University of Western Australia

CITS3001

Agents, Algorithms and Artificial Intelligence

Super Mario Project

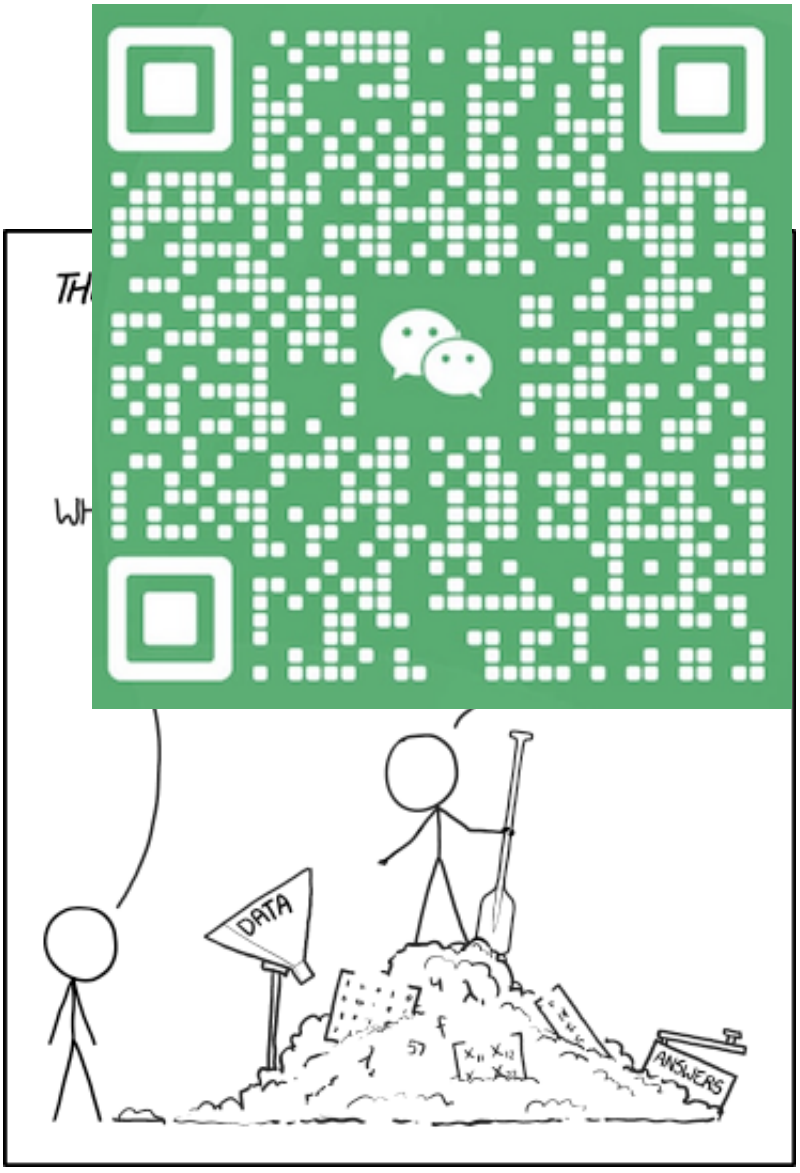
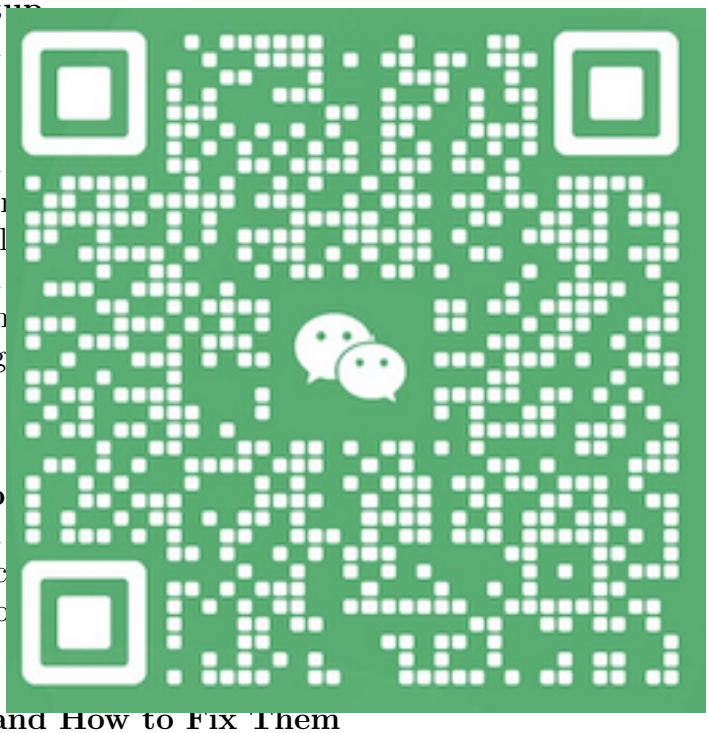


