# Faculty of Science and Technology

# Department of Creative Technology

## Academic Year 2023/24

| | | |
|---|---|---|
| **Unit Name:** AI Games Programming | | |
| **Assignment Title:** Code and Report | | |
| **Assignment Number:** 1 of 1 | | |
| **Course/s**:<br>BSc Games Software Engineering | **Unit Level:**<br>5 | **Unit Credit Value:**<br>20 |
| **Primary Marker:**<br>Simant Prakoonwit | **Quality Assessor:**<br>Wen Tang | |
| **Assignment Issue Date:**<br>Wednesday November 15th, 2023 | **Assignment Submission Date:**<br>Friday January 19th, 2024<br>**Time:** 12:30pm | |
| **Individual or Group Information:**<br>This is an **Individual** assignment which carries **100%** of the final unit mark | | |
| **Feedback Method:**<br>Via Brightspace | **Assignment Weighting:**<br>100% of the final unit mark | |
| **Submission Method:** Online: Large/multiple files | | |

**THE ASSESSMENT TASK**

**The assignment consists of two parts. You must complete all two parts.**

# PART A NPC Decision Making and Player Imitation using ANN

## Task A1

Imagine you are given a task by your game company to create an intelligent non-player character (NPC). You would like to train an artificial neural network (ANN) that acts as the brain of your NPC. Your ANN must learn to handle your NPC's decision-making process in your game - that is, whether the NPC will attack or flee, depending on the NPC's power and the enemy's power, as described in the following table.

| INPUT | | OUTPUT |
|---|---|---|
| **NPC's power** | **Enemy's power** | **NPC's decision/action** |
| Strong (1) | Strong (1) | Attack (0) |
| Weak (0) | Strong (1) | Flee (1) |
| Strong (1) | Weak (0) | Flee (1) |
| Weak (0) | Weak (0) | Attack (0) |

**Task A1.1**. Develop a computer program to implement an ANN, with the Sigmoid function as the activation function, to make the decision according to the table. Your ANN should consist of 2 input neurons, 1 hidden layer with 2 neurons and 1 output neuron. Then train your ANN using the backpropagation technique to learn the NPC's decision/action table.

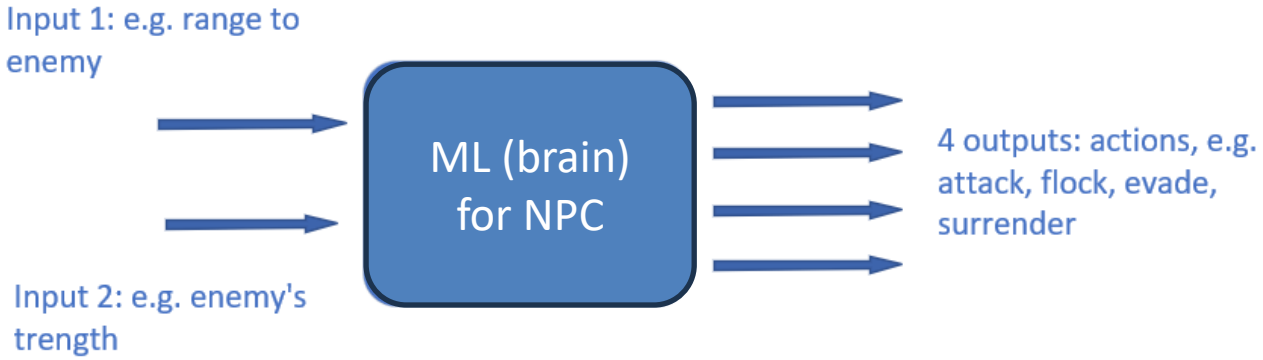
**Task A1.2**. Analyse and evaluate the effect of the learning rate and find the optimal learning rate in this case.

**Task A1.3**. Analyse and evaluate the effects of activation functions by comparing Sigmoid function with the Sign function and Step function. Find out what is the best activation function for this application.

## Task A2

Imagine you are given a task by your game company to create an intelligent NPC. You would like to train a machine learning model that acts as the brain of your NPC. Instead of making a decision according to a table, now it has to make a decision like a human so that it can interact realistically with a human player.

