AI Pathfinder Report

1. Explanation of Search Strategy Implementation

Breadth-First Search (BFS)

BFS is implemented using a queue (deque) to explore the grid level by level, starting from the start node. Each node's neighbors are generated in a fixed six-direction order as defined in the assignment. A visited set ensures nodes are not revisited. BFS guarantees the shortest path in terms of number of steps because it expands nodes uniformly by depth.

Depth-First Search (DFS)

DFS is implemented using an explicit stack. The algorithm explores one path deeply before backtracking. To avoid infinite loops and excessively long paths, nodes are marked as visited immediately when pushed onto the stack. Neighbors are added in reverse order to maintain consistent directional priority during exploration.

Uniform Cost Search (UCS)

UCS is implemented using a priority queue (heapq) ordered by cumulative path cost. Movement costs account for straight and diagonal movement. The algorithm always expands the lowest-cost node first, ensuring optimal paths even when movement costs vary. A cost map is maintained to update paths efficiently.

Depth-Limited Search (DLS)

DLS is implemented as a recursive depth-first search with a predefined depth limit. Each recursive call tracks the current depth and stops expanding nodes once the limit is reached. This prevents infinite exploration in deep or cyclic grids but may fail to find a solution if the limit is too low.

Iterative Deepening Depth-First Search (IDDFS)

IDDFS repeatedly runs DLS with increasing depth limits. Each iteration clears previous search data and restarts from the initial node. This approach combines DFS's low memory usage with BFS's completeness, ensuring a solution is found if one exists within the maximum depth.

Bidirectional Search

Bidirectional Search runs two BFS processes simultaneously: one from the start and one from the target. The search terminates when the two frontiers meet. The final path is reconstructed by combining the forward and backward paths. This significantly reduces the number of explored nodes compared to standard BFS.

---

2. Pros and Cons of Each Algorithm

Breadth-First Search (BFS) Pros

- Guarantees shortest path in unweighted grids
- Simple and reliable behavior

Cons

- High memory consumption
- Slower on large grids

Depth-First Search (DFS)

Pros

- Low memory usage
- Can reach deep solutions quickly

Cons

- Does not guarantee optimal paths
- Performance highly depends on grid layout

Uniform Cost Search (UCS)

Pros

- Guarantees optimal paths
- Handles different movement costs correctly **Cons**
- Slower than BFS in simple grids
- Higher computational overhead

Depth-Limited Search (DLS)

Pros

- Prevents infinite depth exploration
- Efficient when depth limit is appropriate **Cons**
- Fails if depth limit is too small
- Not optimal

Iterative Deepening DFS (IDDFS)

