



MEng and MMath Degree Examinations 2022–23
DEPARTMENT OF COMPUTER SCIENCE

Evolutionary & Adaptive Computing
Reassessment

Open Individual Assessment

Issued: 12 noon on 8 June 2023

Submission due: 12 noon on 28 June 2023

Feedback and marks due: 26 July 2023

Assignment Project Exam Help

All students should submit their answers through the electronic submission system: <http://www.cs.york.ac.uk/student/assessment/submit/> by 12 noon on 28 June 2023. An assessment that has been submitted after this deadline will be marked initially as if it had been handed in on time, but the Board of Examiners will normally apply a lateness penalty.

Your attention is drawn to the section about Academic Misconduct in your Departmental Handbook: <https://www.cs.york.ac.uk/student/handbook/>.

Any queries on this assessment should be addressed by email to Simon O'Keefe and Dimitar Kazakov at semok100@york.ac.uk or dlk2@york.ac.uk. Answers that apply to all students will be posted on the VLE.

Your exam number should be at the top of your Jupyter or Google Colab notebook and the corresponding PDF printout, at the start of the Info section of the NetLogo file, and at the top of your answer to Part 3. You should not be otherwise identified anywhere on your submission.

Rubric

Your submission should be a single zip file named after your exam number, `Yxxxxxxx.zip`, which contains:

- For Part 1 of this assessment
 - a single Jupyter or Colab notebook `EVAC1.ipynb` combining code and explanations
 - a PDF `EVAC1.pdf` of the state of the same notebook after all of its code has been executed.
- For Part 2 of this assessment
 - a single NetLogo file `EVAC2.nlogo` , possibly accompanied by additional image and data files.
- For Part 3 of this assessment
 - a single PDF file `EVAC3.pdf`.

If there are discrepancies between the contents and output of the Jupyter/Colab notebook and what is shown in the PDF, the former will be used for marking.

Make sure you do not use archive formats other than zip. **If your submission is not a single zip archive, you will be penalised.** Your code should assume any auxiliary files that you may generate are in the same folder as the notebook from which they are accessed.

Part 1: Evolution of a controller

1 The Problem: Evolve a Player for the Video Game Snake

Snake is a classic video game where the player controls a snake moving around a grid. In each time-step the snake moves forward in the direction it is facing. The aim of the game is for the snake to collect apples that appear around the grid. Each apple increases the score by one. The game is over if the snake's head collides with its own body, collides with the wall around the grid, or if it has been too long since the snake last ate. Each apple eaten makes the snake longer, so the game becomes increasingly difficult.

2 Your task

Design and implement a revolutionary algorithm in Python using DEAP to create an agent to play this game. You should take a principled and inventive approach to the design and implementation of your algorithm.

You must follow these rules of implementation:

- (a) You must not manually add any further movement options to the snake. It can only change direction using the provided commands (e.g. you cannot add relative movement directions).
- (b) You must not hand-code solutions for the snake. They must be found by evolutionary computation.
- (c) You may add any environmental sensing functions that you like to the snake (from the full state of the grid, down to local sensing).

3 The files provided

Go to the exam page on the EVAC VLE site and download the two files provided. These are:

- (a) `snakePlay.py`
- (b) `snakeProblem.py`

File (a) will allow you to familiarize yourself with the game by playing it. You can do this by executing the game from the terminal with: `python snakePlay.py`

File (b) is code on which you should build your algorithm. Snake movement commands are already included, along with a simple example of how to sense the environment.

4 Marking criteria for Part 1

This part is worth [45 marks].

You are expected to show a principled and inventive approach to the above investigation and present your findings effectively and methodically.

The marking criteria assume there is working code. Partial marks may be allocated for design alone. No results will be accepted if the corresponding parts of the code involved in their generation cannot be executed.