Assuming that your NPC (with a machine learning brain) has inputs and outputs as follows:




Example:

| Case number | Input 1 (range to enemy) | Input 2 (enemy's strength) | Outputs (actions) | | | |
|---|---|---|---|---|---|---|
| | | | Output 1 Attack | Output 2 flock | Output 3 evade | Output 4 surrender |
| 1 | 0.0354 | 1.0331 | 0 | 1 | 0 | 0 |
| 2 | -0.1302 | 1.3282 | 0 | 1 | 0 | 0 |
| 3 | 0.1665 | 0.6873 | 0 | 1 | 0 | 0 |
| 4 | 1.3223 | -0.1472 | 0 | 0 | 1 | 0 |
| 5 | 0.0643 | -0.0250 | 1 | 0 | 0 | 0 |
| ... | ... | ... | ... | ... | ... | ... |
| 1000 | -0.1051 | 0.9224 | 0 | 0 | 0 | 1 |

**Task A2.1** Conduct your own independent literature survey to identify one possible machine learning method, apart from ANN, which can be used to learn and imitate players' behaviours. Explain your justification for why you choose the method.

**Task A2.2** Systematically evaluate the method in 2.1, in comparison to ANNs. Note that in this task, you theoretically evaluate and analyse the techniques based on your literature survey. You do not need to implement the methods.

**Task A2.3** Develop a computer program to implement an ANN to learn the player's behaviour.

**Task A2.4** Systematically design and conduct experiments to determine the optimal number of neurons and layers in the ANN suitable for this task.

**Task A2.5** Evaluate the performance of your ANN.

## Conditions for Task A1 and Task A2

- For convenience and reproducibility, you must use Google Colab Notebook as your programming environment to develop and implement your task using Python programming language. Google Colab Notebook can be run on any machine.

- You are allowed to use PyTorch, TensorFlow or Keras to code your ANN. Standard auxiliary Python libraries such as NumPy, and Matplotlib are allowed. You can also use your code developed or given in the labs or lectures. If you want to use any other specific libraries or code, please discuss and obtain permission from the Unit Leader first.

# PART B Generative Artificial Intelligence

Generative AI is important for the game industry because it can offer many benefits and opportunities for game development and innovation. For example, it increases efficiency and scalability in game development by automating the content creation process and reducing the time and resources needed. It can be used to create endless possibilities for in-game content, such as images, 3D models, environments, etc., making games more diverse and engaging for players.

## Task B1

As a newly graduated programmer and you are new to Generative AI, your team leader gives you an opportunity to familiarise yourself with a simple generative model to generate simple images before moving to more advanced and realistic models. Your team leader asks you to start by using CLIP, which is a model built by OpenAI for generating images from text (https://openai.com/research/clip), or an equivalent model of your choice, to write a Python program to generate simple images from text and conduct some research and experiments to enhance your understanding in Generative AI.

**Task B1.1** Write a Python program to implement CLIP model, or an equivalent model, to generate an image sized 224x224 pixels from a text prompt.

**Task B1.2** Conduct a literature survey on how to improve the generated images in terms of the loss functions, optimisers and regularisation and other necessary hyperparameters.

**Task B1.3** Use the findings in Task B1.2 to improve the generated image quality.

**Task B1.4** Evaluate the performance of your program. To be consistent and for a fair comparison, you must use "A cat face" as your text prompt to generate images.

## Conditions for Task B1

- For convenience and reproducibility, you must use Google Colab Notebook as your programming environment to develop and implement your task using Python programming language. Google Colab Notebook can be run on any machine.
- You are allowed to use PyTorch, TensorFlow or Keras to code your ANN. Standard auxiliary Python libraries such as NumPy, Matplotlib are allowed. You can also use your code developed or given in the labs or lectures. If you want to use any other specific libraries or code, please discuss and obtain permission from the Unit Leader first.
- You can use the template Colab Notebook given in the labs or lectures in this task.

**THE DELIVERABLE AND SUBMISSION FORMAT**

You must submit the following items for assessment:

**Electronic Submission.** This assignment is to be electronically submitted by 12:30 pm on the due date (please allow sufficient time to upload files before the deadline) via:

**Report:** Turnitin Link on Brightspace
**Code and user guide:** Large File Submission Link on Brightspace

**You must submit the following items for assessment:**

**1. Report**: write a well-structured formal academic report (approx. 900 words max) in pdf format, which includes the details from the tasks. Here is an example structure and contents of the report:

**Part A**

**Task A1**

- Task A1.1: Explain your ANN, e.g. architecture, activation function, and equations involved here. This is the description of the ANN you implement.
- Task A1.2: Describe your analysis and evaluation here.
- Task A1.3: Describe your analysis and evaluation to determine the best activation function here.