Pros

- Complete and memory-efficient
- Combines benefits of BFS and DFS

Cons

- Repeated exploration increases runtime
- Slower than BFS in shallow solutions

Bidirectional Search

Pros

- Explores significantly fewer nodes
- Faster than BFS in large grids

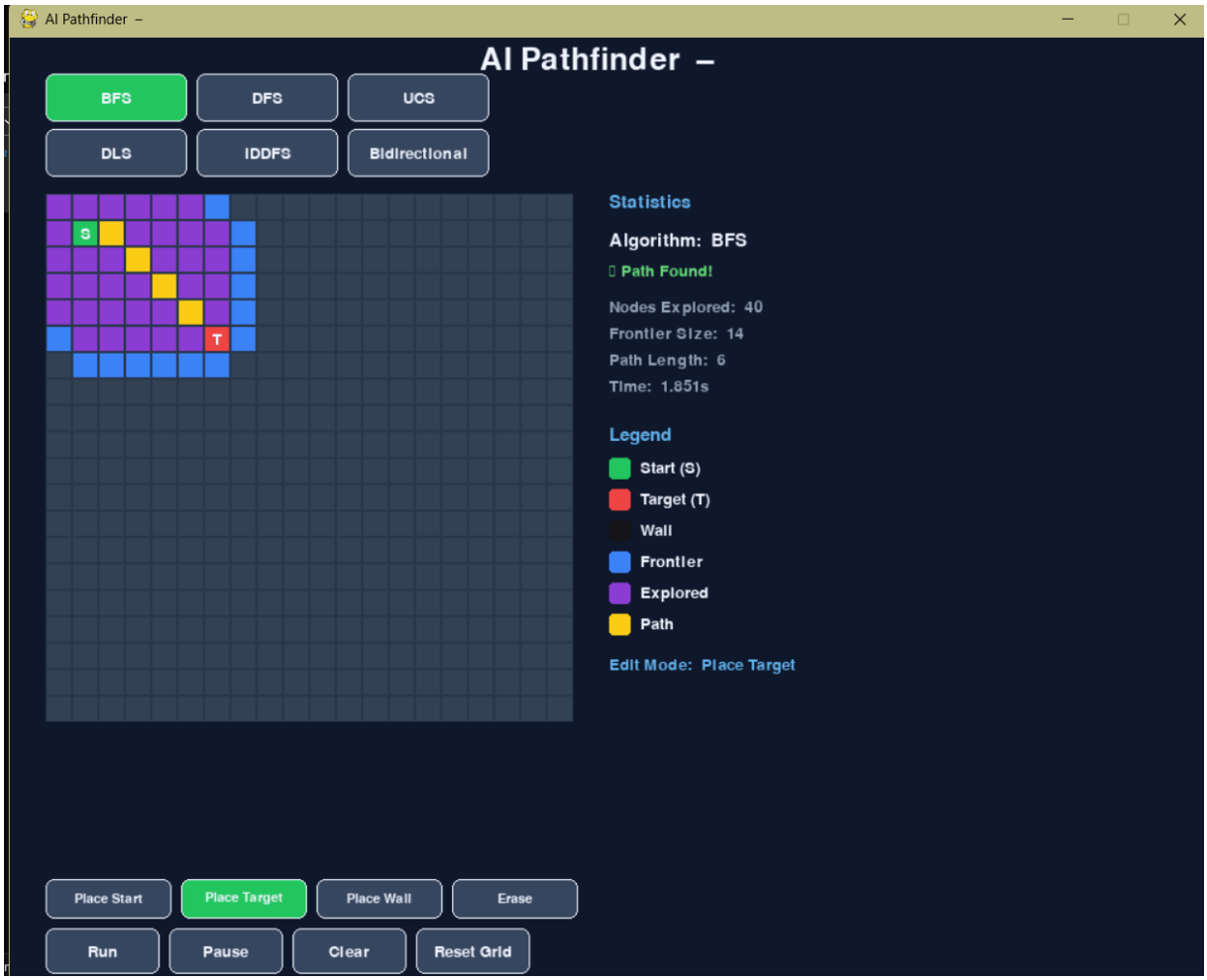
Cons

- More complex implementation
- Requires efficient path merging

BFS:

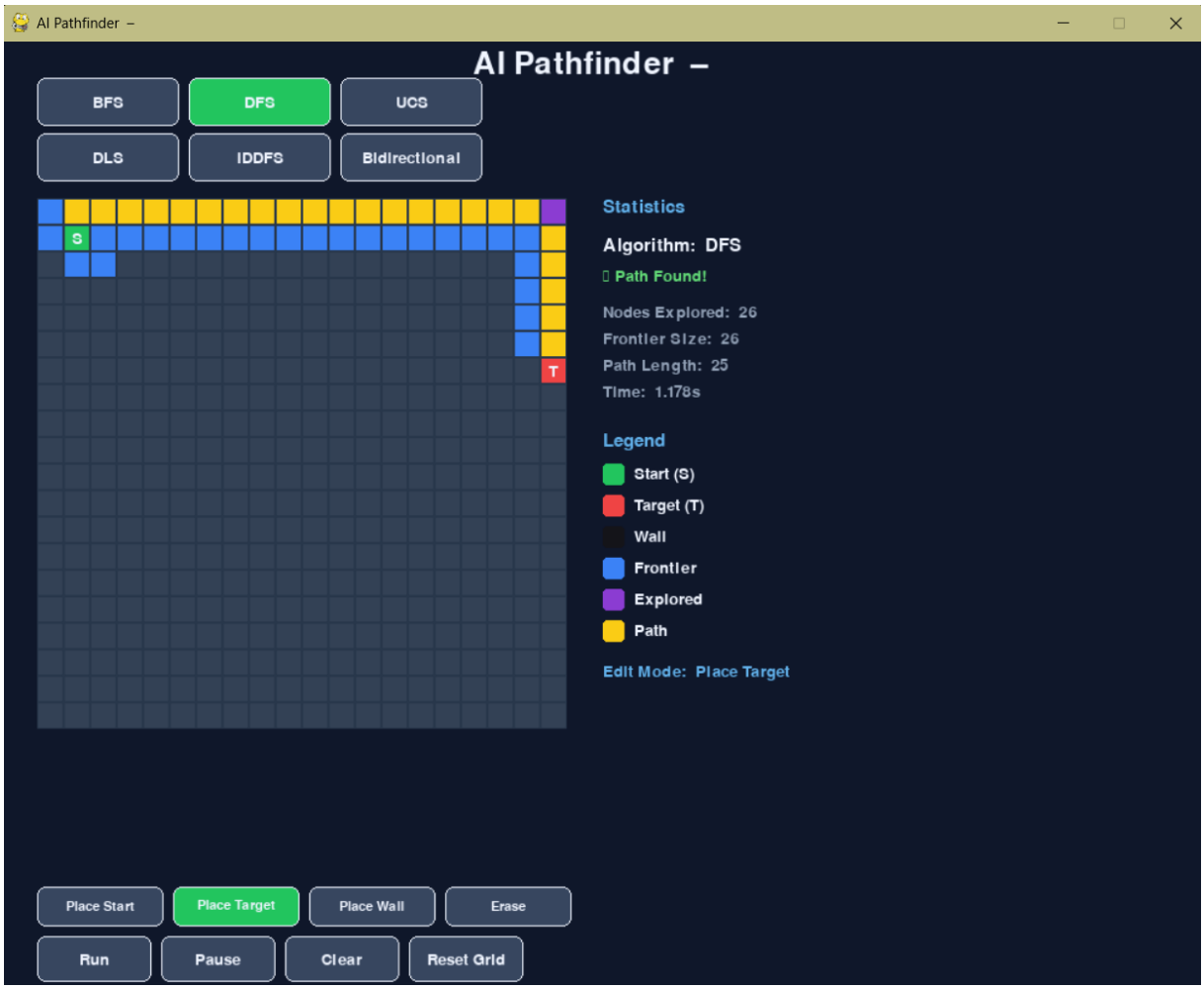
Best Case:


AI Pathfinder –
AI Pathfinder –
BFS
DFS
UCS
DLS
IDDFS
Bidirectional
S
T
Statistics
Algorithm: BFS
Path Found!
Nodes Explored: 359
Frontier Size: 39
Path Length: 18
Time: 17.233s
Legend
Start (S)
Target (T)
Wall
Frontier
Explored
Path
Edit Mode: Place Wall
Place Start
Place Target
Place Wall
Erase
Run
Pause
Clear
Reset Grid

