

WORKSHEET 2: ARTIFICIAL INTELLIGENCE USING BEHAVIOUR TREES

Version 1.0
Computing
COMP280

Joseph Walton-Rivers

Introduction

In this worksheet, you will implement a set of AI behaviours for the enemies in a simple game.

Begin by **forking** one of the following git repositories:

This repositories contain a project based on the classic arcade game *Pac-Man*. The player uses the WASD keys to run around a maze collecting pills. The maze is also inhabited by four ghosts: these will kill the player upon contact, unless a power pill is active, in which case the player kills the ghost. In the template project, the ghosts (implemented in the GhostCharacter class) stay still in the centre of the maze; your task for this worksheet is to implement suitable AI behaviours.

Unreal

<https://github.com/Falmouth-Games-Academy/comp280-worksheet-2>

Unity

<https://gamesgit.falmouth.ac.uk/users/josephwr/repos/comp280-unity-ai/browse>

To complete this worksheet:

- (A) **Fork** the repository linked above.
- (B) **Implement** a basic Behaviour Tree based AI controller for the ghosts. A good starting tutorial is available at https://wiki.unrealengine.com/Blueprint_Behavior_Tree_Tutorial, Unity users should refer to the tutorial mentioned in the comments and the lecture slides.
- (C) **Iterate** on your AI to produce suitably complex and engaging behaviours.
- (D) **Open a pull request** by 22nd November 2019 to receive formative feedback.

Additional guidance

In many genres of game, the design of AI agents is key to delivering a good player experience. It would be easy to create virtually unbeatable ghost AI for Pac-Man, however this would not lead to enjoyable gameplay.

The following are some ideas for how you might make your solution more interesting (all of these are optional suggestions):



Pac-Man is a classic example of relatively simple AI behaviours giving rise to a compelling gameplay experience.

Unreal

- Give each of the ghosts its own distinctive behaviour. (Note that if you choose to subclass GhostCharacter to allow for this, you can edit the GhostSpawners in the level to cause them to spawn instances of your new class.)
- Have the ghosts behave differently based on whether a power pill is active.
- Give the ghosts limited perception so that they have to find the player.
- Have the ghosts take the state of the game, e.g. locations of pills left, into account.

Unity

- Give each of the ghosts its own distinctive behaviour. (Note that if you choose to modify BtController to allow for this - for example, swapping the tree based on the name of the agent).
- Have the ghosts behave differently based on whether a power pill is active.
- Give the ghosts limited perception so that they have to find the player.
- Have the ghosts take the state of the game, e.g. locations of pills left, into account.

For more guidance, please read the over-arching assignment brief for this assignment on LearningSpace.

Marking Rubric

To **pass** this assignment (achieve 40% or more), you must submit a reasonable attempt at the worksheet by the formative deadline stated on LearningSpace.

Criterion	Weight	Near Pass	Adequate	Competent	Very Good	Excellent	Outstanding
Behaviour design	60%	AI behaviours are non-existent or are extremely basic.	AI behaviours are basic, but the gameplay experience is not well balanced.	AI behaviours show efforts towards sophistication, but the gameplay experience is not well balanced.	AI behaviours are moderately sophisticated, with efforts towards a well balanced gameplay experience.	AI behaviours are sophisticated, well thought out, and give a well balanced gameplay experience.	AI behaviours are extremely sophisticated, well thought out, and give a very well balanced gameplay experience.
Sophistication and maintainability	40%	<p>The project shows little or no understanding of relevant techniques and tools.</p> <p>Code is very poorly structured and commented, to the point of being unreadable.</p>	<p>The project shows basic understanding of relevant techniques and tools.</p> <p>Code is readable but lacking in structure and commenting.</p>	<p>The project shows some knowledge of relevant techniques and tools.</p> <p>Efforts towards structuring and commenting of code.</p>	<p>The project shows good knowledge of relevant techniques and tools.</p> <p>Code is well structured and commented.</p> <p>Some parameters to the AI are easily tunable outside the blueprints / behaviour trees.</p>	<p>The project shows extensive knowledge of relevant techniques and tools.</p> <p>Code is very well structured and commented.</p> <p>Many parameters are easily tunable outside the blueprints / behaviour trees.</p>	<p>The project shows mastery of relevant techniques and tools.</p> <p>Code is extremely well structured and commented.</p> <p>All relevant parameters are easily tunable outside the blueprints / behaviour trees.</p>