**Task A2**

- Task A2.1: Describe your literature survey and your findings here.
- Task A2.2: Describe your analysis and evaluation here.
- Tasks A2.3 and A2.4: Describe your ANN architecture, your experiment design and experiments to find the optimal number of neurons and layers here. Also, mention the implementation.
- Task A2.5: Describe your performance evaluation here.

**Part B**

**Task B1**

- Task B1.1: Describe how you implement the model here.
- Task B1.2: Describe your literature survey and findings here.
- Task B1.3: Explain how you use your finding to improve the image quality here.
- Task B1.4: Describe your evaluation process and the results here.

**2. Code for Tasks A1, A2 and B1**: Submit 3 Google Colab Notebooks separately, one for each task. You must ensure that assessors who do not know your program can successfully run your program on their machines. If they cannot run your program, they cannot assess your program.
*It is recommended that if you wish to compress your submission for upload that you use a standard ZIP file because this works best with BrightSpace (rather than, e.g. RAR, 7ZIP).*

**3. Demonstration and narrative video clip**: A short video demonstrating step-by-step that your code works and can produce the required results for Tasks A1, A2 and B1. This should be of suitable quality. The resolution should be a maximum of HD (1920x1080), 1GB maximum file size, 5 minutes maximum duration. This should be compressed using MP4 that can be run by VLC on the lab computers. It is highly recommended that you test this before submission. Please do not submit a large uncompressed video

**4. User Guide**: a simple text file in your project directory to explain how to run and use your program. An in-program user guide would also be extremely helpful. Again, you must ensure that assessors who do not know your program can follow your instruction and successfully run your program on their machines. If they cannot run your program, they cannot assess your program.

**THE SUBMISSION DEADLINE**
You must submit your work by the following deadline:

**Friday January 19th, 2024**

*The deadline for all submissions is 12:30pm unless otherwise stated.*

**THE MARKING CRITERIA**
Your assignment will be assessed using the following marking criteria:

The following criteria will be used to assess the assignment:

| | | |
|---|---|---|
| **Task A1 (40%)** | | |
| Criteria 1 | The description/video and implementation of the algorithms and code | 12% |
| Criteria 2 | The design and implementation of the evaluation method/experiments | 12% |
| Criteria 3 | Analysis and experimental results to support your evaluation | 12% |
| Criteria 4 | Quality of the report/video/transferrable skills | 4% |
| | **Subtotal** | **40%** |
| **Task A2 (40%)** | | |
| Criteria 1 | The description/video and implementation of the algorithms and code | 12% |
| Criteria 2 | The design and implementation of the evaluation method/experiments | 12% |
| Criteria 3 | Analysis and experimental results to support your evaluation | 12% |
| Criteria 4 | Quality of the report/video/transferrable skills | 4% |
| | **Subtotal** | **40%** |
| **Task B1 (20%)** | | |
| Criteria 1 | The description/video and implementation of the algorithms and code | 6% |
| Criteria 2 | The design and implementation of the evaluation method/experiments | 6% |
| Criteria 3 | Analysis and experimental results to support your evaluation | 6% |
| Criteria 4 | Quality of the report/video/transferrable skills | 2% |
| | **Subtotal** | **20%** |
| | **Total** | **100%** |

NB: see the marking rubric attached.

**THE LEARNING OUTCOMES:**

This assignment will assess the following ILOs:

1. Demonstrate critical understanding of algorithms used for AI in game development;
2. Show an understanding in the applications of AI for computer games programming;
3. Model and implement artificial intelligence algorithms for a computer game;
4. Examine the high-level relationships between AI and game systems.

**QUESTIONS ABOUT THE ASSIGNMENT BRIEF:**

You should address any questions about this assignment brief to the Unit Leader, whose details are shown below:

Unit Leader: **Simant Prakoonwit**

Unit Leader Email: **sprakoonwit@bournemouth.ac.uk**

You can also ask questions in lectures and lab sessions.

**SIGNATURE OF UNIT LEADER: Simant Prakoonwit (via email)**

## ASSIGNMENT GUIDANCE NOTES – Academic Year 2023-2024

You must keep a copy of your assignment – the university will not take responsibility for lost assignments. Please make sure you back up your work carefully.