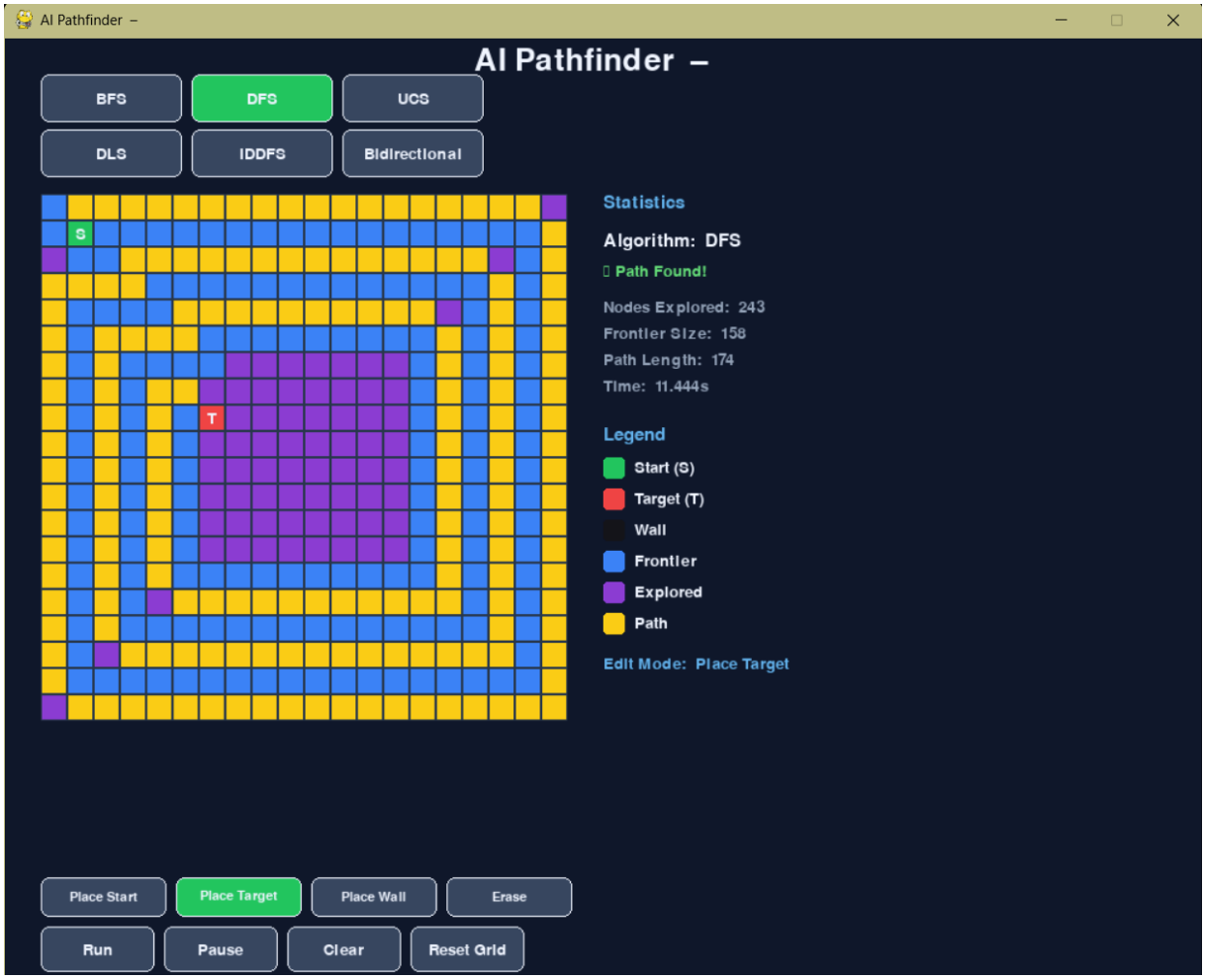

AI Pathfinder –
AI Pathfinder –
BFS
DFS
UCS
DLS
IDDFS
Bidirectional
S
T
Statistics
Algorithm: BFS
Path Found!
Nodes Explored: 40
Frontier Size: 14
Path Length: 6
Time: 1.851s
Legend
Start (S)
Target (T)
Wall
Frontier
Explored
Path
Edit Mode: Place Target
Place Start
Place Target
Place Wall
Erase
Run
Pause
Clear
Reset Grid

DFS:

Best Case:

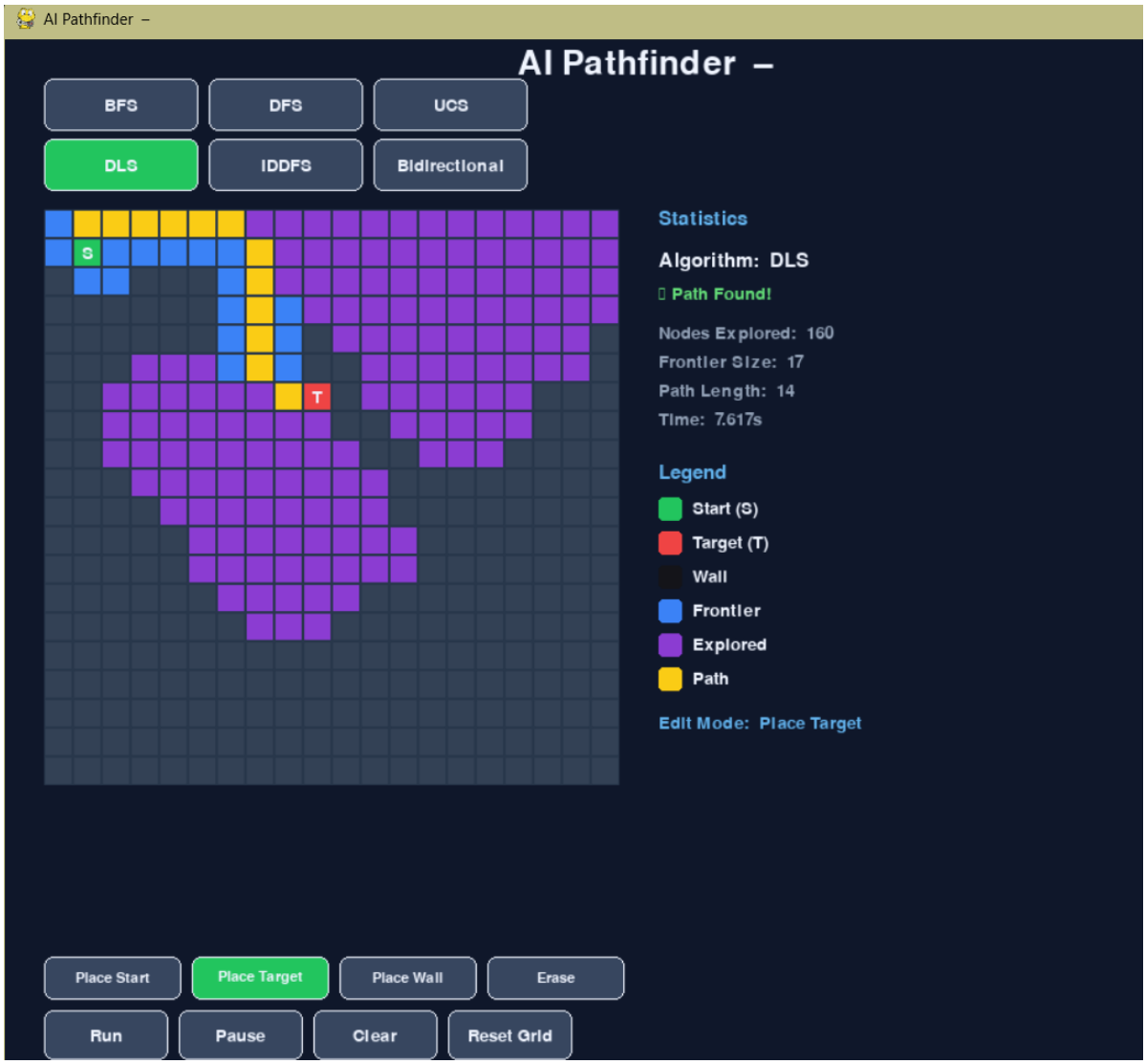

AI Pathfinder –
AI Pathfinder –
BFS
DFS
UCS
DLS
IDDFS
Bidirectional
S
T
Statistics
Algorithm: DFS
Path Found!
Nodes Explored: 26
Frontier Size: 26
Path Length: 25
Time: 1.178s
Legend
Start (S)
Target (T)
Wall
Frontier
Explored
Path
Edit Mode: Place Target
Place Start
Place Target
Place Wall
Erase
Run
Pause
Clear
Reset Grid

