

Programming Project 4: AI

Due: Tue, Oct 28 (Design document due Oct 21)

Objectives:

- learn how to implement steering behaviors to control movement
- learn how to implement finite state machines for decision making

For this project, you will implement one or more AI driven characters. These characters should use steering behaviors to control movements such as, for example, pursuing the player character and evading the player character. And they should use a finite state machine to decide when to use which behavior based on information about the environment and the player character.

As for the game in which to do this, you have three options:

- Use your game from the last assignment and incorporate AI driven characters.
- Design a new game of your own in which AI driven characters play an interesting role.
- Add an AI enemy to jetpac. I will make a basic implementation of the physics components of jetpac available. (By Friday.)

Strategy:

Implement a class Agent, which contains all the AI logic. Each Agents should have a PhysicalBody and be subject to the laws of collisions and gravity just like the player. (So you need to reuse your PhysicalBody and Sprite classes.) Each AI character in your game will be an instance of type Agent. That is, your class should be general enough so that you can use it for more than one character in the game.

The design document:

The design document should have four parts:

- A description of your game (not necessary for jetpac) and how your AI characters

will behave (necessary for all options).

- A visualization of the finite state machine(s) that you are going to use. (Diagram with states and transitions between states as in class.)
- The specification of the Agent class (member variables, methods and their signatures).
- High level pseudo-code for the update method of the Agent class as well as for any new member methods of the Agent class that get called by the update method.

Grading:

The grades will be based on the following criteria:

- whether the basic requirement of implementing a game with AI is met
- whether both steering behaviors and finite state machines are used
- the correctness of the AI implementation
- the overall correctness of the code
- the overall organization of the code
- whether the code is appropriately documented (through comments)
- the quality of the design document

Again, I will give bonus points for effort that I think goes beyond the expected, such as: your own game designs, particularly polished games or nice graphics, particularly innovative ideas,