### Submission Deadlines:

If a piece of coursework is not submitted by the required deadline, the following will apply:

1. If coursework is submitted within 72 hours after the deadline, the maximum mark that can be awarded is 40%. If the assessment achieves a pass mark and subject to the overall performance of the unit and the student's profile for the level, it will be accepted by the Assessment Board as the reassessment piece. The unit will count towards the reassessment allowance for the level; This ruling will apply to written coursework and artefacts only; This ruling will apply to the first attempt only (including any subsequent attempt taken as a first attempt due to exceptional circumstances).

2. If a first attempt coursework is submitted more than 72 hours after the deadline, a mark of zero (0%) will be awarded.

3. Failure to submit/complete any other types of coursework (which includes resubmission coursework without exceptional circumstances) by the required deadline will result in a mark of zero (0%) being awarded.

The Standard Assessment Regulations can be found on **Brightspace**.

### Exceptional Circumstances:

If you have any valid **exceptional circumstances** which mean that you cannot meet an assignment submission deadline and you wish to request an extension, you will need to complete and submit the Exceptional Circumstances Form for consideration to your Programme Support Officer (based in C114) together with appropriate supporting evidence (e.g, GP note) normally **before the coursework deadline**. Further details on the procedure and the exceptional circumstances form can be found on **Brightspace**. Please make sure that you read these documents carefully before submitting anything for consideration. For further guidance on exceptional circumstances please see your Programme Leader.

### Avoiding Plagiarism:

You must acknowledge your source every time you refer to others' work, using the **BU Harvard Referencing** system (Author Date Method). Failure to do so amounts to plagiarism which is against University regulations. Please refer to http://libguides.bournemouth.ac.uk/bu-referencing-harvard-style for the University's guide to citation in the Harvard style. Also be aware of Self-plagiarism, this primarily occurs when a student submits a piece of work to fulfill the assessment requirement for a particular unit and all or part of the content has been previously submitted by that student for formal assessment on the same/a different unit. Further information on academic offences can be found on **Brightspace** and from https://www1.bournemouth.ac.uk/discover/library/using-library/how-guides/how-avoid-academic-offences

### Generative Artificial Intelligence:

When you submit your assessed work, you do so on the understanding that it is your original work. The use of Generative Artificial Intelligence (GAI) may be permitted, please refer to the Homepage - Academic Skills Hub (bournemouth.ac.uk)for guidance for appropriate use of GAI and referencing.

**Accessing Learning Support:**

Students with **Additional Learning Needs** may contact Learning Support on www.bournemouth.ac.uk/als

**Ethical Compliance:**

You should not be conducting any primary research (i.e. carrying out an investigation to acquire data first-hand, for example, where it involves approaching participants to ask questions or to participate in surveys, questionnaires, interviews, observations, focus groups, etc.) unless otherwise specified in the brief. However, if there is a genuine requirement to collect primary research data you will require ethical approval before doing so. In the first instance, please discuss with the Unit Leader. The collection of primary data without appropriate ethical approval is a serious breach of Bournemouth University's Research Ethics Code of Practice and will be treated as Research Misconduct.

**Disclaimer:**

The information provided in this assignment brief is correct at time of publication. In the unlikely event that any changes are deemed necessary, they will be communicated clearly via e-mail and Brightspace and a new version of this assignment brief will be circulated.

