[Files of Interest Reference Guide starts on Page 3]

System Requirements:
Java Runtime Environment 1.7 or higher

Code Execution:

Windows / Apple: Run “PacDaddyEngine.jar”

Linux: On some machines you may have to execute the “run.sh” script from the command line

Interaction:

KEYS	DESCRIPTION
ARROW KEYS	Move Pacman. Pacman's AI typically takes supervisory control.
ENTER	Restart game when lives reach 0
R	Reload the game features
Q / ESC	Quit the game
P	Pause the game update loop.
1 / 2	Change number of lives
3 / 4	Change the game speed, in updates per second (UPS)
5 / 6	Change the render speed, in frames per second (FPS)
7 / 8	Change the level
9	Toggle Wall Shuffling
0	Toggle Player AI

Recommendations:

- To analyze the decision making of the AI, we recommend lowering the game update speed or frequently pausing the game, since the AI processing is independent of the game update speed.
- If the display appears jittery, please lower the framerate.

Bugs:

- Sometimes pellets won't get eaten by the player, which messes with the AI and game logic. This was due to an optimization of the engine for more efficient collision checking, which was necessary because large scale games would lag. If this happens, please reload by pressing the R key; otherwise, the level system cannot progress.
- Sometimes tile shuffling can warp pellets or ghosts into walls. This is part of the same issue with the collision check optimization.

[Files of Interest Reference Guide]

Within the “AI-Project-Mac-Pan” directory.

Artificial Intelligence

The AI files are in:
/features/Reloadable/AI/

- The implementation of BFS can be found in **bfs.lua**
- The implementation of A* can be found in **astar.lua**
- The Gravity Map implementation can be found in **GravityMap.lua**
- The Gravity Field (theoretical gravity) implementation can be found in **GravityField.lua**
- The Tile Shuffling implementation can be found in **wallShuffler.lua**
- The player AI is setup in **setupPlayerAI.lua**
- The enemy AI is setup in **setupEnemyAI.lua**
- The rate in which AI calls are made and in what sequence is setup in **setupAI.lua**

The “setup” files include (keyword “require” in lua) the search and gravity algorithms.

Separation of Concerns through Threaded Routines

The setup of the threaded components of the system can be found in:
/features/Init/**wrapPacDaddyGame.lua**

The separate threads for AI are constructed and injected here.

Reloadability

All of the specifically reloadable parts of the system are in:
/features/Reloadable/

The feature which makes the “Reloadable” directory reload can be found in
/features/Reloadable/Controls/**setupControls.lua**

Pactors

Different types of bodies in the system are called “Pactors”. The different types of “Pactors” can be found in:

/PacDaddyGameWrapper/

Ex: **Enemy.lua**, **Player.lua**, **Pellet.lua**, **Energizer.lua**

Levels

The Level layouts can be found in:

/levels/

The .txt files are tile layouts and the .lua files populate the tiled world

The numbers represent tiles, which were setup here:

/features/Reloadable/Level/**setupAdditionalWorldTiles.lua**

Data Structures, etc.

General purpose data structures, timers, threads, and futures can be found in: /luasrc/