# Instruction

# Introduction

The environment in the Robot Navigation problem is a N x M grid, with N and M both bigger than one. In addition, a number of occupied wall cells are inaccessible. The robot starts off in one of the cells and has to find a way to one of the given destinations by avoiding the barriers. We may utilise a variety of search algorithms, including Depth-first and Breadth-first search, to identify the best course of action in solving this problem. As a result, we will examine many popular algorithms that have been applied to the programme in this study and contrast them according to their memory usage and complexity.

## Glossary

* BFS: Breadth-first Search Algorithm
* DFS: Depth-first Search Algorithm
* A\*: A\* Search Algorithm
* GBFS: Greedy Best-First Search Algorithm
* IDA: Iterative Deepening A\* Search Algorithm
* stack: a data structure that stores elements in a last-in-first-out fashion
* queue: a data structure that stores elements in a first-in-first-out fashion.

# Search Algorithms

This research will compare six search algorithms for efficiency and memory utilisation. There will be an introduction to greedy best-first search, bidirectional search, bidirectional A\* search, depth-first search (DFS), and breadth-first search (BFS).

## Uninformed Search Algorithms

Uninformed search algorithms are basic algorithms that apply a brute-force method to identify a solution in a search tree, without leveraging any domain-specific information to direct the search. Some common uninformed algorithms include breadth-first search, depth-first search, and bidirectional search:

1. **Breadth-first search (BFS)**

BFS is a graph traversal method that searches nodes in a level-by-level manner, beginning with the root node. The approach starts with the shallowest, least-explored nodes, ensuring that all nodes at a given depth are inspected before going on to nodes at the next depth level. This attribute makes BFS appropriate for determining the shortest path between the root node and the solution, providing that all actions have equal weights.

1. **Depth-first search (DFS)**
2. **Bidirectional search**

## Informed Search Algorithms

# Implementation

# Testing

# Features/Bugs

# Research

# Conclusion

# Acknowledgements/Resources

# References