Juan Barrera
Professor Mark Voortman

Module 5 Assignment (Part A) (Project Idea Layout)

Lane assignment system for athletics competitions

This assignment presents a complete, real-world system for automatic lane assignment and round progression (heats → semifinals → finals) in track events. It integrates core AI concepts from Chapter 2 (Intelligent Agents), Chapter 3 (Search), and Chapter 6 (Constraint Satisfaction Problems), and includes an optional expansion for dynamic re-assignment across rounds. It concludes with sample Python code implementing an 8-lane meet for 80 competitors, generating heats, semifinals, and finals, with the progression rule: heat winners qualify automatically, and remaining slots are filled by the next fastest times.

Problem definition

A meet organizer must assign athletes to heats and lanes fairly and reproducibly, then progress athletes into semifinals and finals. The system must follow competition logic: fastest athletes occupy central lanes, heats are balanced, and round advancement is determined by placement and time.

Inputs

- **Athletes:** Unique ID, name, seed time.
- **Track configuration:** Number of lanes (here, 8).
- **Meet parameters:** Total athletes (here, 80), event distance, round structure (heats → semifinals → finals).
- **Qualification rule:** Winners of each heat advance; remaining slots filled by next fastest times to complete the target field size for the next round.

Outputs

- **Heats:** List of heats with assigned athletes and lanes.
- **Semifinals and finals:** Progressed fields and lane assignments.
- **Decision log:** Ordering method, tie resolution, progression decisions, and reproducibility parameters.

Core constraints

- **Capacity:** Each heat has at most 8 athletes (equal to lane count).
- **Centrality pattern:** Fastest athletes in central lanes.
- **Balanced heats:** Avoid concentration of all top athletes in one heat (serpentine distribution).
- **Uniqueness:** Each athlete assigned to exactly one lane in one heat per round.

- **Progression rule application:** First in each heat auto-qualifies; remaining slots by next fastest overall times to reach target field size.

How chapters are used

Chapter 2: Intelligent agents

We model the system as a rational agent:

- **Percepts:** Athlete seed times, lane count, competition round, qualification rule.
- **Actions:** Generate heats; assign lanes; compute progression; reassign in later rounds.
- **Performance measure:** Fairness (centrality and balance), compliance (rules/constraints), reproducibility (deterministic outcomes), efficiency (timely assignments).
- **Agent function vs agent program:**
  - **Agent function:** Maps inputs (percepts) to actions (assignments).
  - **Agent program:** Implements deterministic rule-based logic and maintains internal state (rounds, assignments, and decisions).
- **PEAS for lane assignment:**
  - **Performance:** Valid, fair, reproducible assignments and progression.
  - **Environment:** Athlete pool, round structure, lane pattern.
  - **Actuators:** Assignment and progression outputs (JSON/CSV/UI).
  - **Sensors:** Input parsing, validation, and tie detection.

This ensures decisions are rational and traceable, with condition–action rules for central lane placement and progression.

Chapter 3: Solving problems by searching

We view progression and lane assignment as search in a constrained state space:

- **State space:** All valid athlete-to-lane configurations across heats.
- **Initial state:** Unassigned athletes sorted by seed time.
- **Actions:** Assign next athlete to a lane; generate next heat; compute next-round qualifiers.
- **Transition model:** Assignments produce new partial states; complete states produce round outputs.
- **Goal test:** All athletes placed; constraints satisfied; progression computed.
- **Search strategy:** Heuristic-driven greedy assignment:
  - **Serpentine distribution** to balance heats (minimizes inter-heat variance of performance).
  - **Lane order heuristic** to maximize centrality for top seeds.
- **Optimization objective:** Balance heats (variance of seed times per heat), minimize constraint violations, and maximize central lane allocation to top seeds.

This approach yields efficient, scalable generation for large fields (e.g., 10 heats for 80 athletes).

Chapter 6: Constraint satisfaction problems (CSP)

We formalize assignments under constraints:

- **Variables:** Athlete positions (heat index, lane index).
- **Domains:** Available heats and lanes.
- **Constraints:** Capacity, uniqueness, centrality pattern, balanced distribution, round-specific qualification quotas.
- **Solving approach:**
  - **Backtracking search:** Assign athletes, backtrack if constraints violated.
  - **Forward checking:** When placing an athlete, prune future invalid lane options.
  - **Arc consistency (AC-3):** Pre-prune infeasible patterns (e.g., duplicate lanes).
  - **Heuristics:**
    - **MRV:** Prioritize assignments with least available lanes (in practice, lane slots are fixed; MRV applies when constraints reduce options).
    - **Degree heuristic:** Prioritize heat slots that constrain more future placements (e.g., early heats in serpentine).
    - **Least constraining value:** Choose lane that leaves maximum flexibility for remaining athletes (matches central-first lane ordering).
- **Global constraints:** Round capacity targets and progression quotas (e.g., winners auto-qualify; remaining slots filled by fastest times).

This structure guarantees valid assignments and clean progression logic while remaining efficient.

Qualification model and round sizes

For 80 athletes and an 8-lane track:

- **Heats (Round 1):** 10 heats × 8 lanes.
  - **Qualifiers to Semifinals:** Winners of 10 heats (10) + next fastest 14 times across all heats (to reach 24 semifinalists).
- **Semifinals (Round 2):** 3 semifinals × 8 lanes = 24 athletes.
  - **Qualifiers to Final:** Winners of 3 semifinals (3) + next fastest 5 times across all semifinals (to reach 8 finalists).
- **Final (Round 3):** 1 final × 8 lanes = 8 athletes.

This adheres to the rule: first of each heat auto-qualifies; remaining slots are filled by the next fastest times to complete the target round size.

Lane assignment pattern

For 8 lanes, assign positions #1–#8 in the following lane order to center fastest athletes:

- **Lane order:** [4, 5, 3, 6, 2, 7, 1, 8]
- **Serpentine heat distribution:** Spread seeds across heats to avoid clustering top athletes.