COMP 452 TME 2

Jason Bell 3078931

June 26, 2016

Overview

The game executable can be found in the /dist folder

The sprites were drawn myself, and are clear evidence of my complete lack of artistic talent.

The only sound file is the "Wilhelm Scream", which has no owner and is so ubiquitous to be in the common domain (https://en.wikipedia.org/wiki/File:Wilhelm_Scream.ogg)

Implementation

This assignment was implemented in Python, with permission from Dr. Esmahi, to gain experience with that language: I already have extensive experience with Java. He required that I provide both source and built executables to ease marking.

There were no technical hurdles in creating this assignment, and I know of no bugs.

The two games use the same A* pathfinder. This pathfinder is interruptible and generates the path in steps so that it's progress can be viewed. This also means that it could be invoked across multiple frames.

The state machine is implemented verbatim from the assignment requirements, except that I was unsure if food should disappear. I opted to do so, to prevent population explosions. There are four states: dead, seeking food, returning food, and seeking water

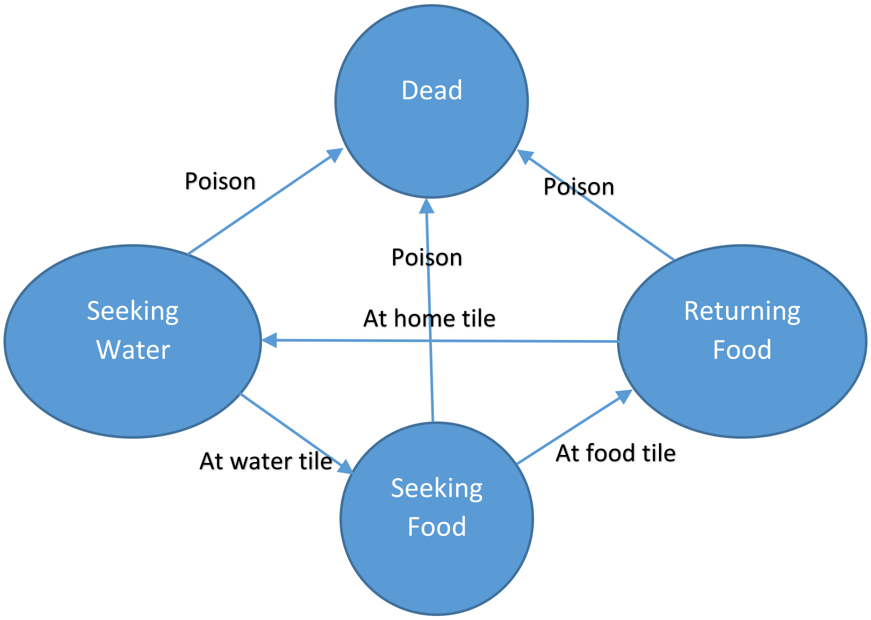
In any current state, if the ant is on a poison tile it enters the dead state.

If the ant is in the dead state, no action can be taken. It's dead.

If the ant is in the seeking food state, it moves to random adjacent tiles until it enters a food tile. That tile becomes grass, and the ant enters the returning food state and a path to the home tile is created.

If the ant is in the returning food state, it follows the path until it reaches the home tile. Then it spawns a new ant and enters the seeking water state.

In the seeking water state, the ant moves to random adjacent tiles until it enters a water tile. It then enters the seeking food state.



If you wish to run from the scripts, you will need to install 32-bit Python 2.7.

Then, install any version of Visual Studio 2015. Full versions are available through DreamSpark, and the community and express versions will also work. You'll need to install python tools for visual studio: you'll see an option to do this in the "new project" window.

Once python tools have been installed and this project is opened, you may need to install pygame. To do this, you should see a "Python Environments" tab in the upper right next to the solution explorer. Click it, then select Python 2.7. Use the pip option to search for and install pygame (pip is Python's equivalent to Maven).

I do not recommend trying to use py2exe to build an executable: I suggest removing BuildExe.py from the project. Then, start the game with Game1.py or Game2.py as the startup script.