Spike: Graphs and Search

Context:
Graph can be used to represent many aspects of a game, and selecting and using the right type of graph search is critical to finding solutions to problems in games.

Knowledge/Skill Gap:
The developer needs to be able to use appropriate graph search algorithms to solve problems for characters and controllers game situations.

Goals/Deliverables: [CODE] + [SPIKE REPORT]
Clearly demonstrate the appropriate use of the Dijkstra's (search for item) and A* (search to position).

- Modify the graph search lab code, or create your own simulation.
- Add in a simple moving agent to moves to each way-point in a successful graph search result.
- Be able to demonstrate either search-for-item or search-to-point examples.
- Make sure your agents (or their graph searching algorithms) correctly consider wall, mud or water tiles in the map.
- Display path cost for comparison
- Clearly demonstrate the need for different search algorithms

Start-End Period: Week 9 – Week 10

Planning Notes:

- Keep this very simple. Use the lab code and add a moving agent.
- Avoid complex agent movement code. Simple (non-steering force based) movement will suffice. (Look at the fleet-movement code from the Planet Wars code for an example of direct fixed-speed movement between two points.)
- Don't worry about path smoothing.
- Don't worry about multiple agents.
- Even though you might think about search-for-item as a "health pack" or similar, DO NOT implement the full code for this feature! (It's not the point of the lab.) Just show a simple icon (square, circle) on search to indicate the objectives.

Extensions:

- Add multiple agents, each doing their own planning. Demonstrate the performance hit from multiple agents planning routes at once.
- Add a central path-planning system with search-limited queue support, and demonstrate the performance benefit and consequences.
- Add path smoothing. (Consider both quick and accurate.)
- Create a tower defence game. Add frag-map history. Get very distracted …
- Add reverse-search as well and compare the time (steps) and the combine search tree cost or benefit.