CPSC 420 Midterm Review: Overview

- AI basics

- Search as a problem solving strategy

- Game playing

- Propositional logic

- No Lisp questions.

AI Basics

- Disciplines with ties to AI: think about how they did and would contribute

- What are the hard problems in AI? Why are they hard?

- Just read over the slides so that you have the general idea.

Uninformed Search

- Description of a search problem: initial state, goals, operators, etc.

- Considerations in designing a representation for a state

- Evaluation criteria

- BFS, DFS: time and space complexity, completeness

- When to use one vs. another

- Node visit orders for each strategy

- Tracking the stack or queue at any moment

Uninformed Search / Informed Search

- DLS, IDS, BDS search order, expansions, and queueing

- DLS, IDS, BDS evaluation

- DLS, IDS, BDS: suitable domains

- Repeated states: why removing them is important

- Constraint Satisfaction Search: what kind of domains? why important?

Informed Search

- Best-first-search: definition

- Heuristic function $h(n)$: what it is

- Greedy search: relation to $h(n)$ and evaluation. How it is different from DFS (time complexity, space complexity).

- Difference between heuristic search (or hill-climbing) and greedy search.

- A^*: definition, evaluation, conditions of optimality

- Complexity of A^*: relation to error in heuristics

- Designing good (admissible) heuristics: several rule-of-thumbs

Informed Search: Iterative Improvement Algorithms

- IDA^*: evaluation, time and space complexity (worst case)

- What is a dominant heuristic and why is it better?

- Hill-climbing basics and strategies

- Beam search concept

- Simulated annealing details: core algorithm, effect of T and ΔE, source of randomness.

Game Playing

- Game playing: what are the types of games?

- Minimax: definition, and how to get minmax values

- Minimax: evaluation

$\alpha - \beta$ Pruning

- $\alpha - \beta$ pruning: the algorithm, rationale, and why it saves time

- $\alpha - \beta$ pruning algorithm: know how to apply pruning

- $\alpha - \beta$ pruning properties: evaluation

- Games with an element of chance: what are the added elements? how does the minmax tree get augmented?

Logic

- Propositional Logic: basic laws

- Inference rules: what is inference, basic inference rules

- Limitation of propositional logic

Logic: Normal Forms and Theorem Proving

Applies to both propositional logic and first-order logic.

- Normal forms: definitions, know how to convert, applying basic laws and inference rules.

- Theorem proving: basic approaches. forward and backward chaining concept, and resolution.

Logic: First-Order Logic Specific

- Representing relations in predicate calculus: domains,

- Interpretation in predicate calculus: what is an interpretation and how it related to a domain. When is an interpretation true or false.

- prenex normal form: why it is useful, how to convert to, the basic rules used in conversion

- skolemization: why it is useful, how to do it

More FOL

- substitution and unification: why are these necessary and how to do them.

- unification algorithm

Logic: Resolution

Applies to both propositional logic and first-order logic.

- Given a theorm to prove, know how to convert it into a form suitable for resolution.

- Know how to do resolution in propositional logic

- Know how to do resolution in first-order logic

General Rule-of-Thumb

- Get used to basic concepts (representation of search problems, game playing basics, propositional and first-order logic constructs and rules)

- Try not to blindly memorize **what's there**, try to understand **why/how something works or does not work**:
  - example: time and space complexity of search strategies, alpha-beta pruning.

- Try out the exercises done in the class, and the homework.

- Try out exercises in the textbook related to the material presented in the lectures.

- Try out the past exam.

- Laws of logic will be provided with the exam.