Figure 1: Sourced from XKCD [1]

Contents



1 Project Overview

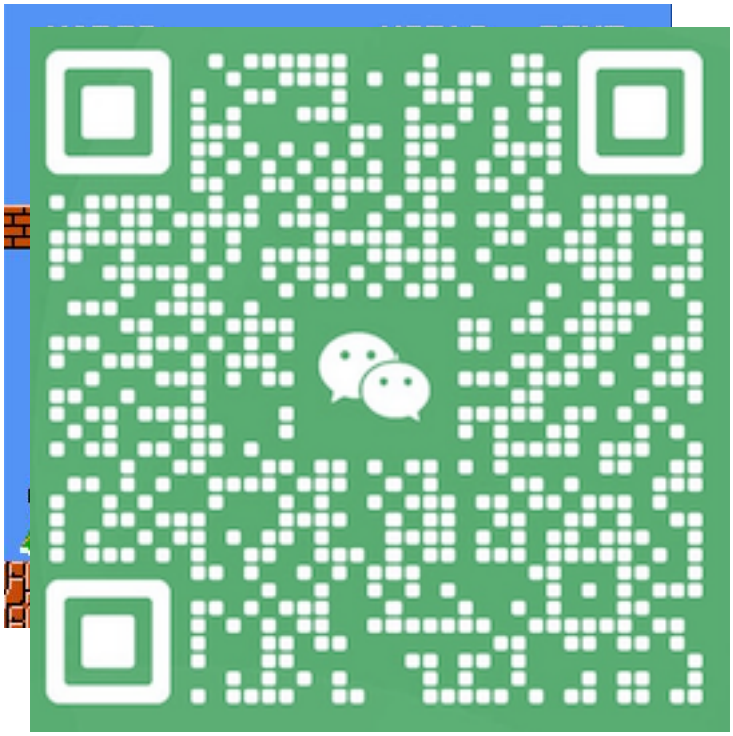
In this project, you will develop AI agents to control the iconic character Mario in the classic game Super Mario Bros using the gym-super-mario-bros environment. The main objective is to implement at least two distinct AI algorithms/methods and compare their performance, strengths, and weaknesses in the context of playing the game.

You can undertake this project in teams of 2 that you select, if you are looking for a partner please reach out to your lab demonstrators by emails

1.1 Requirements

1.1.1 Gym-Super-Mario-Bros Environment Setup

- Set up the gym-super-mario-bros [2] environment on your local machine or any designated platform, see 4.2.1Environment Creation.



1.1.2 AI Algorithm Implementations

- Choose and implement at least two AI algorithms. You may wish to consider the following:
  - Reinforcement Learning: Q-learning [3], TD(λ) [4]
  - Rule-Based AI: logic and heuristics.
  - Monte Carlo Tree Search (MCTS) [5]

You are welcome to use more advanced algorithms that utilise deep learning such as DQN's [6] or Proximal Policy optimisation etc. but these are not covered in the unit and lab facilitators may not be able to assist with your implementations. These algorithms will also have to be referenced in your project report.

