RAJALAKSHMI INSTITUTE OF TECHNOLOGY
BELIEVE IN THE POSSIBILITIES
AN AUTONOMOUS INSTITUTION

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE ACADEMIC YEAR 2025 - 2026 SEMESTER III ARTIFICIAL INTELLIGENCE LABORATORY

MINI PROJECT REVIEW

" Maze Solver "

REGISTER NUMBER	
NAME	Subina Lakshmi M
YEAR	II
SECTION	AI&DS-F
FACULTY IN-CHARGE	Mrs. S. Divya

PROBLEM STATEMENT

Problem Statement

Artificial Intelligence (AI) helps systems solve complex problems using intelligent search and decision-making.
This project, “Maze Solver using Depth-First Search (DFS) Algorithm),” focuses on developing a program that finds a valid path from a start point to a destination in a maze using DFS traversal and backtracking.

Why It Matters:
Maze solving reflects real-world AI challenges such as robot navigation, game pathfinding, and autonomous planning.

Expected Result:
A Python-based GUI dashboard that visually demonstrates the DFS process and highlights the discovered path.

THEORETICAL BACKGROUND

Theoretical Background

The Depth-First Search (DFS) algorithm is a key AI search technique used to explore paths in a maze. It goes deep along a path before backtracking, making it suitable for maze-solving applications.

Other algorithms like BFS, A* and Dijkstra's can also solve pathfinding problems, but DFS is simple, efficient, and easy to visualize.

Justification:
DFS is chosen for its simplicity, low memory use, and ability to clearly demonstrate recursive exploration.

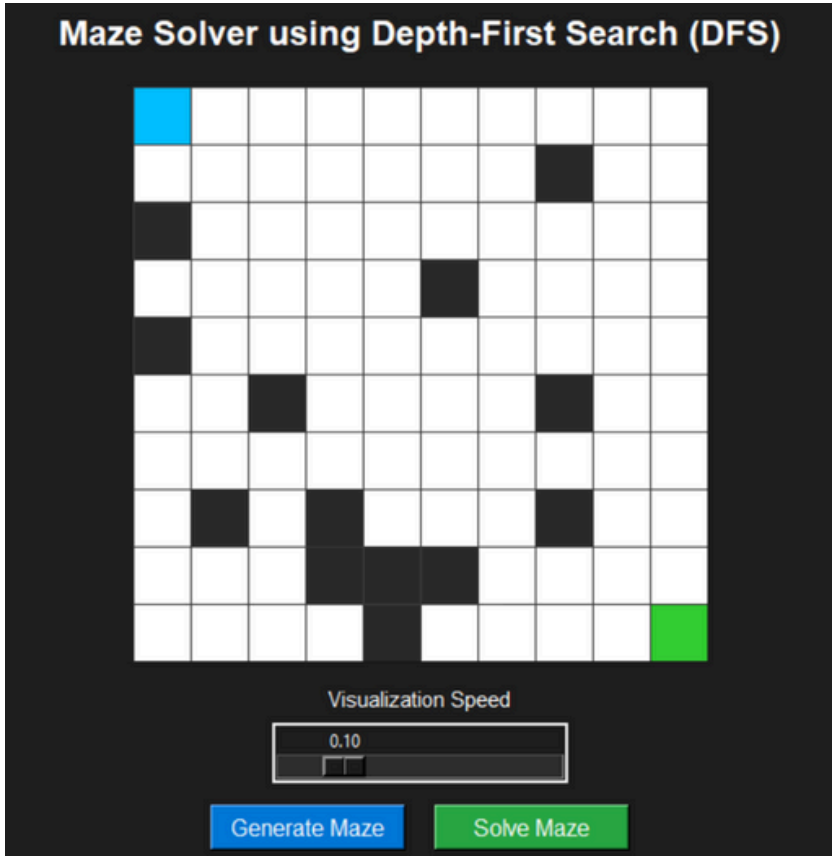
Example:
DFS starts at the source cell, explores neighboring cells, and backtracks from dead ends until it reaches the goal.

