Overview

Your task for this assignment is to explore pathfinding and decision making algorithms. Working alone and using Processing (http://www.processing.org/), your task is to implement the algorithms described below and to analyze your results in a 3–5 page writeup. Note that for some of these tasks you will be integrating your new solutions with your code from Assignment 1.

First Steps

Create two graphs that you will use for your experiments. Both graphs should be a weighted singly-connected digraph. Remember, only use positive weights.

Your first graph should represent something meaningful in the world. Perhaps a road map of Raleigh or your hometown, the offices in EB2, etc. Your choice of what you represent is up to you. Hint: read ahead to figure out how to maximize your work effort. This graph should be large enough so that you can learn some interesting things about the algorithms, but small enough to enable efficient computation in your experiments. As a rule of thumb, 10 vertices is probably too few, but 50 is likely too many.

Your second graph should be designed to test the limits of the algorithms. It should be big. Very big. You can generate it randomly, find a graph somewhere on the internet (make sure you cite where!), or try to generate it using some data set of your choosing. One thing is for sure, you should not author this graph by hand.

Dijkstra’s Algorithm and A*

Your next task is to implement both Dijkstra’s Algorithm and the A* Algorithm. Hint: get them both working on your first graph before you begin testing on your second graph. For A*, pick a simple heuristic like a constant guess or some form of Manhattan Distance. What you choose isn’t all that important for this section of the assignment.

Compare and contrast the performance of the two algorithms on both of your graphs in terms of runtime, number of nodes visited, and memory used. What else can you say about these algorithms? What effects does the graph structure have on performance? You are expected to present data in your writeup to support your analysis.

Heuristics

Looking solely at the A* algorithm, design and implement at least three heuristics. Note, the heuristic you used in the previous portion of the assignment can count as one of these three.

Putting it All Together

Combine your pathfinding algorithms with the appropriate algorithms from Homework 1. Design an “indoor environment” that contains a number of obstacles for your character to avoid. Using a principled technique of your choosing, create a graph representation of that environment and use it to perform pathfinding. Hint: Although more computationally intensive, having a dense graph can avoid the need for obstacle avoidance in your movement system. Videos or screenshots of your character in action are a must.

Behavior Trees

Your final task for this assignment is to incorporate decision making into your movement and pathplanning system. Your task is to devise a parameterization of your environment that will enable you to build a behavior tree model to control changing of targets or movement behaviors. For example, you may decide that whenever your character reaches maximum velocity, its behavior should change to wander. Or whenever it wanders near a wall, its behavior should change to seek with a particular target location (like the center of the screen). Or after seeking for 3 seconds, the target location should randomly change. Your behavior tree will be responsible for the decisions made about when and where to update target locations, when to pathfind to those targets, when to change movement behaviors, etc. Use your imagination. And more importantly, as always, write about it!

Extra Credit: Collision/Obstacle Avoidance

For extra credit, implement collision detection and obstacle avoidance algorithms for your movement system. Then integrate that into your behavior tree.

Writeup

Now that you have implemented and evaluated a number of algorithms, write a 3–5 page paper summarizing your findings. It is strongly suggested that you do not limit yourself to only answering those questions posed in this assignment. Think creatively about what you have done. What other parameters can you tweak and what effect do they have on the results? The most successful writeups will contain evidence that you have thought deeply about these algorithms and what they can produce and have gone beyond what is written in the assignment.

As an appendix to your paper, please include all relevant screenshots to support your analysis. The appendix does not count toward your 3–5 page requirement.

What to submit

By the start of class on 3/16/11, please upload a zip archive to moodle. This archive should contain all of your appropriately labeled files from each part of the assignment, a README file, plus your writeup in pdf format. For this homework, rather than require a specific naming convention for each of your files, please include a README with instructions for compiling and running your code. In the README, you should list which files correspond to which parts of your homework assignment.