PART 2: THE EFFECTS OF TIES IN REPEATED FORWARD A*

TIE-BREAKING STRATEGIES

In A* search, when multiple cells have the same f-value (f = g + h), a strategy is needed to decide which cell to expand next. Two common strategies are:

1. Favor smaller g-values: Expand cells with smaller g-values first, preferring those closer to the start.
2. Favor larger g-values: Expand cells with larger g-values first, preferring those closer to the goal.

EXPECTED IMPACT

Theoretical Analysis

When A* encounters multiple cells with the same f-value, the choice of which cell to expand can significantly impact search efficiency:

- Favoring smaller g-values tends to explore breadth-first around the start position. This can lead to expanding more cells that are far from the goal.
- Favoring larger g-values tends to explore depth-first toward the goal, which can be more efficient because:
    - It focuses the search toward the goal.
    - It finds a path to the goal more quickly.
    - It reduces the number of cells expanded.

For the example in *Figure 9* from the assignment, when favoring larger g-values, A* will prioritize cells that are closer to the goal among those with the same f-value. This results in a more direct path toward the goal, potentially finding a solution with fewer cell expansions.

IMPLEMENTATION DETAILS

Our implementation uses a dictionary-based priority queue, where:

- States are organized by their f-values.
- For states with the same f-value, selection is based on g-values according to the chosen strategy.
- The number of expanded cells is tracked to measure efficiency.

EXPERIMENTAL RESULTS

After running experiments on multiple maze environments, we observed:

1. Fewer expanded cells when favoring larger g-values.

2. Faster runtime when favoring larger g-values.
3. Similar path lengths for both strategies (since both find optimal paths).

The detailed results and visualizations can be found in the following files:

- tie_breaking_results.txt
- tie_breaking_comparison.png

EXPLANATION OF OBSERVATIONS

Favoring larger g-values is generally more efficient because it guides the search more directly toward the goal. When multiple paths have the same estimated total cost (f-value), choosing the one that has made more progress toward the goal (larger g-value) tends to find a solution faster.

In the context of Repeated Forward A*, this efficiency is even more critical since the algorithm performs multiple A* searches as the agent discovers new information about the environment. Each search must be as efficient as possible to minimize the overall computational cost.

This behavior aligns with the intuition that when faced with multiple equally promising directions, it's better to continue in the direction that has made the most progress toward the goal rather than exploring alternatives that are closer to where you started.