Worst Case

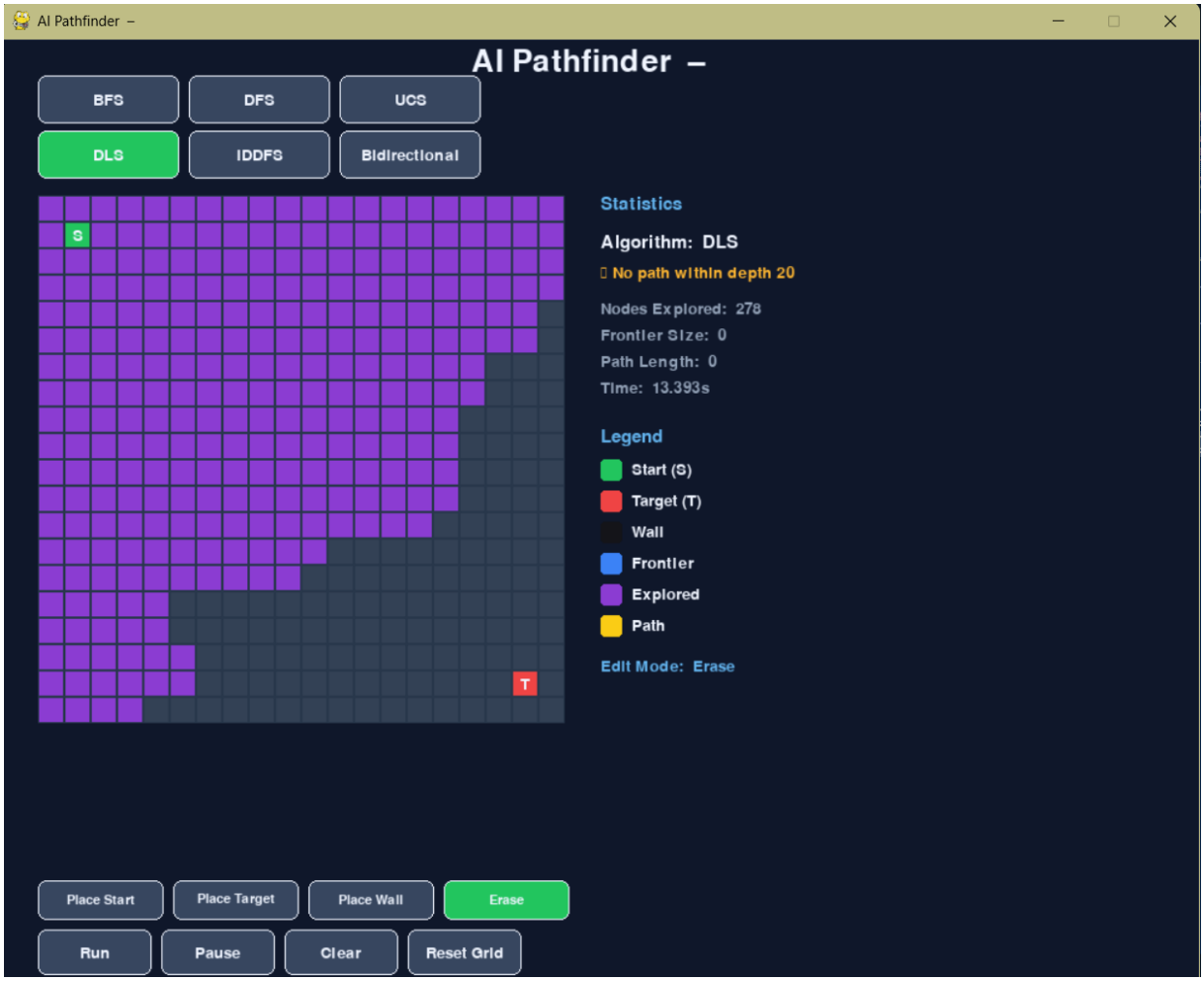

AI Pathfinder –
AI Pathfinder –
BFS
DFS
UCS
DLS
IDDFS
Bidirectional
S
T
Statistics
Algorithm: DFS
Path Found!
Nodes Explored: 243
Frontier Size: 158
Path Length: 174
Time: 11.444s
Legend
Start (S)
Target (T)
Wall
Frontier
Explored
Path
Edit Mode: Place Target
Place Start
Place Target
Place Wall
Erase
Run
Pause
Clear
Reset Grid

DLS:

Best Case:


AI Pathfinder –
AI Pathfinder –
BFS
DFS
UCS
DLS
IDDFS
Bidirectional
S
T
Statistics
Algorithm: DLS
Path Found!
Nodes Explored: 160
Frontier Size: 17
Path Length: 14
Time: 7.617s
Legend
Start (S)
Target (T)
Wall
Frontier
Explored
Path
Edit Mode: Place Target
Place Start
Place Target
Place Wall
Erase
Run
Pause
Clear
Reset Grid

Worst Case


AI Pathfinder –
AI Pathfinder –
BFS
DFS
UCS
DLS
IDDFS
Bidirectional
S
T
Statistics
Algorithm: DLS
No path within depth 20
Nodes Explored: 278
Frontier Size: 0
Path Length: 0
Time: 13.393s
Legend
Start (S)
Target (T)
Wall
Frontier
Explored
Path
Edit Mode: Erase
Place Start
Place Target
Place Wall
Erase
Run
Pause
Clear
Reset Grid

UCS:

Best Case:

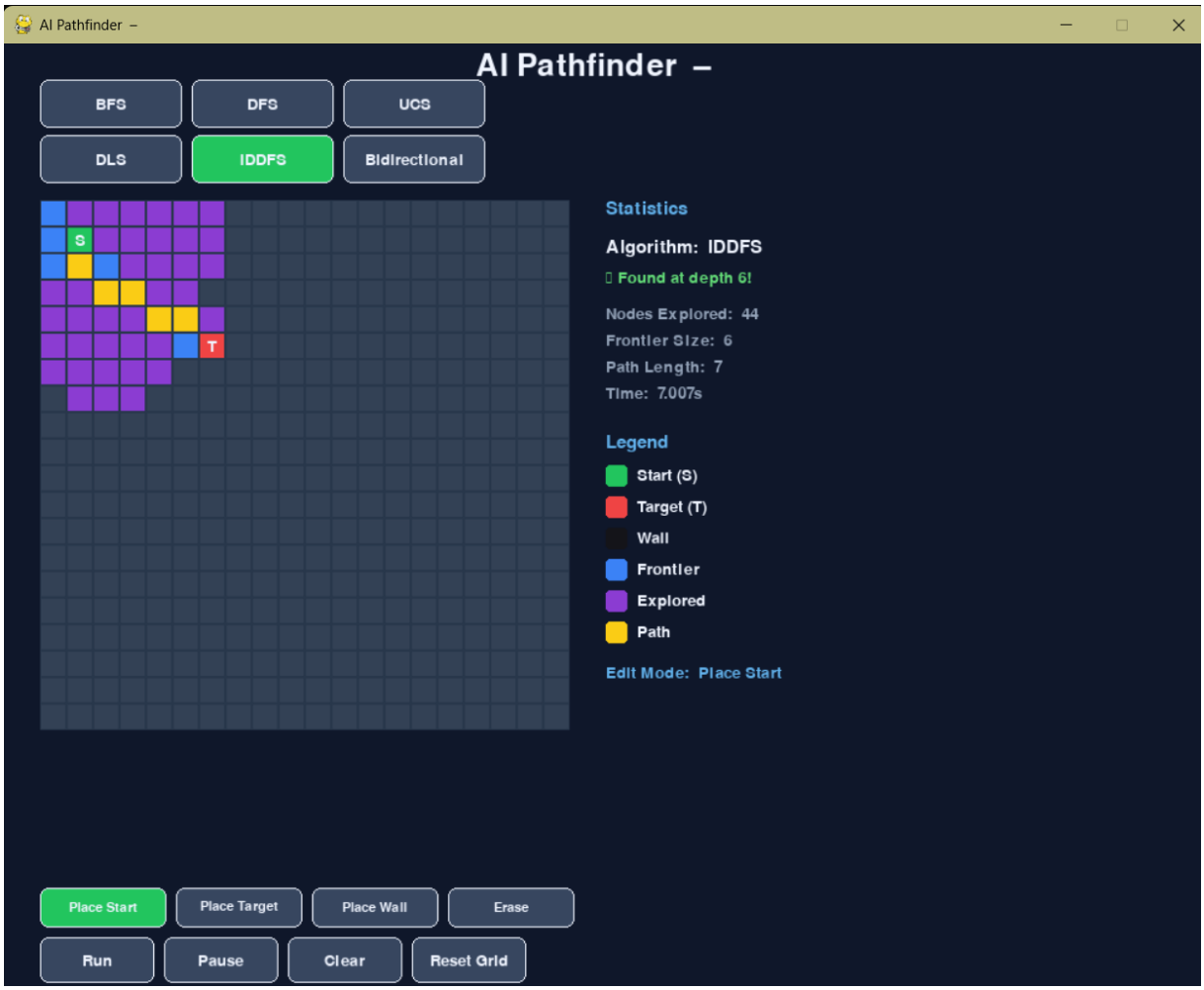

AI Pathfinder –
AI Pathfinder –
BFS
DFS
UCS
DLS
IDDFS
Bidirectional
S
T
Statistics
Algorithm: UCS
Path Found!
Nodes Explored: 35
Frontier Size: 14
Path Length: 6
Time: 1.626s
Legend
Start (S)
Target (T)
Wall
Frontier
Explored
Path
Edit Mode: Place Start
Place Start
Place Target
Place Wall
Erase
Run
Pause
Clear
Reset Grid

