CSA2001 - Fundamentals of AI and ML
Project Based Learning

Project 1

Design and implement an autonomous delivery agent that navigates a 2D grid city to deliver packages. The agent must:

- Model the environment (static obstacles, varying terrain costs, dynamic moving obstacles).
- Be rational: choose actions that maximize delivery efficiency under constraints (time, fuel).
- Implement uninformed (BFS/Uniform-cost), informed (A* with admissible heuristic), and a local search replanning strategy (e.g., hill-climbing with random restarts or simulated annealing) to handle dynamic obstacles / changing traffic costs.
- Compare algorithms experimentally on several map instances and report results (path cost, nodes expanded, time).
- Provide analysis describing when each method performs better and why.

Required deliverables

- Source code (well documented). Preferably Python (you may choose another language) with CLI to run each planner.
  - Encourage students to commit code to Git with README with instructions and dependencies. Tests and reproducibility are required.
  - Required at least one proof-of-concept of dynamic replanning (log showing obstacle appears and agent replans).
  - At least 4 test maps: small, medium, large, and one with dynamic obstacles (moving vehicles). Include grid file format.
- A short report (max 6 pages) containing: environment model, agent design, heuristics used, experimental results (tables + short plots), analysis and conclusion.
- A short recorded demo (5 min) or sequence of screenshots showing agents acting on dynamic map.

Constraints / assumptions

- Grid cells have integer movement cost ≥ 1 (different terrains).
- Moving obstacles occupy cells and move deterministically according to a known schedule (so agent can plan knowing future positions for one horizon) or unpredictably (for local search testing).

- Agent can move 4-connected (up/down/left/right). Diagonals optional (state in report).