, Π_i , ist abhängig von der eigenen Entscheidung und der Entscheidung des anderen Spielers. Implementieren Sie das Spiel in Python. Ihre Implementierung sollte eine Methode oder Funktion enthalten, der man als Argumente die Strategien der beiden Spieler gibt und die als Output deren jeweiligen Payoffs ausgibt wenn der Payoff nach folgender Tabelle berechnet wird:

Strategie S_1	Strategie S_2	Π_1	Π_2
Kooperation	Kooperation	4	4
Kooperation	Defektion	0	6
Defektion	Kooperation	6	0
Defektion	Defektion	2	2

Aufgabenteil 2 (Gruppe A)

Bearbeitet diese Aufgabe in Teams von ca. 4 Leuten. Versucht eure Lösung möglichst klar darzustellen und so zu kommentieren, dass für andere klar wird wie das Modell funktioniert. Nach ca. 45 Minuten tauscht ihr den Code mit einem Vertreter der Gruppe B. Versucht deren Code nachzuvollziehen und zu verstehen, was das zugrundeliegende Problem war. Überlegt euch konstruktives Feedback, sowohl zur Implementierung als auch der Art der Darstellung und Verständlichkeit des Codes.

3. Betrachten Sie folgenden hypothetisches Zielsystem:

- Es gibt ein Ökosystem mit Hasen und Katzen.
- Die Hasenpopulation wächst jeden Zeitschritt um 10%.
- Die Katzen fressen Hasen.
- Pro Katze wird pro 2000 Hasen ein Hase pro Katze gefressen.
- Die Katzenpopulation wächst pro gefressenem Hasen um 10%.
- Beide Tierarten werden zudem von Autos bedroht: mit einer Wahrscheinlichkeit von 5% wird jedes Tier von einem Auto überfahren
- Schreiben Sie ein Python Programm das dieses Ökosystem simuliert. Dabei muss es sich nicht notwendigerweise um ein agentenbasiertes Modell handeln.
- Die Beziehungen sind in der folgenden Tabelle zusammengefasst:

	Hasen	Katzen
Variable	H_t	K_t
Hasenwachstum	0.1	
Interaktion	$-0.0005H_tK_t$	$0.1 \times 0.0005H_tK_t$
Autos	$-0.05R_t$	$-0.05K_t$
Anfangspopulation	400	50

Aufgabenteil 2

ufgabenteil 3 *Die folgende Aufgabe sollte innerhalb eurer Gruppe bearbeitet werden. Nach ca. 45 Minuten entwickeln wir im Plenum eine gemeinsame Lösung.*

4. Implementiert das Spiel Schere-Stein-Papier als agentenbasiertes Modell in Python. Die Spieler sollten ihre eigene Klasse bekommen. Am Ende sollte es eine Klasse `Simulation` und eine Klasse `Agent` mit den jeweils notwendigen Methoden geben. Falls zeitlich möglich sollte die Klasse `Simulation` auch über eine Methode zur Visualisierung der Ergebnisse erhalten.