Worst Case

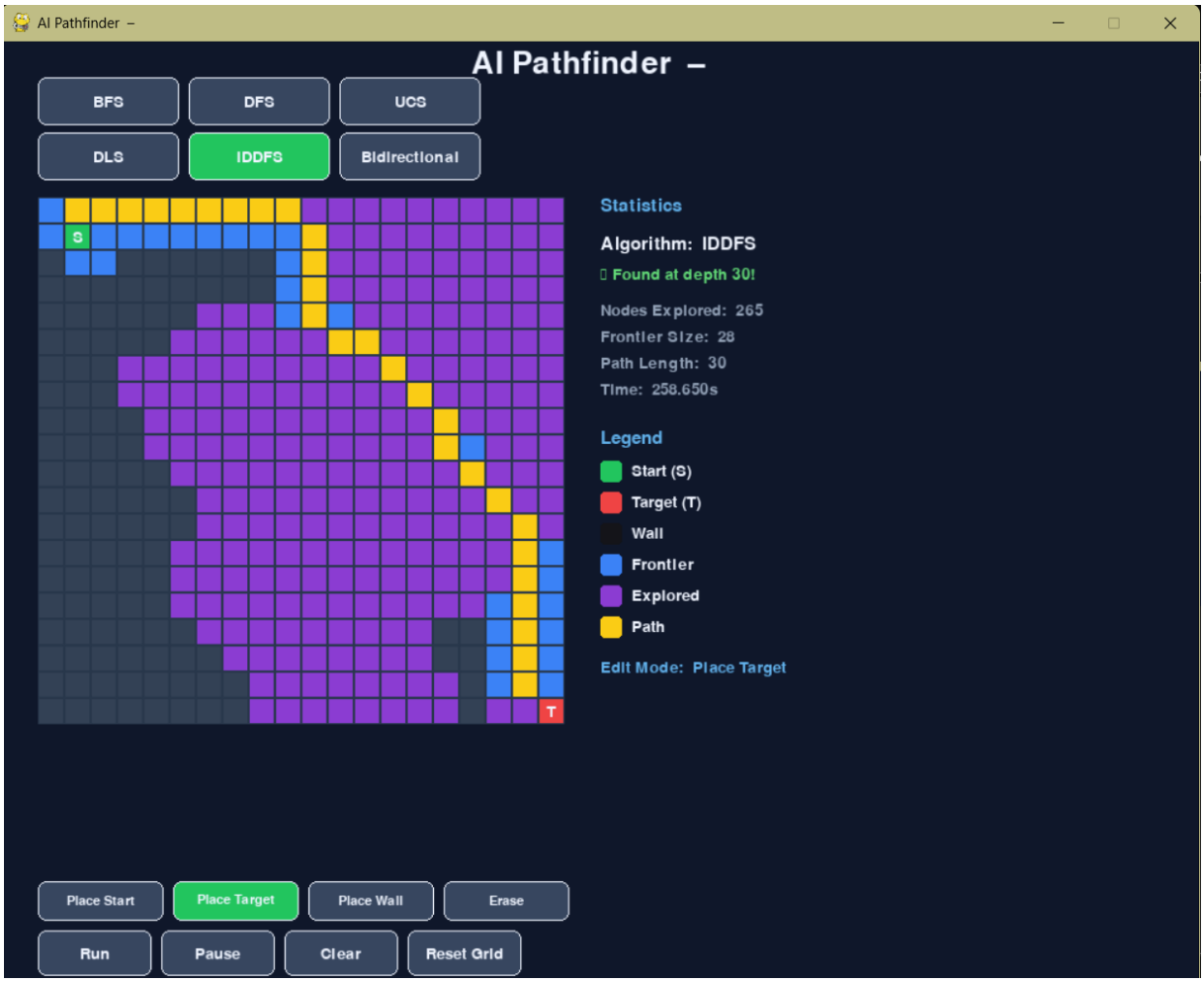

AI Pathfinder –
AI Pathfinder –
BFS
DFS
UCS
DLS
IDDFS
Bidirectional
S
T
Statistics
Algorithm: UCS
Path Found!
Nodes Explored: 393
Frontier Size: 7
Path Length: 18
Time: 18.615s
Legend
Start (S)
Target (T)
Wall
Frontier
Explored
Path
Edit Mode: Place Target
Place Start
Place Target
Place Wall
Erase
Run
Pause
Clear
Reset Grid

IDDFS:

Best Case:

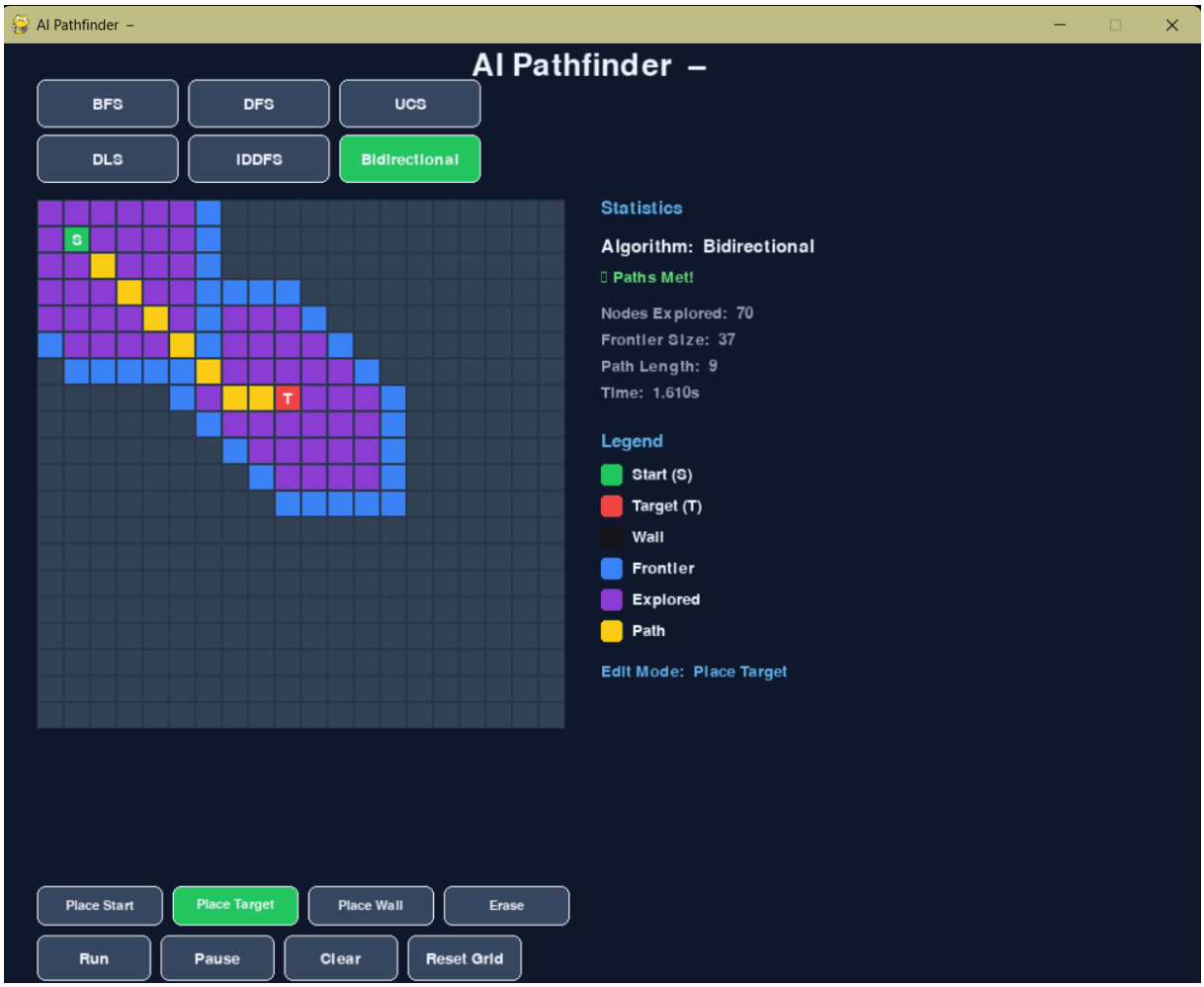

AI Pathfinder –
AI Pathfinder –
BFS
DFS
UCS
DLS
IDDFS
Bidirectional
S
T
Statistics
Algorithm: IDDFS
Found at depth 6!
Nodes Explored: 44
Frontier Size: 6
Path Length: 7
Time: 7.007s
Legend
Start (S)
Target (T)
Wall
Frontier
Explored
Path
Edit Mode: Place Start
Place Start
Place Target
Place Wall
Erase
Run
Pause
Clear
Reset Grid