1. Quality of the design of the algorithm (e.g. consideration and choice of representation, fitness assessment, other details). Describe the chosen representation in your notebook and explain the reasons behind it. [10 marks]
2. Providing suitable code and solution for the evolutionary algorithm using Python and DEAP, and a good final solution that runs on a test game when compiled [10 marks]
3. Additional adjustments, investigation or inventive tweaks on how to improve evolution, using principles taught across the module (or beyond) [5 marks]
4. Evaluating your snake(s) and variants of your algorithms and presenting appropriate summary statistics and plots in your notebook; giving appropriate interpretation and discussion of results. [15 marks]
5. Summarising the findings in your notebook, and highlighting key points; demonstrating critical reflection on the implementation and solution; providing thoughtful suggestions for future work.[5 marks]

Part 2: Adaptive agents

5 Setup

A population of N snake agents exists on a square, wrapped around grid of size 20×20 . At each time tick, the environment may display with a probability P an apple in a random location, which does not coincide with a snake's body. The interaction of the snakes with the environment is the same as described in Part I. Each snake has its own controller defined by its chromosome. The chromosome makes use of local perceptions (grid content at Manhattan distance $d \leq 4$) and internal state to choose a movement in one of the 4 cardinal directions in the next step. Snakes are also allowed to share the directions of their last move, and this information can also be used by their genetically inherited behaviour.

Assignment Project Exam Help

6 To do

<https://tutorcs.com>

Your main objective is to (1) implement from scratch in NetLogo the environment described above. The environment should provide appropriate interface to set all relevant simulation parameters. The environment should include a default snake behaviour choosing each of the 4 directions with the same probability, then (2) use evolutionary algorithms, possibly in combination with a form of machine learning, in order to develop a population of snakes which by the end of the game reaches a total length L_{tot} that is as large as possible. An average of several runs should be used in the final evaluation. The default behaviour and the evolved behaviour should be encoded using the same representation. Hard-wiring the default behaviour in the code in another way would be penalised.

If you wish you may transfer the parameters of the controller from DEAP to NetLogo, and make use of any lessons or results learned. Nevertheless, the snakes' behaviour that is tested must have been affected by evolution implemented in the NetLogo platform. The submission should consist of a single NetLogo file, possibly accompanied by additional image and data files. If these are referenced in the Info section of the Netlogo file, a link to them should also be incorporated. The interface should make it possible to test the default behaviour of snakes, to evolve it, or to run the previously evolved behaviour that will be discussed in Info section of the submission.

7 Marking criteria for Part 2

This part is worth [45 marks].

The marking criteria assume there is working code implementing the individual objectives. Partial marks may be allocated for design alone. No results will be accepted, if the corresponding parts of the code involved in their generation cannot be executed.

1. Choose a simple and efficient representation of your agents' behaviour that would also allow for adaptation (and learning, if needed). Describe the chosen representation in the NetLogo Info section and explain the reasons behind it. [5 marks]
2. Provide the necessary, working code, describing how running the simulation is going to provide the basis for estimating the fitness of your agents' adaptive behaviour, and show how training examples for the learning component of your algorithm will be generated, if needed. [10 marks]
3. Describe the design (7 marks), and implement (8 marks) a procedure that uses adaptation, and possibly learning, to produce a viable behaviour for your agents. [15 marks]
4. Design and describe an evaluation procedure that allows you to compare the behaviour obtained through adaptation (and possibly learning) to a non-adaptive behaviour, and draw conclusions that are supported by results that are statistically significant, and based on the most appropriate statistical tests (where applicable). Assume all agents are capable of changing (learning/adaptation) their behaviours at the same time. [10 marks]
5. Collect experimental evidence, carrying out and showing the results of the evaluation procedure described above. [5 marks]

The criteria to be used in the marking also include clarity, simplicity, generality and rigour of the methods chosen, as well as the ability to describe, analyse and visualise experimental results in an effective way.

Part 3: Analysis of behaviours

8 To do

Compare the behaviours evolved in Part 1 of this assessment (individual snakes) with the behaviours evolved in Part 2 of the assessment (a population of snakes). Your answer to this part should be no more than one page, with the main body of text limited to 300 words (which includes captions but excludes the content of tables and diagrams), all submitted as a PDF document.

9 Marking criteria for Part 3

This part is worth [10 marks].

A good answer to this part may demonstrate the following

- A clear explanation of the observed behaviours
- Identification of linkage between the observed behaviours and the evolved controller for the snake or snakes (between phenotype and genotype)
- Identification of differences or similarities in behaviour
- Reflection on the significance of differences or similarities
- Reflection on the impact of parameters or methods on differences in behaviour
- Suggestions for further experimentation or analysis to understand the system(s) being evolved.

The criteria to be used in the marking also include clarity and simplicity, in addition to the ability to describe and analyse in an effective way.

End of examination paper