**Assignment Reference: AIGP_Ass1_AY2324_RELEASE**

**Version Number: 301023_RELEASE**

Marking Rubric

| | **1. Description of algorithms (10%) the implementation/code (20%)** | **2. Design and implementation of the evaluation method/experiments (30%)** | **3. Analysis and experimental results to support the evaluation (30%)** | **4. Quality of the report/transferable skills (10%)** |
|---|---|---|---|---|
| **80%+** | Algorithms: The work demonstrates outstanding knowledge of the algorithms. An exceptional conceptual understanding has been demonstrated<br><br>Code: Outstanding code quality, clearly designed structure, well organised and written to a consistent standard. Outstanding code comments, clearly enhancing code readability, written to a standard for documentation.<br><br>Code performs task outstandingly. Demonstration of a broad factual and conceptual understanding of the algorithms. | The work demonstrates appropriate and effective techniques in design and implementation of the evaluation methods/experiments. It also demonstrates an exceptional competence and confidence to apply an appropriate range of knowledge in the evaluation process. | The works demonstrates logical and well-structured argument/reasoning using well selected collated and presented evidence. | The work is exceptionally well-organised. Arguments are well structured and rigorous. Sentence structure and grammar indicate excellent academic skills. |
| **70-79%** | Algorithms: The work demonstrates a detailed knowledge of the algorithms. Excellent conceptual understanding is demonstrated<br><br>Code: Excellent code quality, clearly designed structure, well organised and written to a consistent standard. Excellent code comments, clearly enhancing code readability.<br><br>Code performs task extremely well. Demonstration of a broad factual and conceptual of the algorithms. | Very appropriate techniques of evaluation have been selected. The work demonstrates an excellent application of a range of knowledge competently address the issues in the evaluation process. | A range of information has been analysed. Critically selected information supports logical and well-structured argument or reasoning. | The work is very well-organised with structured and rigorous arguments. It is well communicated with few mistakes in presentation, grammar and sentence structure. |
| **60-69%** | Algorithms: The work demonstrate a very good understanding of the algorithms. Sound conceptual understanding has been demonstrated.<br><br>Code: Good code quality, readable and consistent in style, with a clearly defined structure and good attempt at design. Good and clear code comments at appropriate places which enhance code readability.<br><br>Code performs the basic task well. Demonstration of sound understanding of the algorithms. | The work demonstrates a very good application of a range of knowledge competently address the issues in the evaluation process. Appropriate techniques of evaluation have been selected. | The work demonstrates very good analysis of the selected information in support of the argument. Discussion is logically developed from sequentially established results. | The work presents a logical structure and few mistakes are evident in grammar/sentence structure and the use of English. Presentational style is sound |

| | | | | |
|---|---|---|---|---|
| **50-59%** | Algorithms: The work demonstrates some good understanding of the algorithms.<br><br>Code: Acceptable code quality with some errors, mostly readable, some structure and organisation, but inconsistencies in style. Some meaningful code comments.<br><br>Code performs the basic task. Some evidence of enhanced understanding of the algorithms. | The work demonstrates and emerging application of theory to develop the evaluation process. | The work demonstrates some critical skills in analysis and evaluation of the results. | There is some attempt at structure and organisation but this is inconsistent. Presentation style and communication skills show some limitations. |
| **40-49%** | Algorithms: The work demonstrates a basic knowledge and understanding of the algorithms. Knowledge has not been developed in depth and there are some difficulties with theories and concepts.<br><br>Code: Poor code quality with some clear errors, badly formatted, difficult to read, poor attempts at structure. Minor attempt at code comments.<br><br>Code runs as expected and performs basic tasks. Demonstration only a basic level of conceptual understanding of the algorithms. | The work has demonstrated only a basic evaluation process. Only limited ability to design and implement the evaluation process is presented. | The work has demonstrated only a basic analysis or the results and some unsubstantiated opinions. | The structure of the work is weak and/or inconsistent and not well organised. There are mistakes in grammar or sentence structure and the writing style needs development. References are poorly or inconsistently presented. |
| **30-39%** | Algorithms: The work shows an insufficient level of factual and conceptual understanding of the algorithms.<br><br>Code: Very limited code structure or attempts at organisation. Poor code with many errors. Very limited comments in code.<br><br>Code does not compile/program does not run and does not attempt at answering the task. Shows an insufficient level of factual and conceptual understanding of the algorithms. | The work shows little or no evidence of design and implementation of the evaluation process. | The work is entirely or virtually entirely descriptive, showing little or no evidence of analysis. Unsubstantiated opinions may be common. | The structure is weak or lacking and the content is unclear. There are mistakes in sentence structure and grammar. Some references may be incorrect or missing. |
| **0-29%** | The work shows little or no evidence of factual and conceptual understanding of the algorithms.<br><br>Code: No real code structure or attempts at organisation. Poor code with many errors. No comments in code.<br><br>Code does not compile/program does not | The work shows very insufficient evidence of appropriate evaluation. | The work shows very insufficient evidence of appropriate analysis. | The work is unstructured and/or written in very poor English and/or an inappropriate style. References may be presented incorrectly or missing. The work may be incomplete and/or very brief. |

| | | | | |
|---|---|---|---|---|
| | run and does not attempt at answering the task. Shows no factual and conceptual understanding of the algorithms. | | | |