1.2 Final Project Report

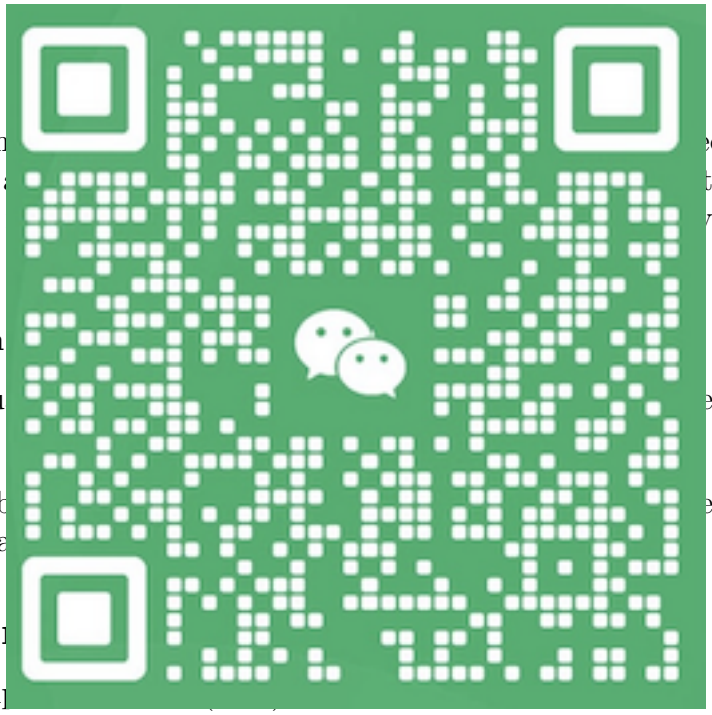
To demonstrate your understanding of your implementations you will be required to write a final project report. Your report must conform to the following guidelines

- At least 3 pages
- No longer than 5 pages
- No smaller than size 12 font

You are allowed to add appendices with extra figures and words but these may not be marked. Your report should cover the following areas:

1.2.1 Analysis

- Analyze and contrast the performance of the chosen AI methods.
- Discuss their respective strengths, weaknesses, and suitability for playing Super Mario Bros.



1.2.2 Performance

You will notice that gym ctive of the game is to move as far right ther performance and evaluation metrics you come up with will be rewarded.

1.2.3 Visualization

- Include what visu e agent's decision-making process.
- Include what deb e algorithms and enhance performa

1.3 Marks Dist

- AI Algorithm Im
  - Successful implementation of two or more AI methods (20%)
  - Code quality, readability, and efficiency (10%)
  - Integration with the gym-super-mario-bros environment (10%)
- Report Comparison and Contrast (30%)
  - In-depth analysis of the algorithms' strengths and weaknesses (15%)
  - Properly conducted experiments and results presentation (10%)
  - Effective comparison of AI methods (5%)
- Performance Metrics and Visualization (20%)
  - Selection and definition of appropriate performance metrics (10%)

  - Quality of visualization techniques (10%)

- Report Quality (10%)
  - Clarity and organization of the report (5%)
  - Overall presentation and writing quality (5%)

1.4 Additional Notes

- You are encouraged to use external libraries and frameworks to support your implementation, but remember to provide proper citations and acknowledgments.
- Regularly check in with the instructor for progress assessments and guidance during the project timeline.
- Ensure that your work is original and properly referenced to maintain academic integrity throughout the project.

1.5 Clarificatio

- any python packa y are included in your environment
- You are permitted e **SuperMarioBros-v3** see 3. Howeve ll be looked upon favourably.

