

PATHFINDING ALGORITHM VISUALIZER V2.0

by Kendrick Phan and Tiffany Mejia

-Visualizes four pathfinding algorithms: Dijkstra's Unweighted Algorithm, A*, DFS, BFS on a grid consisting of a start and end nodes, with barrier nodes

Algorithms:

- Dijkstra's Unweighted Algorithm: Starting from start node, algorithm views neighbors and calculates cost to get to each node. From these neighbors,

algorithm will select the node with the least cost as the next node.

Algorithm repeats until end node is reached.

Since unweighted implementation, each weight is worth 1.

- A* : Starting with the start node, its neighbors are appended to unvisited array of nodes. Their combined Manhattan distance

from start and end is calculated and previous node is set. Next visited node is determined by this distance. The node in the unvisited array with

the least distance is chosen as the next node. This process is repeated until end node is reached. The path is reconstructed

using the previous node attribute.

- DFS : Traverses grid in one direction as far as possible, then changes direction and repeats until end node reached.

- BFS : Starting with the start node, its neighbors are enqueued into a queue. Nodes are dequeued until the end node is reached.

When a node is enqueued, its previous node is updated to its parent.

When a node is dequeued, its Manhattan distance from the start is updated.

When the end node is reached, the shortest path is made starting from the end node, utilizing the previous node attribute.

How to Use:

1. Install pygame through use of pip by opening Command Prompt (for Windows)

Run this command to install : `python3 -m pip install -U pygame --user`

Use this command to test : `python3 -m pygame.examples.aliens`

2. Run main.py

3. First left click will set START NODE

4. Second left click will set END NODE

5. Subsequent left clicks will set WALL NODES

6. To reset a node, right click it

7. Select an algorithm from below

8. Click START BUTTON to commence execution

9. Click RESET BUTTON to clear board before/after execution of an algorithm

10. Red X BUTTON to quit

Example of A*:

