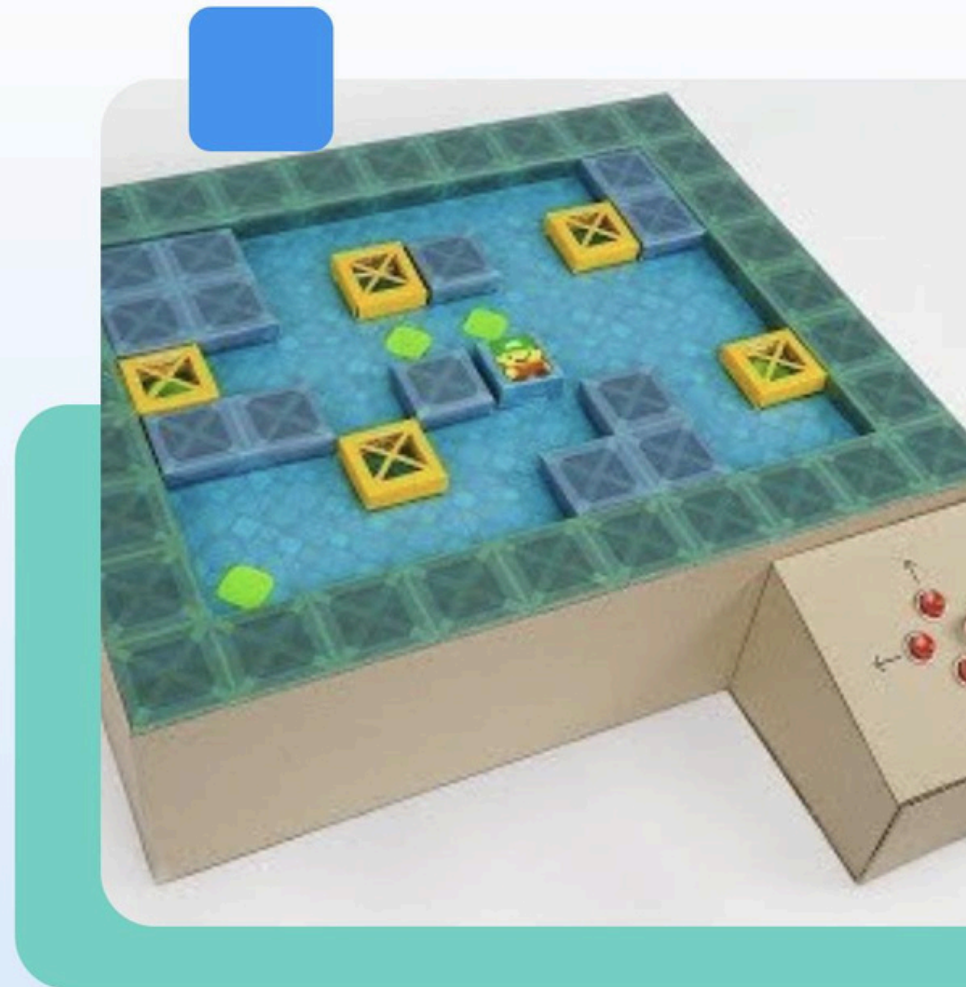


# Solving Sokoban Game By AI Techniques

Exploring various artificial intelligence strategies and algorithms for effectively solving Sokoban game scenarios and their impact on gameplay.

## Group Members :

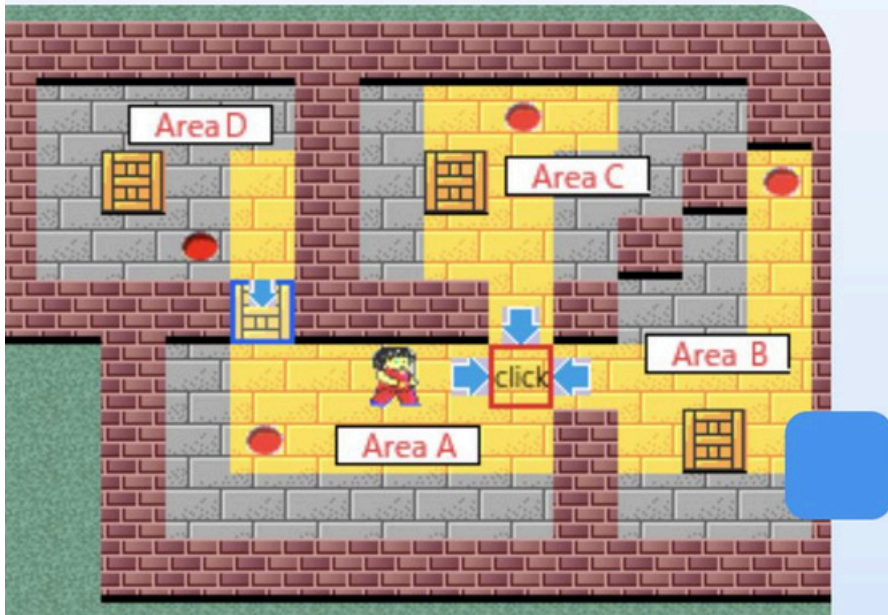
B.Aadi Narayana	- AIE23027
B.Sai Siddartha Vyas	- AIE23028
L.Shanmukh	- AIE23045\
P.Vivek Kumar Reddy	- AIE23050



## SOKOBAN AND AI

# Introduction to Sokoban and AI Techniques

Understanding the Game and Its Solutions



## Overview of Sokoban

01

Sokoban is a strategic puzzle game where a player pushes boxes onto designated target locations. The aim is to position all boxes correctly while minimizing the number of moves.

## Common AI Techniques for Problem-Solving

02

AI techniques such as search algorithms, heuristics can automate the process of solving Sokoban puzzles, enhancing efficiency and accuracy.

## Challenges and Limitations

03


Sokoban puzzles present various challenges, including state space complexity and deadlock situations, etc.

## Conclusion and Future Prospects

04

The intersection of Sokoban and AI presents opportunities for future research, including improved algorithms and exploring new AI methodologies for enhanced puzzle-solving.

# Overview of sokoban

 Sokoban is a classic puzzle video game where players push boxes into designated locations .



## Game Objective :

The main objective is to move all boxes to their respective goals. Players can only push one box at a time and cannot pull boxes or move through walls, making strategic planning essential.

## Analyze the Level :

study the layout of the maze and identify where each box needs to be moved to reach the goals effectively.

## Plan Your Moves :

Before making any moves, determine box positions strategically to avoid dead ends.

## Execute with Precision :

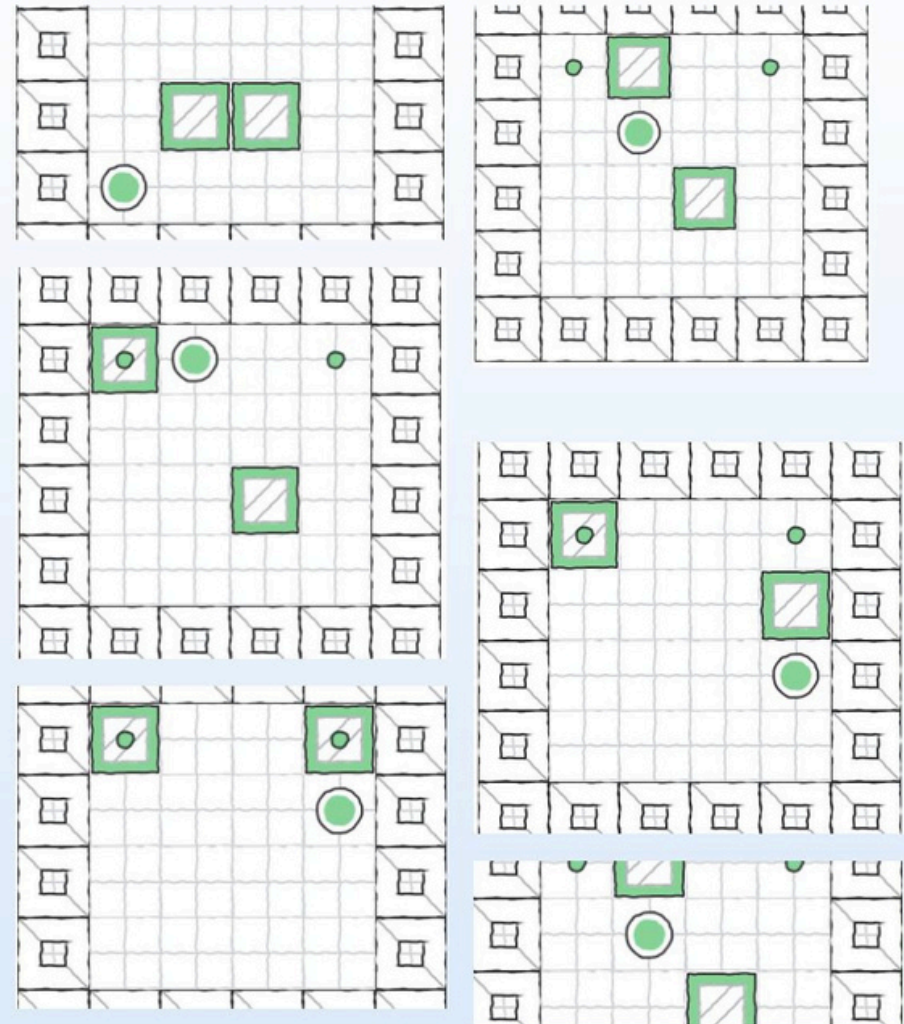
Push boxes carefully, moving them to the goals in a logical sequence, and use backtracking if we encounter obstacles.

## if I'm completely stuck?

If you're stuck, consider restarting the level with a new strategy.

## Conclusion :

Winning at Sokoban requires strategic thinking and careful planning. Following these steps enhances puzzle-solving skills.





## SOKOBAN AI CHALLENGES

# Challenges and Limitations

Key Issues and Solutions



### Deadlocks

Code lacks advanced logic to identify and avoid deadlock states, leading to wasted computations.



### Heuristic Limitations

Many heuristic functions may fail to provide accurate estimates of the true cost to reach the goal, hindering optimal pathfinding.



### State Explosion

Large state space can lead to high memory usage and longer computation times for bigger puzzles.



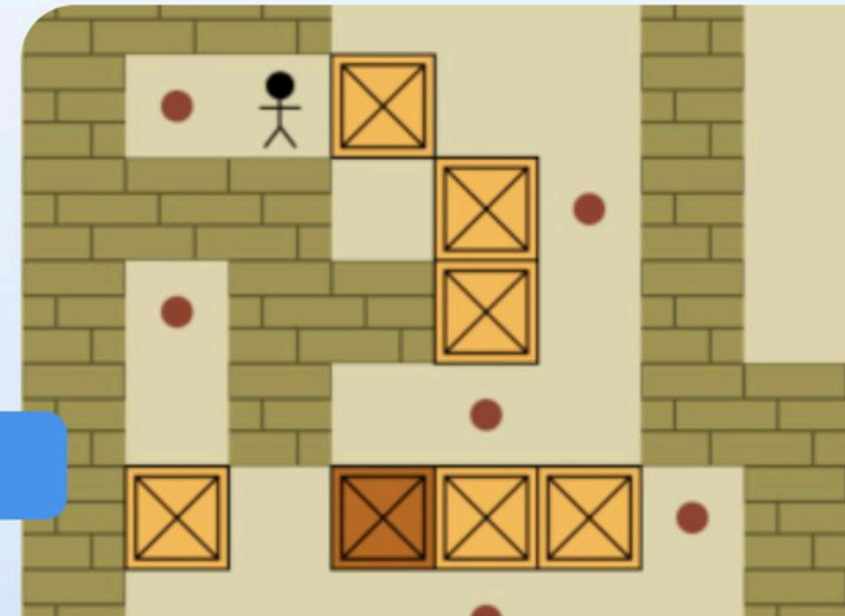
### Limited Scalability

High computational cost limits use for larger or more complex levels.



### Handling Multiple Boxes

Large state space The algorithm struggles with puzzles involving tight spaces and multiple boxes blocking each other, and lead to high memory usage and longer computation times for bigger puzzles.



# Techniques and Algorithms

- A\*

- > Finds the optimal path using  $f = g + h$  (cost + heuristic).

- Constraint Satisfaction

- > Ensures boxes are only moved into valid spaces

- State-Space Exploration

- > Explores all possible board configurations resulting from valid moves

- Board Representation

- > 2D grid with '#' (wall), '.' (empty), 'B' (box), 'G' (goal), 'P' (player).

- Valid Move Logic

- > Checks bounds, walls, and valid positions for boxes and player.

- Path Tracking

- > Records the sequence of player moves leading to the solution.