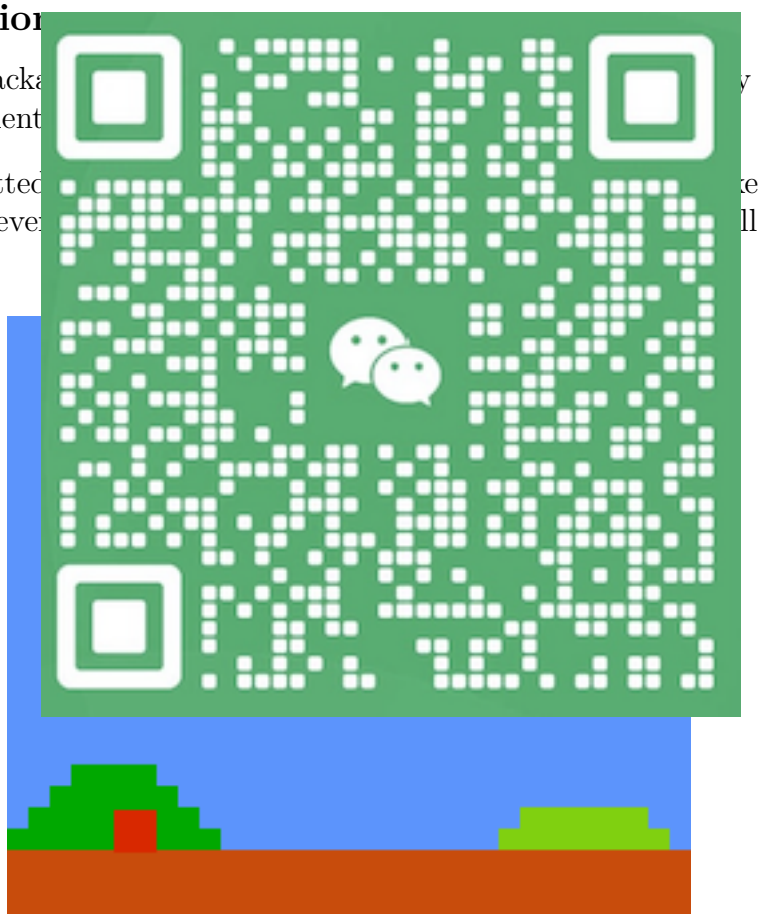


Figure 3: SuperMarioBros-v3 example environment which is allowed

2 Submission Guidelines

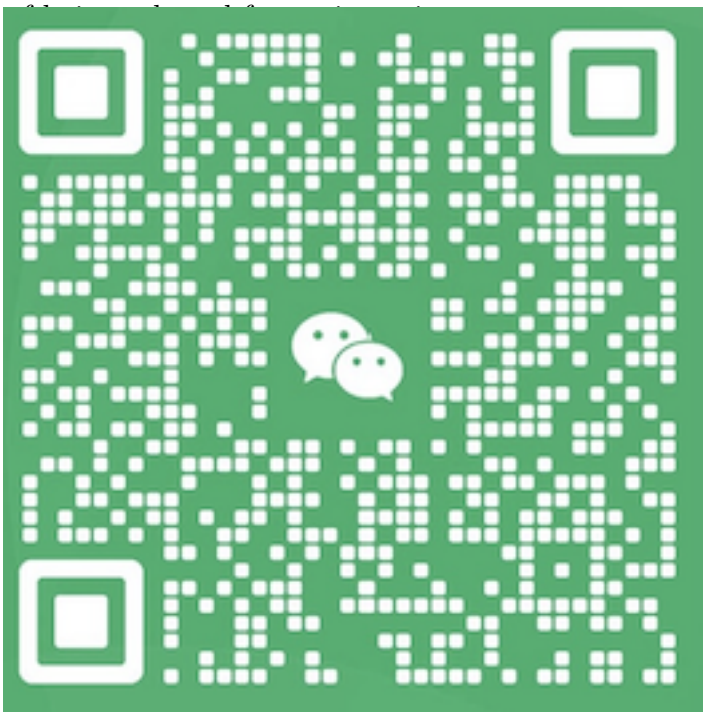
You will be required to submit the following:

- A zip file containing the code for your agents
- Your final report as a PDF file

Please ensure your code submission contains a README.md file explaining how to setup the environment for your project and how to run the two separate agents individually

3 Interviews

If markers suspect instances of plagarism or excessive reliance on ChatGPT or any external sources, a procedural measure has been established. Students found in such situations may be requested to attend an interview with their respective lab instructor. The purpose of this interview is to provide students with an opportunity to demonstrate their understanding of the project. It is worth noting that consistent attendance in laboratory sessions significantly mitigate the likelihood



4 Environment Setup

You are using an official public package, so please don't spam the github issues. I have created a fork for our unit on github that you can file specific requests under

4.1 Conda

Anaconda Python is the recommended method of installation for this project

4.1.1 Linux

Download installer from anaconda
you may need to run

```
conda init bash
```

depending on your shell

4.1.2 Macos

follow the instructions

4.2 Windows

firstly follow the instru

You will need to in e microsoft build tools.