Worst Case

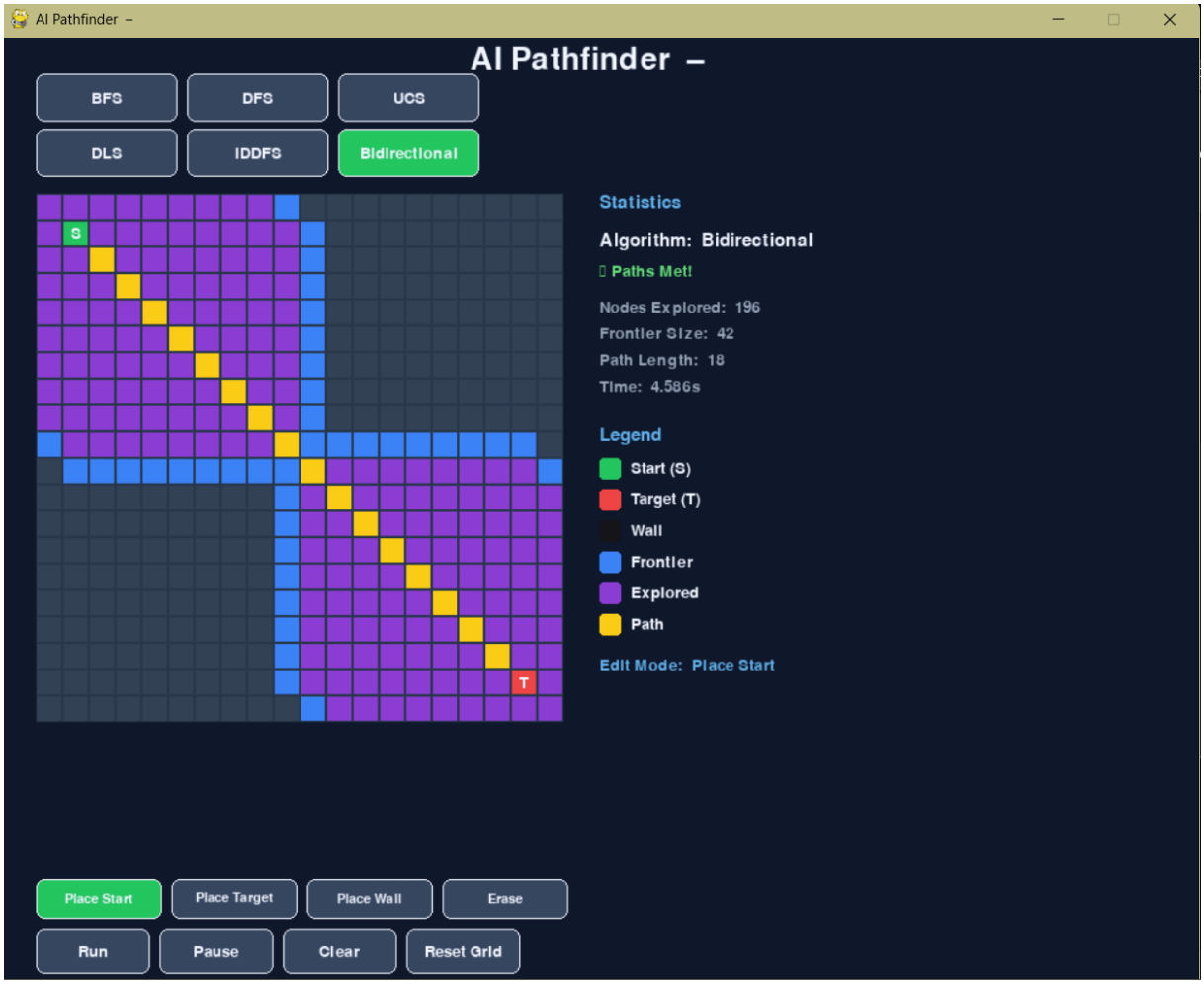

AI Pathfinder –
AI Pathfinder –
BFS
DFS
UCS
DLS
IDDFS
Bidirectional
S
T
Statistics
Algorithm: IDDFS
Found at depth 30!
Nodes Explored: 265
Frontier Size: 28
Path Length: 30
Time: 258.650s
Legend
Start (S)
Target (T)
Wall
Frontier
Explored
Path
Edit Mode: Place Target
Place Start
Place Target
Place Wall
Erase
Run
Pause
Clear
Reset Grid

Bidirectional:

Best Case:


AI Pathfinder –
AI Pathfinder –
BFS
DFS
UCS
DLS
IDDFS
Bidirectional
S
T
Statistics
Algorithm: Bidirectional
Paths Met!
Nodes Explored: 70
Frontier Size: 37
Path Length: 9
Time: 1.610s
Legend
Start (S)
Target (T)
Wall
Frontier
Explored
Path
Edit Mode: Place Target
Place Start
Place Target
Place Wall
Erase
Run
Pause
Clear
Reset Grid

Worst Case


AI Pathfinder –
AI Pathfinder –
BFS
DFS
UCS
DLS
IDDFS
Bidirectional
S
T
Statistics
Algorithm: Bidirectional
 Paths Met!
Nodes Explored: 196
Frontier Size: 42
Path Length: 18
Time: 4.586s
Legend
Start (S)
Target (T)
Wall
Frontier
Explored
Path
Edit Mode: Place Start
Place Start
Place Target
Place Wall
Erase
Run
Pause
Clear
Reset Grid

Link to Github:

https://github.com/Ali-ilA-irf/AI_Pathfinder