Conclusion on the Data Structure and Algorithms Chosen

For solving the maze problem, we opted for graphs as our primary data structure. This decision was driven by the simplicity and efficiency that graphs provide when representing complex structures like mazes. The choice of graphs allowed for intuitive management of nodes and edges, which mapped to the corridors and paths within the maze. Additionally, graphs offered a better performance profile compared to alternative data structures such as matrices or lists, where the complexity and memory overhead could become significant.

Algorithm Selection and Performance Insights

Initially, we selected the Breadth-First Search (BFS) algorithm to traverse and solve the maze due to its proven effectiveness in finding the shortest path in unweighted graphs.

We also experimented with an original approach we named the "Bottom-Right Algorithm," which aimed to leverage specific properties of the maze's layout to find solutions. However, after several paper-based tests, we found that it fell short in performance compared to BFS. The Bottom-Right Algorithm showed higher time complexity in practical scenarios and did not consistently yield efficient results, reinforcing the reliability of BFS.

Lessons Learned and Remaining Questions

From solving this problem, we have gained a deeper understanding of how theoretical complexities translate into real-world performance. The experience highlighted the importance of selecting algorithms that balance simplicity, performance, and applicability to the problem at hand.

However, there are still questions that we would like to discuss further during the restructuring lecture:

- How can we better anticipate the divergence between theoretical and empirical performance when designing algorithms?
- … (to be continued)

These questions would help refine our understanding of algorithm design and guide better decision-making in future projects.