6COM2012 Practical Assignment

Semester A 2024/2025

There are 35 marks to achieve.

Submission requirements

You may discuss your general ideas and thoughts with peers but the work handed in must be distinctly yours and your own. The following sets of documents must be submitted through canvas as individual files, not a directory.
(If you have many Python files please merge this down into 3 or less.) Please ensure that every instruction given in the text is addressed, as this will impact your score. Comment your code thoroughly, so the assessor can easily identify where the required features are implemented. In the end, you need to submit the following items:

(a) Your code that runs the different simulations

(b) The csv-files for each simulation

(c) A single report to support the simulations

Task 1: Implementation of a simplified Sugarscape Model [10 marks]

Implement a simplified Sugarscape model with the following parameters:

The world is a 20 by 20 grid-world, with wrap-around borders. Each location in the grid can be identified by 2 coordinates, going from 0 to 19. Each location has two variables. One is called **capacity**, and it is set to the sum of its coordinates at the beginning of the simulation. The other variable is called **sugar**, and it is set to the same value as the location's **capacity** at the beginning of the simulation. At the beginning of the simulation there are 20 agents, each located in a random location. No two agents can be in the same location. Each agent has an internal variable called **energy**. It is set to 10 at the beginning of the simulation. Each agent has also an internal variable called **sight**, set to the value 3.

The simulation runs in turns, with each turn consisting of the following phases in this order:

1. sugar growth phase
2. agent movement phase
3. consumption phase

In the sugar growth phase each of the locations has 1 added added to its current **sugar** value, but it cannot exceed its **capacity**. If adding 1 would take it to a value higher than the capacity, the sugar value would be set to **capacity** instead.

In the agent movement phase each agent, in random order, performs the following action. First, the agent observes the current **sugar** value for all locations visible to it, and determines the location with the highest value. In case of a tie it chooses one randomly. The agent can see all locations up to its **sight** range in each cardinal direction. So, in case of sight = 3, it can see the three locations directly above, below, left and right of it, and the location it is on. Overall, an agent with sight 3 can see 13 locations. It will not choose a location with another agent in it. It then moves to the chosen location and consumes all sugar there. This means it increases its own **energy** by the **sugar** value of the new location, and sets the **sugar** value of the location to 0.

In the consumption phase, each agent consumes 1 unit of energy, i.e. each agent's internal **energy** value should be decreased by one. Then check if the current energy is equal or lower than 0, in which case the agent dies and is removed from the simulation.

It should be possible to run this simulation for 500 turns. Test the simulation to make sure it runs and that it acts as specified. Submit the code for this simulation to the assignment as *sugarscape1*. The simulation should have no other features implemented besides the ones specified here, apart from code needed to test it, and to collect the data for task 2.

Task 2: Data Analysis of Simple Sugarscape [5 marks]

Run the simulation from Task 1 for 500 turns and collect the following data in a CSV file.

1. The sum of the **energy** of all living agents for each turn.
2. The position of all agents at turn 1, 50 and 500.
3. The amount of sugar at each location in turn 1, 50 and 500.

The canvas page has some code detailing how to write a CSV file.

In your supporting documentation start a section with the title “Simple Sugarscape Analysis”, which will be part of yoru submission package. Add the following items to this section.

1. a graph plotting the sum of all energy in living agents for each turn. Make sure to label your axes.

2. text discussing what we learn about the simulation from graph one. Discuss possible explanations for this.

3. a visualization showing the agent positions in turn 1, 50 and 500, and the sugar levels in each location for the same turns.

4. text discussing what we see in the second visualization. What do we learn about the simulation, and what explanations could create this behaviour?

Submit both the CSV files and the report for this assignment.

Task 3: Evolution in Sugarscape [10 marks]

Modify the simulation from Task 1 in the following way.

During the initialization of agents, instead of setting the sight variable to 3, set it to a random integer value between 2 and 5. Change the visible locations during the movement phase accordingly.

During the energy consumption phase add another behaviour, after checking if agents die. For each agent, if their energy is higher than 20 have them procreate. This means, that if they have an empty location next to them, they will create a new agent in that location. The agent will have the same internal variable for sight as their parent. Both parent and child will have their energy set to half the energy of the parent, i.e. they split the energy evenly. Then, perform a mutation. Generate a random integer number between 0 and 10. If the value is 0, lower their sight value by 1, if it is a 1, raise it by 1, in all other cases there is no mutation. Only perform this action for the child, and only once when it is created.

Run this simulation for 500 steps and capture data about the number of agents, and the distribution of sight variables across the population. Create a section in your report titled “Evolution in Sugarscape”. Add at least one appropriate graph that shows something relevant to how the sight variable evolves in the simulation. Discuss what we are seeing and offer possible explanations. Submit the code (as *sugarscape2*), the report and the CSV file as part of the assignment.

Task 4: Empowerment in Sugarscape [10 marks]

Recall the Empowerment measure from the lectures. You can probably approximate it by counting how many locations an agent could move to in the next turn.

Design a modification for the simulation so that some agents take empowerment into account when making their movement decisions. Consider a trade-off between the sugar they are collecting and their possible, future empowerment. Design it so that there is a variable in the agent, something like **UsesEmpowerment**, that determines if the agent uses the simple, or the empowerment-enhanced behaviour. Make this a variable that gets inherited from parent to child.

In a new section, titled "Empowerment in the Sugarscape" in your report, discuss how you plan to modify the simulation in detail. Predict how it will affect the simulation. Then run the simulation, and gather appropriate data to support your hypothesis, or disprove it. You might have to add new data collection routines.

Plot at least some of the data, and discuss it in relation to your hypothesis.

Submit the code (titled *sugarscape3*), the CSV files, and the report, as part of your assignment.

Submission checklist

Python files that show all your simulations
(If you have many Python files please merge this down into 3 or less.):

- Code for the simplified Sugarscape, called *sugarscape1*
- Code for the Evoluation in Sugarscape, called *sugarscape2*
- Code for the Empowerment and Sugarscape, called *sugarscape3*

csv-files for each simulation (you can create a zip archive)

Report with the following sections, titled (as pdf):

- Simple Sugarscape Analysis
- Evolution in Sugarscape
- Empowerment in the Sugarscape