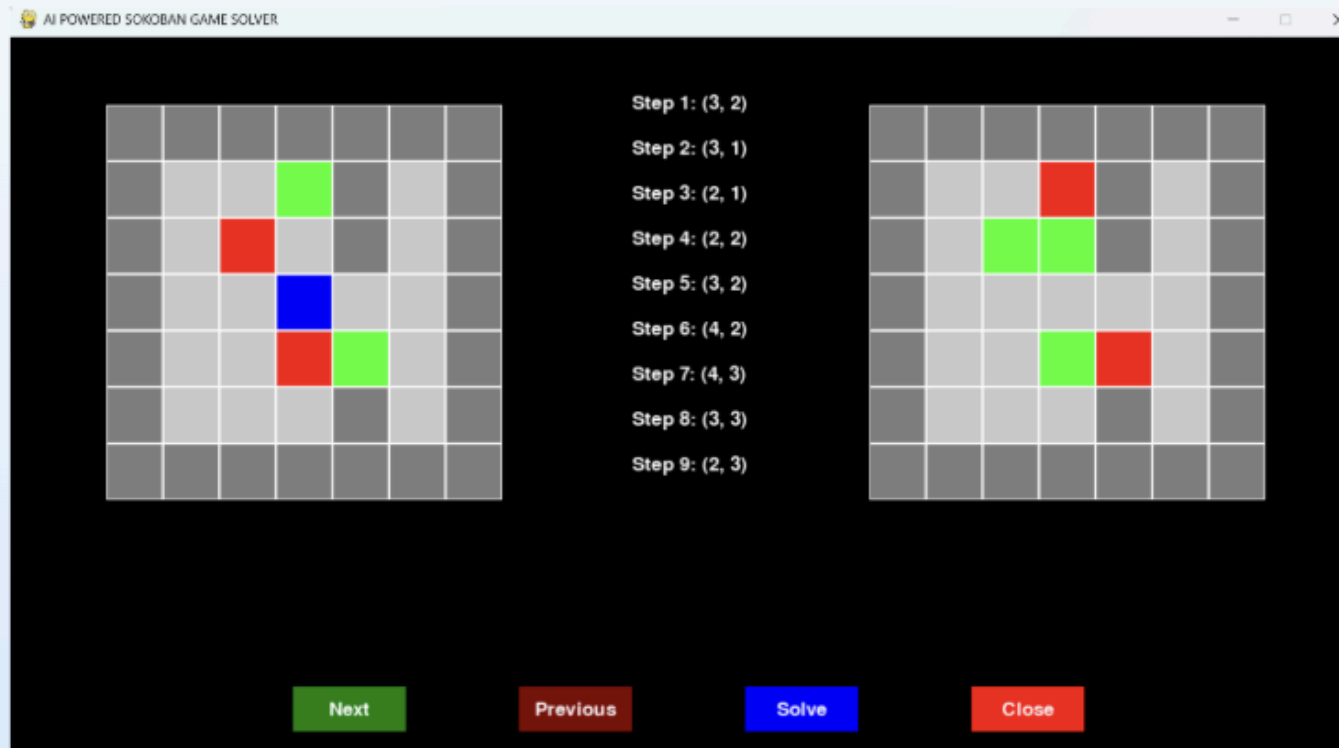
- Dynamic Visualization

- > By Pygame we are representing screen for result.

# Algorithm And Output Of Project



The A\* (A-star) algorithm is particularly efficient for solving Sokoban and similar pathfinding or optimization problems because of its combination of search strategies and the use of a heuristic function. Here's why A\* is often considered efficient compared to other algorithms



# Conclusion and Future Prospects

Summary of Insights and Directions for Advanced AI in Puzzle Solving



01

## Future Prospect

**Enhanced Heuristics:** Implement smarter heuristics to better handle complex puzzles.

02

## Future Prospect

**Algorithm Approach:** Combine A\* with BFS/DFS for flexibility

03

## Future Prospect

**Scalability Improvements :** Optimize memory usage and speed for larger levels.

04

## Conclusion

The Sokoban solver demonstrates the effective use of A search\* and constraint satisfaction for solving logical puzzles.

05

## Conclusion

It provides dynamic visualization and interactive controls, making solutions understandable and engaging.

06

## Conclusion

While efficient for small to medium-sized puzzles, it faces challenges in scalability and advanced deadlock detection.