So install the follow

- visual studio (not
- visual studio buil

the build tools are r g the gym.

4.2.1 Environment

Open anaconda prompt t open powershell or command prompt as

Then create the new conda environment with:

```
conda create -n mario python=3.8
```

then activate the environment.

```
conda activate mario
```

4.3 Package Installation

There isn't a working conda package for gym-super-mario-bros so it is best to install with pip, this package will be installed inside the conda environment if you have it activated

```
pip install gym-super-mario-bros
```

4.4 Poetry

You only need to use one package manager, so use conda or poetry not both
Poetry has been less successful so use at your own risk per platform.

4.4.1 Tool Installation

See poetry documentation on how to install for your system.

```
curl -sSL https://install.python-poetry.org | python3 -
```

4.4.2 Package Installation

```
poetry add gym-super-mario-bros
```

```
poetry install
```

5 Test Code

```
from nes_py.wrapper
import gym_super_ma
from gym_super_mari
import gym

env = gym.make('Sup                                    ue, render_mode="human")
env = JoypadSpace(e

done = True
env.reset()
for step in range(5
    action = env.ac
    obs, reward, te                                    )
    done = terminat

    if done:
       state = env.

env.close()
```

6 Running Test Code

6.1 Windows

Due to the graphics library requirements of gym-super-mario-bros it will be challenging to use WSL to complete this project. As such a native windows approach is recommended.

6.2 VS-Code (reccomended)

Use **Ctrl-Shift-p** to open up the command prompt and run

```
Terminal: Select default profile
```

and change it to command prompt. Anaconda prompt will not be available as an option, so don't worry. We are changing this as the default option for windows is powershell which doesn't play nicely with conda environments.

Next open up the command prompt again and run

```
Python: Select Interpreter
```

and change this to the conda or poetry environment you have created (this assumes you have already created one, if not see the section on environment creation)

With this configured you should be able to click the run button in the top left hand corner on any test code. If you get and import error try pressing the run button again as there have been issues where the python file is run before the conda environment has been activated.

6.3 Command Prompt

Use anaconda prompt (not powershell or command prompt) to run the following

```
conda activate mario
```

Then run the test c

```
python test.py
```

6.4 Linux/Mac

On a terminal make su

```
conda activate mari
```

Then run the test c

```
python test.py
```

7 Example Is

7.1 Tuple unpa

```
DeprecationWarning: ‘np.bool8‘ is a deprecated alias for ‘np.bool_‘.  (Deprecated NumPy 1.24)
Traceback (most recent call last):
  File "/home/dadams/Desktop/Other/super-mario/test.py", line 11, in <module>
    state, reward, done, info = env.step(env.action_space.sample())
                                ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "site-packages/nes_py/wrappers/joypad_space.py", line 74, in step
    return self.env.step(self._action_map[action])
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "gym/wrappers/time_limit.py", line 50, in step
    observation, reward, terminated, truncated, info = self.env.step(action)
    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
ValueError: not enough values to unpack (expected 5, got 4)
```

This issue was caused by an api cmpatability error so simply put:

```
env = gym.make('SuperMarioBros-v0', apply_api_compatibility=True, render_mode="human")
```

References

[1] Randall Munroe. Seashell. `https://xkcd.com/1236/`, 2013. [Online; accessed October 20, 2019].

[2] Christian Kauten. Super Mario Bros for OpenAI Gym. GitHub, 2018.

[3] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8:279–292, 1992.

[4] Harm Seijen and Rich Sutton. True online td (lambda). In *International Conference on Machine Learning*, pages 692–700. PMLR, 2014.

[5] Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 4(1):1–43, 2012.

[6] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level contro 8(7540):529–533, 2015.