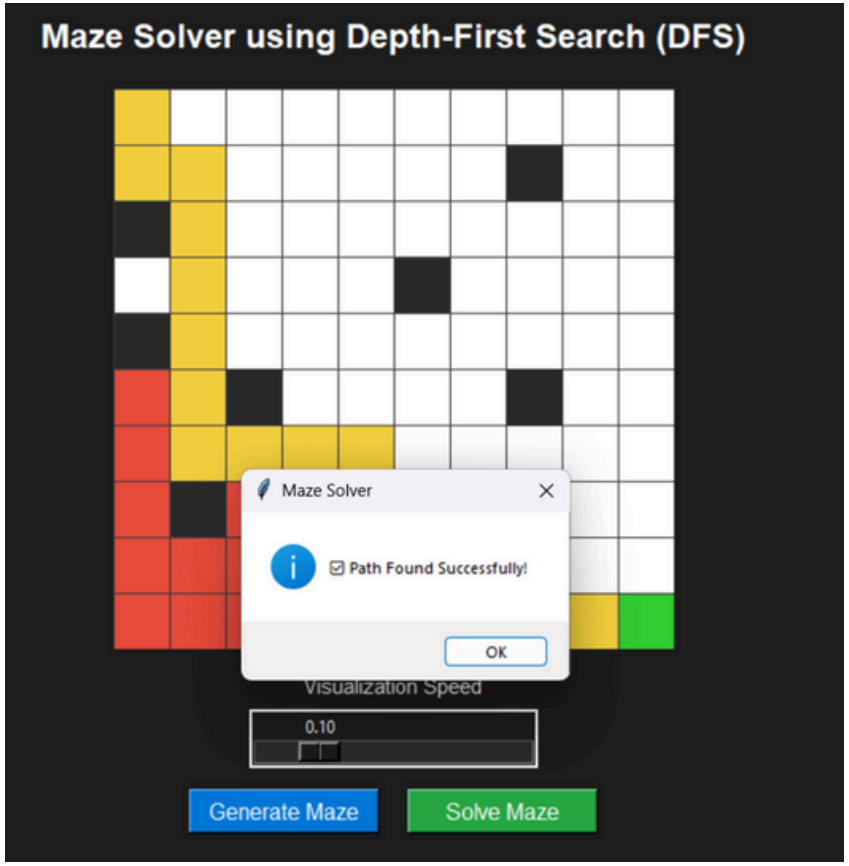
IMPLEMENTATION AND CODE

- https://github.com/subina0987/Maze-Solver-DFS

List	Git-hub Repository Links
Implementation of Code Link	**https://github.com/subina0987/Maze-Solver-DFS**
Word Document Report Link	**https://github.com/subina0987/Maze-Solver-DFS/blob/main/Maze-Solver-Report.docs.docx**
PPT Link	**https://github.com/subina0987/Maze-Solver-DFS/blob/main/Maze-Solver-Presentation.pptx**

OUTPUT AND RESULTS





The Maze Solver displays the maze in a graphical grid.

The DFS algorithm explores paths from the start to the goal recursively.

Once the goal is reached, the correct path is shown, highlighting how the algorithm works step by step.

OUTPUT AND RESULTS

Results:

The system successfully solves mazes using the Depth-First Search algorithm.

It accurately explores paths, applies backtracking when necessary, and finds a valid route to the goal.

The interface visually demonstrates the DFS logic and provides an interactive AI-based learning experience.

Future Enhancements:

- Add visual path animation for step-by-step traversal.
- Allow users to design or upload custom maze structures.
- Implement other algorithms such as BFS, A*, or Dijkstra for comparison.
- Develop a web-based or mobile version for wider accessibility.

REFERENCES

1. Stuart Russell & Peter Norvig, "Artificial Intelligence: A Modern Approach"

2. Thomas H. Cormen et al., "Introduction to Algorithms"

3. GeeksforGeeks: "Depth-First Search Algorithm in Python"

4. Python Official Documentation: https://docs.python.org/

5. Towards Data Science: "Exploring Pathfinding Algorithms using DFS and BFS"