Choose one of the following projects.

Project-1
In this project, you will develop a **Traffic Sign Detection System** using computer vision models, applying the full machine learning algorithms. Using the provided dataset (https://www.kaggle.com/datasets/pkdarabi/cardetection/data), you are required to implement and compare two different image segmentation models (for example, U-Net, Mask R-CNN, etc.). Your analysis must follow standard ML principles, including data preprocessing, train/test splitting, k-fold cross-validation, model selection, hyperparameter tuning, and proper evaluation using metrics such as IoU, precision, recall, and F1 score. (Note: The instructor will provide any additional guidance necessary for this project, as much of the content has not yet been covered in class.)

Project-2
Create an AI agent to play the **2048 Classic Game** (https://classic.play2048.co/) using any two appropriate search algorithms eg. Minimax/Alpha-Beta, and compare their performance. A presentable UI/interface is expected.
Reference: https://medium.com/data-science/a-puzzle-for-ai-eb7a3cb8e599

Project-3
You will design a **New York City Route Planner** where ~20-30 major cities in New York State (e.g., Rochester, Buffalo, Syracuse, Albany, Ithaca, Binghamton, Niagara Falls, New York City, etc.) are represented as nodes in a graph. The edges between cities will be weighted by actual road distances (miles). The heuristic cost for informed search will be the straight-line distance between city coordinates (compute using the Haversine formula or any other alternatives). The start node is Rochester (RIT), and the user will choose the destination city. You should implement DFS, BFS, IDS, UCS, Greedy Best-First Search, A*, and IDA* search to find paths from Rochester to the chosen destination. For each algorithm, compare the path cost, number of expanded nodes, and runtime, and clearly identify which algorithm(s) return optimal solutions under these conditions. A graph drawing with nodes and weighted edges must be developed. You are required to prepare the dataset yourself (nodes, road distances, and heuristic values), making any reasonable assumptions whenever needed or when exact data is not obvious.

Project-4
You will implement a solver for a **Customized Sudoku Puzzle** using constraint satisfaction techniques. The base rules of Sudoku apply that each row, column, and region must contain all digits without repetition. You are required to implement and use two algorithms: (1) plain backtracking search and (2) backtracking enhanced with CSP methods such as forward checking or arc consistency. Do some research on what customization could be done in this game, and solve the same customized puzzle. Report the difference in efficiency between the two algorithms and explain why the CSP-enhanced version performs better.

Project-5: Find another project that interests you/your group and discuss it with me.